DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Abderrahmane Mira University of Bejaia

Operations Research Department

Thesis Submitted for the Partial Fulfillment of the Requirements for the Degree of Master

Option: Data Science and Decision Aid

Hypoglycemia and Hyperglycemia Prediction using Machine Learning

Submitted by :
AMAROUCHE Faiza
OUARET Djamila

Under the supervision of : Dr. Houda EL BOUHISSI BRAHAMI

Defended on July 2nd, 2024, in front of the Jury :

Miss Z. AOUDIA	Associate Professor Class A	Chair	UAMB - Bejaia
Mr M. M'HAMDI	University Lecturer Class A	Examiner	UAMB - Bejaia
Mrs K. BOUCHEBBAH	Associate Professor Class A	Examiner	UAMB - Bejaia

Academic Year 2023/2024

Thanks

We want to avail ourselves of this opportunity and express our deep sense of gratitude to each one of those unwavering sources of support and motivational energies that helped us along the process of proceeding this thesis.

First and foremost, we express great thanks to Allah, the Merciful, for the immeasurable blessings, continuous favor, and His ever-responsive attitude to our prayers.

This work would not have been possible without the love, care, and sacrifice of our parents. Their infinite love, nurturing care regarding morals and material aspects, has provided the mainstay of our educational career.

We are deeply indebted to all our family members, friends, and colleagues who gave their helping hand in this hard-won struggle. Their encouragement and help provided much-needed impetus for us in tiding over the difficulties that came our way.

Special recognition is reserved for our esteemed supervisor, Professor Houda El Bouhissi. We would greatly appreciate the great patience, continuous support line, and very valuable pieces of advice shown by this exceptional person. Her availability and commitment to the growth of every student academically were incidental in shaping this thesis. We sincerely appreciate her dedication to our success.

We extend our heartfelt gratitude to all the teachers who have supported us throughout our educational career, right from our early years of study in primary school up to university education. Their dedication and teaching efforts have been very fundamental in laying our academic foundation, and for that, we do appreciate them very much.

This thesis represents an outcome of many individuals' collective efforts and support. We are grateful to all those who performed a part in the proceeding.

Contents

U
A
M
B

2023-2024

Abstract

Abstract

Diabetes is a critical global health issue affecting millions worldwide, with cases steadily increasing. Our research focuses on blood glucose levels prediction using a CRNN (Convolutional Recurrent Neural Network) model applied to a newly accessible and extensive dataset, ”HUPA-UCM Diabetes,” aiming to identify hyperglycemia and hypoglycemia events to enhance diabetes management accurately. This hybrid model integrates CNN and LSTM layers to effectively capture spatial and temporal dependencies in the data, achieving superior accuracy. It utilizes variables such as time, glucose, calories, heart rate, steps, basal rate, bolus volume delivered, and carb input. Our model demonstrated an RMSE value of 3.20 on the testing set, outperforming our state-of-the-art. Furthermore, real-time processing, implemented at five-minute intervals, ensures immediate responses to blood sugar variations with an RMSE of 4.63, improving patient outcomes.

Keywords: *Diabetes, hyperglycemia, hypoglycemia, blood glucose levels prediction, CRNN model, CNN, LSTM layers, real-time processing, glucose monitoring system.*

Résumé

Le diabète est un problème de santé mondial critique qui touche des millions de personnes dans le monde et dont le nombre de cas ne cesse d’augmenter. Notre recherche porte sur la prédiction de la glycémie à l’aide d’un modèle CRNN (Convolutional Recurrent Neural Network) appliqué à un ensemble de données étendu et nouvellement accessible, ”HUPA-UCM Diabetes”, visant à identifier les événements d’hyperglycémie et d’hypoglycémie afin d’améliorer la gestion du diabète de manière précise. Ce modèle hybride intègre des couches CNN et LSTM pour capturer efficacement les dépendances spatiales et temporelles dans les données et atteindre une précision supérieure. Il utilise des variables telles que le temps, le glucose, les calories, la fréquence cardiaque, les pas, le débit de base, le volume de bolus délivré et l’apport en glucides. Notre modèle a démontré une valeur de l’erreur quadratique moyenne de 3,20 sur l’ensemble des tests, surpassant l’état de l’art. En outre, le traitement en temps réel, mis en œuvre à des intervalles de cinq minutes, garantit des réponses immédiates aux variations de la glycémie avec une erreur quadratique moyenne de 4,63, ce qui améliore les résultats pour le patient.

Mots-clés: *Diabète, hyperglycémie, hypoglycémie, prédiction du glycémie, modèle CRNN, CNN, couches LSTM, traitement en temps réel, système de surveillance du glucose.*

List of Figures

List of Tables

List of abbreviations

AI: Artificial Intelligence

ANN: Artificial Neural Networks

ARIMA: Autoregressive Integrated Moving Average

AUROC: Area Under the Receiver Operating Characteristic Curve

BG: Blood Glucose

BGL: Blood Glucose Level

BLE: Bluetooth Low Energy

BMI: Body Mass Index

CGM: Continuous Glucose Monitoring

CNN: Convolutional Neural Network

CRNN: Convolutional Recurrent Neural Networks

CSV: Comma Separated Values

DT: Decision Tree

DL: Deep Learning

DNN: Deep Neural Networks

EGA: Clarke Error Grid Analysis

EHR: Electronic Health Record

ENN: Ensemble Neural Network

GDM: Gestational Diabetes Mellitus

GLD: Glucose Lowering Medications

GPU: Graphics Processing Unit

GRU: Gated Recurrent Units

HbA1c: Hemoglobin A1c (glycated hemoglobin)

IDF: International Diabetes Federation

J48: C4.5 Decision Tree Algorithm

LR: Logistic Regression

LSTM: Long Short-Term Memory network

MAE: Mean Absolute Error

MENA: Middle East and North Africa

ML: Machine Learning

MLP: Multi-Layer Perceptron

NAFLD: Non-alcoholic fatty liver disease

NASH: Non-alcoholic steatohepatitis

OneR: One Rule algorithm

RF: Random Forest

RMSE: Root Mean Squared Error

SH: Severe Hyperglycemia

SHAP value: SHapley Additive exPlanation value

SKLEARN: Scikit-learn

SMO: Sequential Minimal Optimization (for SVM)

SNS: Seaborn

SVM: Support Vector Machine

T1D: Type 1 Diabetes

T2D: Type 2 Diabetes

WHO: World Health Organization

XGBoost: Extreme Gradient Boosting

ZeroR: Zero Rule algorithm

General Introduction

Diabetes, a complex metabolic disease, has emerged as one of the leading global public health concerns, affecting millions of individuals worldwide. This chronic disorder, characterized by imbalanced blood sugar levels, manifests in various forms, yet its devastating impact on individual health is universally recognized. Its complications, including cardiovascular diseases, nerve damage, and vision problems, can significantly impair patients' quality of life and even threaten their lives.[39]

Over the past decade, the prevalence of diabetes has sharply increased in Algeria. A substantial portion of the population is either diabetic or at high risk of developing the disease due to the numerous risk factors prevalent within the community, such as obesity, inactivity, and hereditary vulnerability to the disease.[35]

Since its first description by the Greek physician Aretaeus of Cappadocia in the first century AD, to the contemporary work of eminent researchers like Sir Frederick Banting and Dr. Elliot Joslin, this silent killer has been at the center of extensive research aimed at understanding its mechanisms, improving treatments, and devising effective prevention strategies.[11]

In this context, it becomes crucial to take proactive measures to manage this disease effectively. This includes not only regular monitoring of blood glucose levels but also the ability to predict hyperglycemia and hypoglycemia episodes before they occur. Such anticipation would empower individuals with diabetes to take timely preventive actions, thereby reducing the risks of severe complications.

It is against this backdrop that our project takes on its significance. We are committed to developing a hyperglycemia and hypoglycemia prediction model, aiming to facilitate the daily management of this chronic disease. Our objective is to provide diabetic individuals with a reliable and accessible tool, capable of alerting them to potential abnormal glycemic events.

Several studies focus on this objective, with meta-analyses highlighting the power of machine learning and deep learning capabilities and their evaluation performances. In 2018, Ganjar Alfian et al.[10] obtained Root Mean Square Error (RMSE) of '25.621'. In 2020, Kezhi Li et al.[27] achieved an RMSE of '9.38'. In 2023, Francesca Iacono, Lalo Magni, and Chiara Toffanin [19] demonstrated an RMSE value prediction of '6.45'. There are many other studies, each with its specific dataset and key findings.

Our work investigates the development of a practical and dependable machine learning-based prediction tool to enhance the quality of life for diabetic patients. We hope to contribute, to making diabetes management more effective and less burdensome for those affected by this condition.

The chapters of our thesis are:

chapter 1: This chapter contains notions and subjects that concern our study, diabetes and it's types, its complications, diabetes in numbers, machine learning and deep learning.

chapter 2: This chapter discusses various works made on the prediction of diabetes, we explain the approaches of the studies and their results by developing a state of art.

chapter 3: This chapter focuses on our contribution for predicting hypoglycemia and hyperglycemia for diabetes patients. We begin by explicating the theoretical foundations that underpin our proposed approach, followed by the methodological framework and reasons of the selected evaluation techniques.

chapter 4: This chapter provides both the experimental phase and the evaluation of our proposed prediction model. It includes a detailed description of the dataset used, hardware and software environment, and a comprehensive explanation of the phases involved in implementing our approach.

1

From Diabetes in Clinical Context to Blood Glucose Prediction using Machine Learning

Introduction

In the medical field, diabetes represents a major public health challenge, affecting over 500 million people worldwide, a number that is projected to more than double to reach 1.3 billion cases over the next 30 years. This chronic disease, characterized by a dysfunction in glucose metabolism, requires constant and precise management to prevent its complications. In response to this challenge, machine learning is emerging as a revolutionary tool, offering new perspectives for the prevention and treatment of diabetes[38].

Machine learning refers to a set of techniques that enable a machine to learn from data without being explicitly programmed. This field of artificial intelligence has its origins as far back as the 1950s with the emergence of the first neural network algorithms. Since then, numerous advancements have been made with the development of deep learning in the 2000s [24]. Algorithms now analyze patient data to predict the onset of diabetes or detect its complications. Machine learning also enables the analysis of the influence of various factors on the disease to improve treatments and the quality of life of patients.

In this chapter, we focus on the medical description of diabetes, its various types, its complications, as well as the analysis of blood glucose results and its distribution. Finally, we will present a summary of the methodological approach to predicting hyperglycemia and hypoglycemia through machine learning.

1.1 Problematic

Diabetes, a serious disease that can provoke a lot of long-term complications such as cardiovascular diseases, kidney damage, vision problems, nerve damage, and foot ulcers. Consequently, rapid and accurate diagnosis and treatment are the main points to diabetes management.

In 2014, 8.5% of adults aged 18 and older had diabetes. In 2019, diabetes around the globe was the direct cause of 1.5 million deaths, and 48% of all diabetes-related deaths occurred before the age of 70. Additionally, 460,000 other deaths from kidney disease were caused by diabetes, and hyperglycemia is responsible for about 20% of deaths from cardiovascular diseases [36].

For these reasons, this work focuses on developing a robust predictive model to forecast episodes of hyperglycemia and hypoglycemia before they occur. How can we further improve the accuracy of these predictions? What are the next steps in integrating these systems into daily medical practice?

1.2 Diabetes

1.2.1 Definition

Diabetes is a chronic disease that occurs either when the pancreas does not produce enough insulin or when the body cannot effectively use the insulin it produces [39]. Insulin, discovered in 1921 by Canadian researchers Frederick Banting and Charles Best, as detailed in Michael Bliss's book "The Discovery of Insulin," is a hormone produced by the beta cells of the pancreas, a gland located behind the stomach. Insulin acts like a key, allowing glucose to enter cells, where it is used as an energy source or stored as glycogen in the liver and muscles. When there is insufficient insulin production or resistance to this hormone, the body cannot properly regulate blood glucose, resulting in abnormally high blood glucose levels, also known as hyperglycemia.[11]

According to the World Health Organization, diabetes mellitus is defined as a state of permanent hyperglycemia with fasting blood glucose levels greater than or equal to 1.26 g/l (7 mmol/l) on two occasions and/or greater than or equal to 2 g/l (11 mmol/l) at any time of the day.[39]

1.2.2 History

For 2,000 years, diabetes has stood as a formidable and often fatal disease. As far back as the first century A.D., Greek physician Aretaeus recognized its devastating effects, coining the term "diabetes" from the Greek word for "siphon." Despite this early understanding, ancient physicians like Aretaeus lacked effective treatments for the condition.[50]

In the mid-17th century, London physician Dr. Thomas Willis diagnosed diabetes mellitus by tasting his patients' urine for sweetness—a method that persisted until the 20th century.

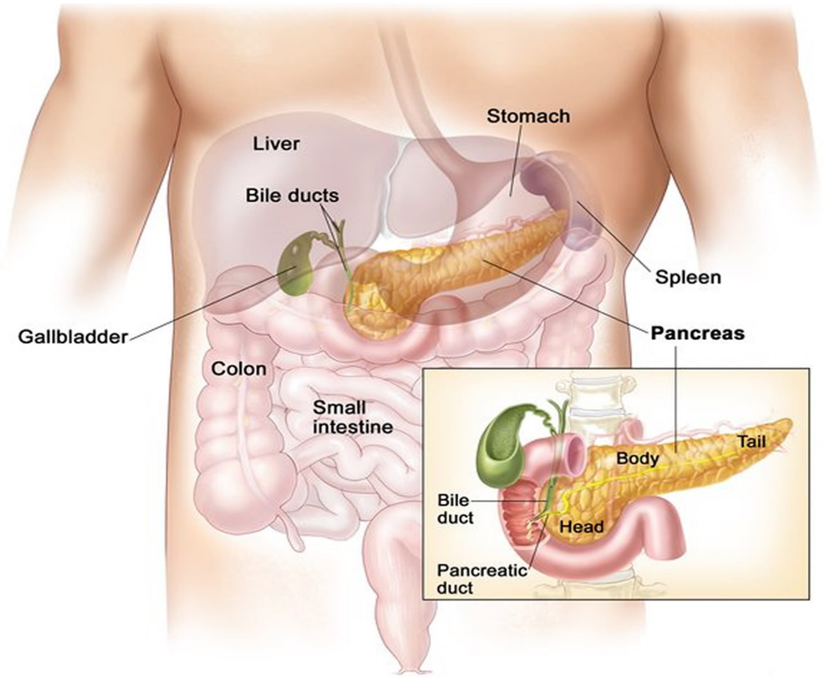


Figure 1.1: Pancreas Location [2]

Before the discovery of insulin, treatment options were limited. While low-calorie diets could extend life, they often left patients weakened and on the brink of starvation.[50]

However, a pivotal moment arrived in 1921 when Canadian doctors administered insulin to patients in critical condition, successfully normalizing their blood sugar levels. This breakthrough marked the beginning of a series of medical advancements aimed at improving the lives of those with diabetes.[50]

In the 1950s, researchers identified two distinct types of diabetes: ”insulin-sensitive” (type I) and ”insulin-insensitive” (type II). Despite millennia of progress, the quest for a cure remains ongoing. From the mysterious sickness described by Aretaeus to the groundbreaking discovery of insulin in a Canadian laboratory, each generation of physicians and scientists has contributed to our understanding of diabetes.[50]

As we enter the 21st century, diabetes researchers continue to push forward, uncertain of the exact path ahead. Whether through another revolutionary discovery akin to insulin or through incremental progress, the journey toward a cure for diabetes persists.

1.2.3 Hyperglycemia and Hypoglycemia

Understanding the phenomena of hyperglycemia and hypoglycemia relies on a thorough understanding of the basics of carbohydrate metabolism and the action of insulin.

After food consumption, carbohydrates are broken down into glucose molecules in the intestine, which are then absorbed into the bloodstream, elevating blood glucose levels. This increase in blood sugar stimulates the secretion of insulin by the pancreatic beta cells. Insulin is crucial for most cells to take up glucose. It binds to specific cellular receptors and facilitates glucose entry into the cell, where it is utilized as an energy source.

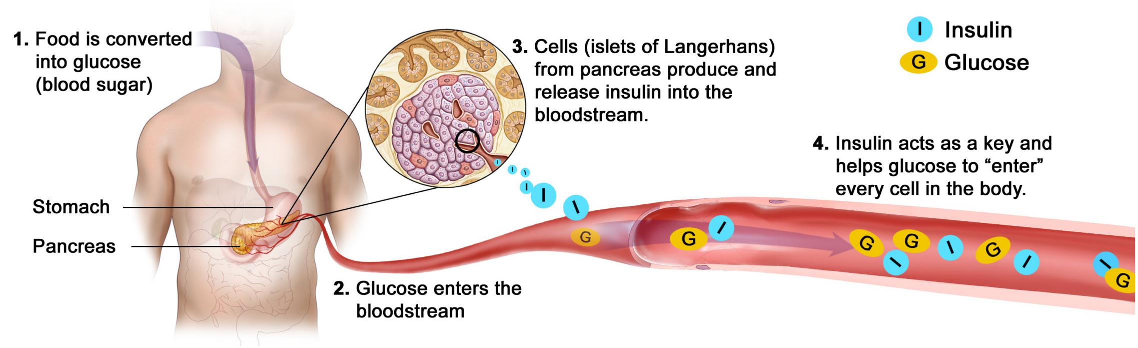

1. Food is converted into glucose (blood sugar)
Stomach
Pancreas
2. Glucose enters the bloodstream
3. Cells (islets of Langerhans) from pancreas produce and release insulin into the bloodstream.
4. Insulin acts as a key and helps glucose to "enter" every cell in the body.
I Insulin
G Glucose
I
G

Figure 1.2: The Normal Use of Glucose [5]

The increased insulin secretion by the pancreas and subsequent utilization of glucose by cells lead to a decrease in blood glucose levels. Lower glucose levels then result in a decrease in insulin secretion. If insulin production and secretion are altered by disease, blood glucose dynamics will also change. A decrease in insulin production will hinder glucose entry into cells, resulting in hyperglycemia. The same effect occurs if insulin is secreted by the pancreas but not utilized properly by target cells. If insulin secretion is increased, blood glucose levels can become very low (hypoglycemia) as large amounts of glucose enter tissue cells and little remains in the bloodstream. Several hormones can affect glycemia, but insulin is the only hormone that lowers blood glucose levels. Counter-regulatory hormones such as glucagon, catecholamines, growth hormone, thyroid hormone, and glucocorticoids all act to increase blood glucose levels, alongside their other effects.

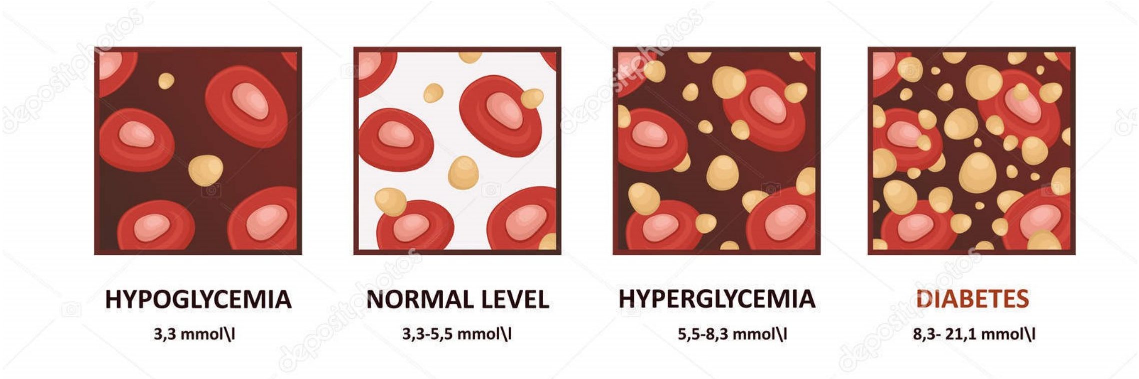


Figure 1.3: Glucose Levels in The Blood [3]

1.2.4 Analysis of blood sugar results

Blood glucose, the level of glucose in the blood, is an essential parameter to analyze in any patient suspected of or already diagnosed with diabetes. Its regular measurement through self-monitoring or in a laboratory is indispensable for disease management.

In a non-diabetic patient, fasting blood glucose should be below 1.26 g/L (7 mmol/L). Between 1.26 g/L and 2 g/L (11.1 mmol/L), fasting hyperglycemia indicates prediabetes. Beyond 2 g/L, the diagnosis of diabetes is confirmed. After an oral glucose tolerance test, a blood glucose level above 2 g/L also confirms diabetes.
[39]

In monitoring, analyzing capillary blood glucose results through self-monitoring 3 to 4 times a day, or even continuously for some patients, is crucial. It guides dosage adjustments of treatments (insulin or oral antidiabetic agents), adjustments to diet, and physical exercise.

Analyzing results of glycated hemoglobin (HbA1c), reflecting the overall glycemic balance of the past three months, is also crucial every 3 to 6 months. A level below 7% is recommended by many medical organizations to minimize the risk of long-term complications associated with diabetes. However, exceeding this target level increases the risk of developing these complications.[39]

Thus, precise and regular analysis of glycemic indicators is the key to optimal diabetes management.

1.2.5 Classification of diabetes

Diabetes is a complex disease that manifests in various forms. The two main types of diabetes, known as type 1 and type 2 diabetes, differ in their etiology, pathophysiology, and management. While type 1 diabetes results from autoimmune destruction of the pancreatic beta cells, leading to insulin deficiency, type 2 diabetes is characterized by insulin resistance and decreased insulin secretion. In addition to these two main types, there are other less common forms of diabetes, such as gestational diabetes and monogenic forms.

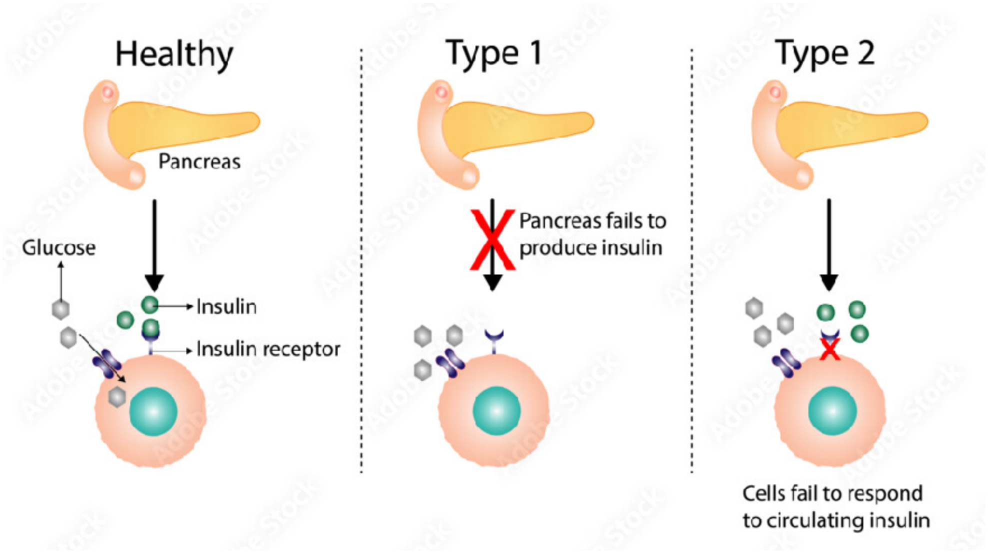

Healthy
Pancreas
Glucose
Insulin
Insulin receptor
Type 1
Pancreas fails to produce insulin
Type 2
Cells fail to respond to circulating insulin

Figure 1.4: Types of Diabetes [1]

Type 1 Diabetes

Type 1 diabetes, also known as insulin-dependent or juvenile diabetes, is characterized by low insulin production, requiring daily insulin injections. This autoimmune diabetes typically affects children and young adults, caused by the destruction of pancreatic cells that produce insulin. This leads to an absolute insulin deficiency, requiring lifelong insulin injections for survival. Without insulin, the body cannot use glucose as an energy source. While not preventable or curable, optimal control of diet, physical activity, and blood sugar can help prevent complications. According to the WHO, in 2017, 9 million people were affected by type 1 diabetes.[20]

Type 2 Diabetes

This is the most common form (90% of cases) resulting from the body's poor use of insulin (insulin resistance) and progressive depletion of pancreatic cells. This leads to chronic hyperglycemia with devastating complications (renal, ocular, neurological, cardiovascular). Often diagnosed late, this diabetes can sometimes be prevented by lifestyle changes. Type 2 diabetes is often preventable by adopting a healthy lifestyle. Contributing factors to its development include overweight, lack of exercise, and genetic predisposition. Symptoms of type 2 diabetes may be mild and unnoticed for several years. Although they may resemble those of type 1 diabetes, they are generally less pronounced. Therefore, the disease diagnosis may be delayed by several years, allowing complications to develop before being detected.[20]

Gestational Diabetes

It occurs during pregnancy in women with no known diabetes. Caused by hormonal changes, it exposes complications during pregnancy and childbirth. After delivery, it usually disappears but is associated with a high risk of developing type 2 diabetes in the following years. Screening is recommended during pregnancy. Gestational diabetes is characterized by the occurrence of hyperglycemia, meaning an elevation of glucose concentration in the blood above normal values, but at levels lower than those leading to the diagnosis of diabetes. Gestational diabetes is very often diagnosed during prenatal screening and not due to the onset of symptoms.[20]

1.2.6 Diabetes Complications

According to the International Diabetes Federation, the following statistics highlight the complications associated with diabetes:

- People with diabetes are up to three times more likely to develop cardiovascular disease.
- One in three people with diabetes will develop some form of vision loss during their lifetime
- Kidney failure is ten times more common in diabetics.
- Every 30 seconds, a lower limb is lost due to diabetes somewhere in the world

Diabetes leads to various serious complications, underscoring the risks associated with this chronic disease.

Cardiovascular Disease (CVD):

Such as heart attacks and strokes, are the leading cause of death among people with diabetes. Risk factors such as hypertension, hypercholesterolemia, and hyperglycemia significantly increase cardiovascular complications.

Eye Disease:

Including diabetic retinopathy, can lead to vision loss up to blindness. Hyperglycemia, hypertension, and hypercholesterolemia are key factors in the development of this complication, emphasizing the importance of regular eye checks.

Kidney Disease:

Resulting from damage to small blood vessels in the kidneys, leads to inefficient kidney function or kidney failure. Maintaining normal glucose and blood pressure levels significantly reduces the risk of kidney disease in people with diabetes.

Non-Alcoholic Fatty Liver Disease (NAFLD/NASH):

Non-alcoholic fatty liver disease (NAFLD) is linked to obesity and increased risk of type 2 diabetes. It can progress to non-alcoholic steatohepatitis (NASH), inducing inflammation and liver damage.

Nerve Disease:

Or Diabetic Neuropathy is caused by hyperglycemia and hypertension, damages nerves and affects digestion, erectile function, and extremities. However, preventing amputations requires comprehensive management and regular examinations.

Oral Complications:

Such as periodontitis (gingivitis), increase the risk of cardiovascular diseases. Regular oral examinations help detect these complications in a timely manner.

Diabetes during Pregnancy:

increases risks for the fetus, with potential complications, highlighting the need to achieve glycemic goals in women with type 1 and type 2 diabetes before conception to minimize risks. Additionally, high blood sugar can lead to fetal overweight and delivery complications, and increase the risk of the child developing diabetes in the future.

These elements emphasize the crucial importance of proactive diabetes management to prevent these potentially serious complications.

1.2.7 Diabetes in numbers

Based on the International Diabetes Federation's statistics in 2021, along with other sources, the following figures can be highlighted:

Diabetes Worldwide

- Approximately 537 million adults (aged 20-79) are living with diabetes.
- The total number of people with diabetes is projected to reach 643 million by 2030 and 783 million by 2045.
- Three out of four adults with diabetes live in low- to middle-income countries.

Diabetes in the MENA Region (Middle East and North Africa)

A total of 31 data sources from 18 countries were used to estimate the prevalence of diabetes among adults aged 20 to 79 in the region. Afghanistan, Bahrain, Egypt, Jordan, Lebanon, Morocco, Pakistan, and Tunisia were studied over the past five years.

- The MENA region has the highest regional prevalence, at 16.2%, and the second-highest projected increase (86%) in the number of people with diabetes, reaching 136 million by 2045.
- The MENA region records the highest percentage (24.5%) of diabetes-related deaths among working-age people.
- Only $32.6 billion USD has been spent on diabetes in the MENA region, representing 3.4% of the global total, despite the region being home to 13.6% of people with diabetes worldwide.

Diabetes in Algeria

According to the International Diabetes Federation´s latest figures, the incidence of diabetes in Algeria has increased to 7.2% of people aged from 20 to 79, or one adult every 16 people. Algeria is one of the top ten countries in the world for both the number of children with type 1 diabetes and the number of new cases of type 1 diabetes [35]. As a result, improving diabetes management remains a national public health priority.

1.3 Machine learning

1.3.1 Definition

Across history, human ingenuity has driven the creation of diverse tools – from machines that facilitate transportation and industrial processes to those that power computation – all in pursuit

U
A
M
B

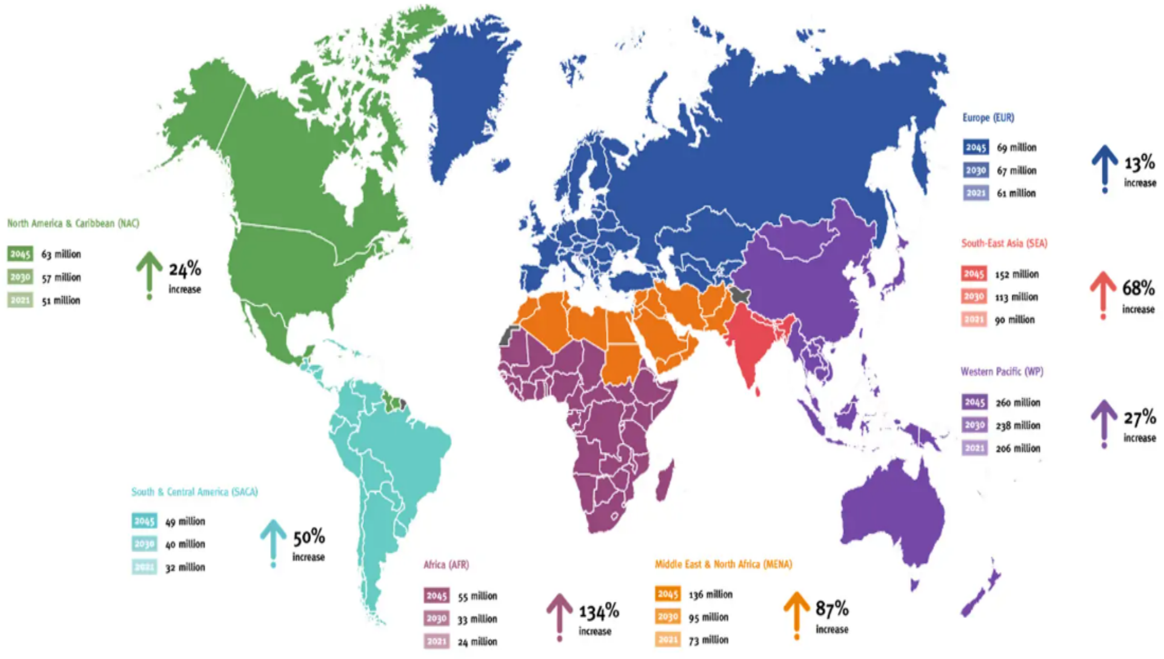


Figure 1.5: Global distribution of diabetes [20]

of a better quality of life. Among these advancements, machine learning stands out as a transformative force.

Arthur Samuel defines machine learning as ”the scientific discipline that enables computers to learn without being explicitly programmed.” Samuel gained renown for his checkers-playing program, which exemplified this concept. Machine learning (ML) is employed to teach machines how to efficiently handle data. At times, despite our efforts to analyze data, deciphering its meaning remains elusive. In such cases, machine learning comes into play, addressing the need to glean insights from complex datasets. The demand for machine learning algorithms has surged in response to the proliferation of datasets [28].

1.3.2 Types of Machine learning

There are several types of machine learning, each suited to specific situations and goals. Here are some described in the figure 1.6

Supervised Learning

Supervised learning involves learning a function that maps input parameters to an outcome using sample input-output pairs. It is a machine learning approach that relies on labeled training data, comprising examples with known outcomes, to deduce a function. Algorithms falling under supervised machine learning require external guidance. Train and test datasets are de-

2023-2024

rived from the input data, with the output variable in the train dataset requiring prediction or categorization[22].

Unsupervised Learning

Unsupervised learning is characterized by the absence of labeled data and external guidance, contrasting with supervised learning. In this approach, algorithms autonomously identify and reveal patterns within data. Unsupervised learning algorithms extract features from the data, and when new data is introduced, they utilize previously learned features to recognize data classes. It is often employed for dimensionality reduction and clustering[22].

Semi-Supervised Learning

Semi-supervised machine learning integrates aspects of both supervised and unsupervised methods. It is particularly beneficial in scenarios where obtaining labeled data is challenging or resource-intensive. With the prevalence of supervised machine learning techniques, algorithms are trained on labeled datasets, but leveraging existing unlabeled data. Various algorithms exist within the realm of semi-supervised learning, offering a hybrid approach[47].

Reinforcement Learning

Reinforcement learning explores how software agents should act within environments to maximize cumulative rewards. It is one of the fundamental paradigms of machine learning, alongside supervised and unsupervised learning. Reinforcement learning agents learn through trial and error, receiving feedback based on actions taken within an environment, aiming to optimize long-term outcomes[22].

UAMB

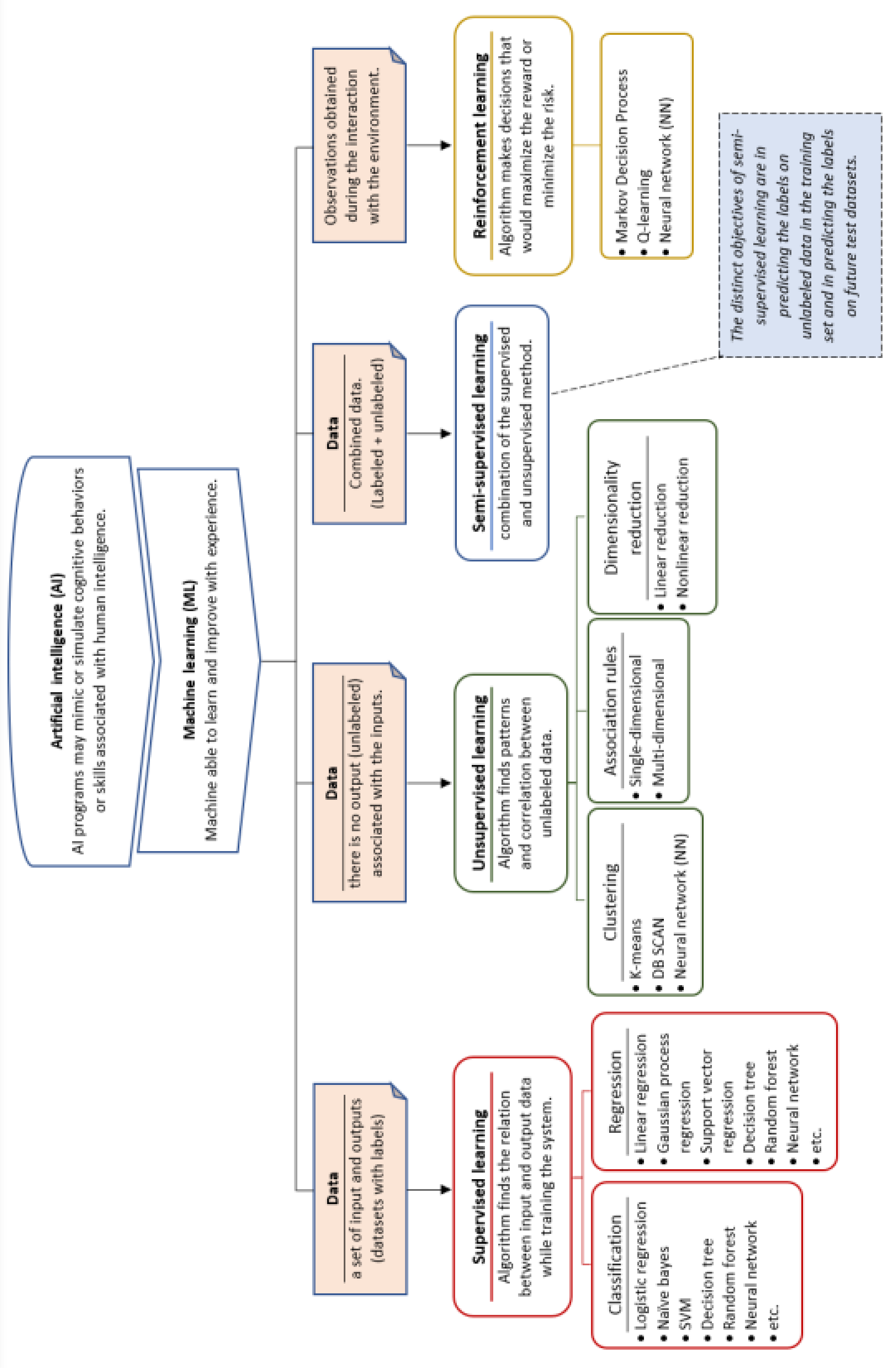

Artificial intelligence (AI)
AI programs may mimic or simulate cognitive behaviors or skills associated with human intelligence.
Machine learning (ML)
Machine able to learn and improve with experience.
Data
a set of input and outputs (datasets with labels)
Data
there is no output (unlabeled) associated with the inputs.
Data
Combined data. (Labeled + unlabeled)
Observations obtained during the interaction with the environment.
Supervised learning
Algorithm finds the relation between input and output data while training the system.
Unsupervised learning
Algorithm finds patterns and correlation between unlabeled data.
Semi-supervised learning
combination of the supervised and unsupervised method.
Reinforcement learning
Algorithm makes decisions that would maximize the reward or minimize the risk.
Classification
• Logistic regression
• Naïve bayes
• SVM
• Decision tree
• Random forest
• Neural network
• etc.
Regression
• Linear regression
• Gaussian process regression
• Support vector regression
• Decision tree
• Random forest
• Neural network
• etc.
Clustering
• K-means
• DB SCAN
• Neural network (NN)
Association rules
• Single-dimensional
• Multi-dimensional
Dimensionality reduction
• Linear reduction
• Nonlinear reduction
• Markov Decision Process
• Q-learning
• Neural network (NN)
The distinct objectives of semi-supervised learning are in predicting the labels on unlabeled data in the training set and in predicting the labels on future test datasets.

Figure 1.6: Machine Learning Types [29]

2023-2024

1.4 Deep learning

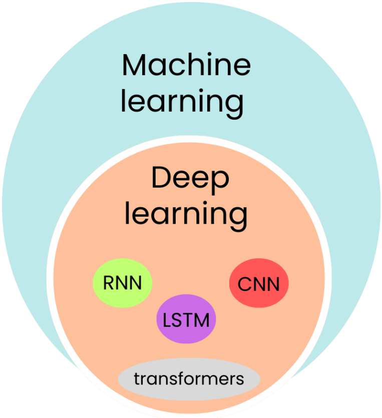


Figure 1.7: Deep learning

Among machine learning techniques, one of the subgroups that has emerged in recent years is deep learning (DL). Deep learning, known as deep neural networks, is a machine learning approach that emulates the structure of the human brain. Thus, deep neural networks can be used to solve complex problems in a very precise way.

Traditional ML algorithms often rely on handcrafted features, requiring domain expertise for tasks like spam email classification. This process can be time-consuming and limit usability. Deep learning models, on the other hand, bypass this step. By employing layers of interconnected neurons, DL models can automatically extract relevant features directly from raw data, such as email text.

The rise of deep learning can be attributed to two key factors: advancements in computing power and the increasing availability of data. Powerful hardware, particularly graphics processing units (GPUs) and distributed computing architectures, has significantly accelerated the training process for deep learning models. Additionally, the development of large datasets and advanced software libraries like TensorFlow and PyTorch has empowered researchers and practitioners to develop and train these models more efficiently.

DL has shown its effectiveness in various fields, such as medicine, computer vision, natural language processing, speech recognition and many others. For example, convolutional neural networks (CNN) are frequently used to extract local features from time series, while recurrent neural networks (RNN) and transformers are particularly well-suited to sequential tasks, capturing long-range dependencies and time series analysis.

Conclusion

This chapter has provided a comprehensive overview of the general concepts related to diabetes and its different types, complications, diabetes in numbers, definition of machine learning and deep learning. Machine learning and deep learning, two innovative and very promising areas to advance our understanding of diabetes, improve its fluctuations and treatment, and ultimately improve the quality of life of people affected by this omnipresent disease. In the next chapter, we will examine the vast body of research and studies that have been conducted in the area of blood sugar prediction. Diabetes is a global health concern, and its early detection and prediction plays a central role in effective management and management. prevention of complications. To gain a comprehensive understanding of the current state of the field, we review and analyze various approaches, methodologies, and technologies used in predicting hypoglycemia and hyperglycemia.

2
State of The Art

Introduction

The rise of artificial intelligence (AI) in healthcare has spurred research into intelligent patient care systems, with diabetes prediction being a particularly active area. This chapter reviews existing research on predicting hypoglycemia and hyperglycemia in diabetes patients, focusing on the methodologies employed in various studies.

Machine learning algorithms, particularly deep learning, play a crucial role in identifying risk factors and predicting blood sugar fluctuations. This chapter explores these methods and their performance criteria, aiming to provide a comprehensive overview of the current state-of-the-art in diabetes prediction.

2.1 Related Works

The aim of Machine Learning is to provide the machine with the ability to efficiently process large quantities of data and perform complex tasks in real time. This capability represents a major challenge for traditional algorithms, which are frequently limited in their ability to execute such processes within such tight timeframes.

Different studies have focused on the prediction of hypoglycemia and hyperglycemia from a variety of wearable smart devices. These include the work of:

Ioannis Kavakiotis et al. [22], this study explores the use of data mining and machine learning in diabetes research, highlighting the critical role these techniques play in turning massive volumes of data, including genetic and clinical data from Electronic Health Records (EHRs) into useful insights. Prediction and diagnosis, diabetic complications, genetic background and environment, and healthcare and management are its four main areas of focus.

U
A
M
B

The authors discovered that machine learning algorithms, in particular supervised learning techniques (85% of the cases), were widely used in diabetes research through a systematic literature review (focusing on articles from 2011 to 2016). Support Vector Machines (SVM) were the most widely used type of algorithm.

Notably, these methods have demonstrated potential in forecasting hyper- and hypoglycemia, comprehending problems associated with diabetes, investigating genetic and environmental factors related to diabetes, and improving healthcare.

Ganjar Alfian et al.[10], using Bluetooth Low Energy (BLE)-based sensors, the authors of this paper provide a personalized healthcare monitoring framework that gathers vital sign data, including blood pressure, heart rate, weight, and blood glucose (BG). Through the integration of real-time data processing made possible by the Apache Kafka streaming platform and MongoDB data storage solution, the system offers timely insights that enable the proper treatment of chronic illnesses, especially hyperglycemia and hypoglycemia.

The authors demonstrate through experimentation the effectiveness of the suggested real-time data processing system using commercial BLE-based sensors for tracking vital sign data in diabetes patients. Additionally, they investigate the use of machine learning algorithms for forecasting blood glucose levels and predicting diabetes using sensor data, showing the promise of Long Short-Term Memory RMSE = 25.621 and Multilayer Perceptrons Accuracy = 77.083% for precise BG level prediction and early diabetes diagnosis, respectively.

To proactively improve patient health and prevent future serious illnesses, the authors suggest combining these predictive algorithms with tailored dietary and exercise advice.

Michael Mayo, Lynne Chepulis and Ryan G. Paul[32], said that different strategies showed variable accuracies across certain glycemic subranges. Significantly, blood glucose levels within the normal and hyperglycemic ranges were best predicted by a linear Support Vector Regression (Linear SVR) model trained with normal and polynomial features, while predictions within the hypoglycemic range were best performed by a Multilayer Perceptron (MLP) trained on oversampled data; a technique used to address class imbalance to enhance accuracy on particular glycemic subranges.

It is imperative to emphasize that the dataset utilized in the studies is derived from industry standard benchmark datasets, which most likely represent a range of patient demographics and features associated with Type 1 Diabetes.

In 2020, **Kezhi Li et al.** [27] proposed a deep learning approach, specifically a convolutional recurrent neural network (CRNN), for BGLs in individuals with T1D, which was validated using the OhioT1DM dataset. The convolutional neural network (CNN) and recurrent neural network (RNN) feature extraction and temporal modeling capabilities, especially long short-term memory (LSTM) are combined in the CRNN model, which is trained on a dataset that includes data on insulin administration, carbohydrate intake, and CGM.

With a root mean squared error (RMSE) of 9.38±0.71 mg/dL over a 30-minute horizon and 18.87±2.25 mg/dL over a 60-minute horizon for simulated cases, as well as 21.07±2.35 mg/dL for 30-minute prediction and 33.27±4.79% for 60-minute prediction on real patient data, the model achieves a leading accuracy in glucose prediction. Furthermore, the model outperforms numerous conventional machine learning methods in terms of effective prediction horizons (P

Heff) with little temporal latency.

Yue Ruan et al.[45], this study aimed to assess the risk of hypoglycemia in hospitalized diabetic patients using machine learning algorithms. Its four years' worth of data, taken from the electronic records (CGMs) of a major teaching hospital, were analyzed. Between July 2014 and August 2018, the 17,658 patients and 32,758 admissions were used to compare 18 prediction models, based on the performance measure (AUROC).

Results showed that the XGBoost model, a machine learning model, outperformed traditional logistic regression models, with an AUROC of 0.96 versus 0.75. Predictive factors included items such as intervention submission, weight, diabetes type, oxygen saturation level, and use of specific medications.

Darpit Dave et al.[12], developed an optimized random forest (RF) model for probabilistic prediction of hypoglycemia risk in young people with T1D. The final model was derived after carefully considering linear models such as logistic regression (58% sensitivity and 92% specificity for a 60-minute horizon) and non-linear models such as random forests (91% sensitivity and 89% specificity for a 60-minute horizon) using a rich combination of extracted features. Other models such as decision trees (DT), gradient boosting and SVM were developed, but their results were at best similar to the two methods chosen.

In between, **Amine Rghioui et al.**[44] present an innovative system for remotely monitoring diabetes patients, leveraging contemporary technology like artificial intelligence and smart devices to optimize monitoring procedures and minimize related costs. An intelligent algorithm at the heart of this technology is intended to identify significant alterations in patients' vital signs and alert users to possible crises (emergencies). The authors have developed a portable device that can measure a diabetic patient's body temperature and blood glucose levels. It can safely connect to smartphones to provide data to healthcare professionals.

By utilizing many sensors and wireless connectivity, the system demonstrates its strength in aiding diabetic patient management and guaranteeing better results. The system also includes a data classification model that improves accuracy by automating the detection and classification of glucose level data.

Crucially, a variety of machine-learning algorithms, such as naïve Bayes, J48, ZeroR, random tree, SMO, and OneR, are used in the study to assess the system's performance. The results show that the J48 algorithm has a remarkable 99.17% accuracy, 99.47% sensitivity, and 99.32% precision in classification. High-performance indicators like these highlight how revolutionary the system can be in the treatment of diabetic patients.

Satoru Kodama et al.[23], looked at ML systems trained for hypoglycemia diagnosis or prediction through a meta-analysis of 33 papers. The findings showed that while current ML algorithms (SVM, XGBoost, RF, LR, ANN...) showed moderate potential in predicting impending hypoglycemia, with sensitivity and specificity averaging 0.80, they were limited in their capacity to detect continuing hypoglycemia. These algorithms showed some proficiency in correctly detecting genuine positive and true negative situations, but there appears to be potential for improvement in terms of dependability based on the positive and negative likelihood ratios.

The review emphasizes how important it is to take patients' unique risk profiles and hypoglycemia-related consequences into account when assessing the clinical value of machine learning algorithms. The authors suggest combining large amounts of diabetes-related data with machine learning (ML)-based decision support systems to increase the precision of hypoglycemia detection and prediction.

The research team, **Micheal O. Olusanya 1, Ropo Ebenezer Ogunsakin, Meenu Ghai and Matthew Adekunle Adeleke**[37], performed a comprehensive search of the literature and examined pertinent articles that were released between 2010 and 2021. Numerous machine learning models, including LR, ANN, DTs, RFs, SVMs, and NBayes, were used in the chosen studies. AUC, specificity, sensitivity, and accuracy were among the metrics used to assess these models' performance.

The meta-analysis's conclusions demonstrated that ML models in particular, DTs and neural networks (NN) exhibited good predictive accuracy for T2D. DT models had a reached accuracy of 0.88 and NN models an accuracy of 0.85. Its conclusion noted that it is crucial to remember that different machine learning models reveal heterogeneity, and more study is required to develop standards for choosing the best ML models for use in diabetes care.

According to **Christopher Duckworth et al.**[13], CGM data is used to predict hypoglycemia and hyperglycemia episodes through the use of machine learning models. With an AUROC of 0.998 and an average accuracy of 0.953 in hypoglycemia prediction, the XGBoost model performed excellently. Similarly, the model's average accuracy for hyperglycemia prediction was 0.931 and its AUROC was 0.989. These outperform both the logistic regression model and the simple heuristic model in terms of results.

Moreover, with an average prediction horizon of 44 minutes, the created models provide precise forecasts up to 60 minutes ahead of time. Techniques like SHAP values (SHapley Additive exPlanations) were used to improve comprehension of these forecasts. The latter provide thorough, local explanations for every prediction by figuring out the marginal impact of each input characteristic.

Another approach proposed by **Francesca Iacono, Lalo Magni and Chiara Toffanin**[19], highlight the importance of warning systems while examining the significant problem of preventing hypoglycemia and hyperglycemia in the management of T1D. The work investigates the prediction of future blood glucose levels using deep learning models, such as personalized Long Short-Term Memory (LSTM) models. Based on LSTM models, promising outcomes from individualized alarms sent out within a 40-minute time frame demonstrate an RMSE value prediction of 6.45 and precise identification of F1 = 78.79% of instances of hypoglycemia and F1 = 83.87% of cases of hyperglycemia.

The goal of successfully integrating these systems with T1D control devices—like artificial pancreases—is to enhance patient welfare and efficiency. Despite the limits of the results, this potential strategy requires more investigation, especially in actual clinical trials, to assist T1D patients in managing stressful circumstances.

The purpose of the present study of **Guangyu Wang et al.**[54] is to enhance individualized insulin titration in the treatment of T2D using a model-based reinforcement learning framework called RL-DITR. This framework analyzes the benefits of glycemic status using model-patient interaction to determine the ideal insulin prescription. In the development stage, RL-DITR achieved a mean absolute error of 1.10 ± 0.03 U, which was superior to other deep learning models and conventional clinical techniques for optimizing insulin titration. With a mean absolute error of 1.18 ± 0.09 U, a step-by-step clinical validation showed that inpatient glycemic management was improved in comparison to junior and intermediate physicians.

A proof-of-concept feasibility experiment including 16 T2D patients is also included in the study. In this trial, the mean daily blood glucose levels significantly decreased without any severe episodes of hyperglycemia or hypoglycemia with ketosis. This strategy could have a major positive clinical impact by enhancing glycemic control in hospitalized T2D patients by utilizing artificial intelligence (AI) to optimize insulin therapy. The investigators are convinced that to validate the tool's generalizability, more extensive and varied clinical trials are required.

The goal of this recent research study by **Mai ShiI and al.**[48] was to create a machine learning model that could forecast a probability that older persons with diabetes would experience severe hypoglycemia (SH) and require hospitalization. In order to train the ML model, they selected 258 predictors related to medical history, drugs, laboratory tests, and demographics using EHRs data from a sizable sample of older persons in Hong Kong.

Six distinct ML algorithms were evaluated for performance, and the findings showed that the XGBoost model performed the best (AUROC = 0.978). This performed better than an 11-variable conventional logistic-regression model (AUROC = 0.906) that included blood glucose, kidney function measures, age, sex, history of SH, hypertension, blood glucose, and use of oral glucose-lowering medications (GLDs). The top predictors of SH included factors such as non-use of lipid-regulating drugs, hospital admissions, urgent emergency triage, insulin use, and history of previous SH.

The scientists concluded that older persons who are at a high risk of experiencing severe hypoglycemia could benefit from preventive intervention by having this machine learning model integrated into electronic health record systems.

2.2 Comparative Analysis

Previously, we have presented the main prediction approaches within the healthcare domain, specifically addressing the prediction of hypoglycemia and hyperglycemia among diabetes patients. In the forthcoming , a comprehensive comparative analysis of the approaches will be conducted, structured following 6 factors:

Approach: designates the proposed approach.

Target: refers to the specific type of diabetes that is the focus of the evaluation or prediction.

Dataset: indicates the data sources used for the implementation of the approach.

Optimal Model Performance: describe the optimal technique or model applied to predict hypo/hyperglycemia.

Advantages: advantages of the approach discussed.

UAMB

Challenges: challenges or limitations of the approach discussed.

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Ioannis Kavakiotis et al.[22]	T1D / T2D	Several clinical datasets, diagnostic data and electronic health record system (EHR) **(different articles from 2011 to 2016)**	SVM (85% of the cases)	- Predictive techniques for hyper and hypoglycemia. - Understanding diabetic complications. - Exploring genetic and environmental factors linked to diabetes. - Enhancing healthcare and management strategies.	- Availability of data ”difficult and expensive to generate”. - Lack of data concerning: a)-lifestyle and behavior, b)-inheritance, and c)-linkage with other pathophysiological conditions, e.g. Alzheimer’s disease.

2023-2024

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Ganjar Alfian et al.[10]	T1D / T2D	Blood Glucose (UCI), PIMA [21]	BG based on LSTM (RMSE = 25.62%) / Classification: Multilayer Perceptron (MLP) (Precision=76.6%, Recall=77.1%, Accuracy=77.08%)	- The system helps diabetic patients manage their condition using BLE sensors and real-time data processing. - BLE sensors wirelessly collect vital signs data like blood pressure and BG for smartphones. - Real-time data processing: using Apache Kafka and MongoDB. - ML-based algorithms: MLP predicts diabetes early, while LSTM forecasts BGLs accurately. - Potential for personalized recommendations.	- Data management and scalability. - Data accuracy and reliability: ensuring quality and addressing inaccuracies are essential. - Further validation on diverse datasets is needed. - Seamless integration and user-friendly interfaces are vital for widespread adoption among diabetic patients. - Privacy and security of sensitive health data is paramount.

UAMB

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Michael Mayo et al.[32]	T1D	OhioT1D dataset [31]	Linear SVR (Hyperglycemia epidodes): MARD = 10.19 / MLP (Hypoglycemia): MARD=12.46	- ML application. - It stresses the importance of considering performance metrics and preprocessing techniques for accurate model selection. - Different regression model/preprocessing combinations exhibit varying accuracies across glycemic subranges.	- Imbalanced data. - Limited research. - Missing CGM data due to difficulties in imputing it and potential biases with existing methods. - Metric selection, like computing accuracy over the entire range may lead to misleading conclusions.

2023-2024

UAMB

Approach	Type of DM	Dataset	Optimal Model Performance	Advantages	Challenges
Kezhi Li et al. [27]	T1D	OhioT1D dataset[31]	CRNN (RMSE = 9.38 over a PH = 30-min and 18.87 over a PH = 60-minute for simulated cases)	- RMSE = 21.07±2.35 mg/dL for 30-minute prediction and 33.27±4.79% for 60-minute prediction on real patient data. - A leading accuracy in glucose prediction. - The model outperforms numerous conventional ML methods in terms of effective prediction horizons (P Heff) with little temporal latency.	- Performance Degradation on Real Patient Data. - Clinical data quality issuesUnaccounted-for factors on what affects your blood glucose. - Need for Personalized Models. - Computational demands.

2023-2024

UAMB

Approach	Type of DM	Dataset	Optimal Model Performance	Advantages	Challenges
Yue Ruan et al.[45]	All types	Data from a hospital EHR system (17658 patients)	XGBoost (Classification): AUROC = 96% (factors: submission of interventions, weight, type of diabetes, oxygen saturation level, and use of specific medications.)	- ML Application. - Utilization of Large Dataset. - Researchers evaluated 18 ML algorithms. - High Predictive Performance.	- Historical information. - Necessity of validation in diverse patient populations. - The accuracy and completeness of EHRs used may affect model performance. - Real-time trials to assess the models' effectiveness.

2023-2024

U
A
M
B

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Darpit Dave et al.[12]	T1D	CGM data (112 patients)	RF model (Sensitivity=95%, Specificity=89% for a 60 minute horizon.)	- Large Dataset. - Feature Extraction: A comprehensive set of 26 features relevant to hypoglycemia prediction was extracted from the CGM signal. - High Prediction Performance. - Evaluation of the impact of contextual information on insulin and carbohydrate intake in the prediction model.	- Pump data, which provides details on insulin administration and carbohydrate count, was available only for 19 out of the 112 patients. - The findings may not directly extrapolate to other age groups or diabetes types. - The actual implementation and real-world feasibility were not evaluated in this research. - Lack of a detailed Comparative Analysis.

2023-2024

Approach	Type of DM	Dataset	Optimal Model Performance	Advantages	Challenges
Amine Rghioui et al.[44]	Not specified	55 diabetic patients (65 days)	In classification:J48 algorithm (Accuracy=99.17%, Sensitivity=99.47%, Precision=99.32%)	- Use of AI, smart gadgets, and information and communication technology (ICTs) to cut costs for all. - Enables doctors to keep an eye on their diabetic patients. - incorporates a clever algorithm to determine when a parameter has surpassed a threshold.	- A complete evaluation or validation of the suggested system's functionality in an actual environment is not included in the paper. - No details about the potential difficulties with the real-world application and deployment of the smart glucose monitoring system.

UAMB

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Satoru Kodama et al.[23]	T1D (25 studies), T2D(3 studies) and not specified (5 studies)	33 studies (14 studies for detecting hypoglycemia and 19 studies for predicting hypoglycemia)	Sensitivity / Specificity (80%) using ML models.	- Meta-analysis methodology enables a comprehensive evaluation of ML algorithms. - Inclusion of diverse studies. - Performance metrics assessment. - Focus on clinical relevance.	- Insufficient algorithm performance, underscoring the necessity for further research and development. - Lack of detailed analysis (strengths, weaknesses, and potential sources of heterogeneity). - The article's literature search is constrained to a specific timeframe.
Micheal O. et al.[37]	T2DM	34 studies (2011-2021)	DTs (Accuracy=88%), NNs (Accuracy=85%)	- Promising prediction techniques in assisting clinicians in interpreting data and implementing effective models for T2DM prediction.	* Further research is needed to establish guidelines for selecting the most suitable ML models due to heterogeneity among the models.

2023-2024

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Christopher Duckworth et al.[14]	T1DM	153 participants aged 14 to 24 years	XGBoost: AU-ROC(hypoglycemia) = 99.8% / mean precision = 95.3% and AU-ROC(hyperglycemia) = 98.9% / mean precision = 93.1%.	- Real-time prediction. - Explainable ML using “SHAP” technique. - Comparative analysis. - Potential for personalized recommendations.	- Limited demographic diversity. - Lack of external validation. - Data availability and selection bias. - Limited prediction horizon. - Lack of long-term outcome evaluation.
Francesca Iacono et al. [19]	T1D	100 simulated patients of the UVA/Padova simulator.	LSTM (Detection of the alarm system): F1(hyperglycemia) = 83.87% and F1 (hypoglycemia) = 78.79%. Prediction (personalized LSTM): RMSE = 6.45	- Personalized Approach. - Neural Network Techniques which are specialized in analyzing sequential data. - In Silico Patient Simulator: an approved metabolic simulator. - Promising Results and effectiveness in alerting patients to critical glucose levels.	- Dataset Limitations. - Lack of External Validation. - It does not provide insights for systems in a clinical setting. - Brief description of the evaluations. - No consideration to time horizon.

U
A
M
B

Approach	**Type of DM**	**Dataset**	**Optimal Model Performance**	**Advantages**	**Challenges**
Guangyu Wang et al.[54]	T2D	16 hospitalized patients	RL-DITR (Mean Absolute Error = 1.10±0.03 U)	- Model-based RL-DITR for optimizing insulin regimens, which is a unique and innovative approach. - Comprehensive evaluation, ensuring a thorough assessment of the proposed system. - Using a large dataset of EHRs of hospitalized patients with T2D. - Superior performance.	- Limited sample size (16 patients). - Absence of long-term results. - Potential risks and safety concerns: Although the AI system underwent comprehensive evaluation events is necessary. - The size and features of the external validation dataset are not provided in the paper.
Mai ShiI et al.[48]	Not specified	1,456,618 records (older adults)	XGBoost (AU-ROC=0.978)	- Enhanced Accuracy. - Personalized risk evaluation for each individual. - Integration with EHRs.	- Limited transportability of the model. - Lack of geographically independent validation.

2.3 Interpretation of the Table

Several AI approaches can be used to build an early prediction system for hypo- and hyperglycemia, according to an analysis of the previously described research. At the moment, DL techniques, ANN, and ML algorithms are the most widely used methodologies. The studies presented in this part demonstrate various techniques for anticipating hypo- and hyperglycemia; algorithm selection and data preprocessing strategies significantly impact the predictive model's efficacy.

2023-2024

These papers show how machine learning, deep learning in particular algorithms can be used effectively to predict and manage diabetes using a variety of addresses and evaluation criteria. AI propositions provide inspiration for much ongoing research in this area in terms of improving accuracy or earlier prediction of hypo and hyperglycemia in diabetes patients. Much research is going toward these conditions being able to anticipate the two cases early enough to help in health management, which is fast getting close in reality to happening in the near future.

After an extensive review of related works, which included various models. Building upon the earlier research of **Kezhi Li et al.** [27] reaching an **RMSE of 9.38** over a prediction of 30 minutes, the objective of this research is to create a framework for combining various data types such as genetic information, clinical records, wearable device data, and real-time sensor data, to obtain a comprehensive understanding and a comprehensive picture of the patient's health state. In addition, advanced machine learning algorithms as **LSTM** networks, will be implemented to build accurate predictive models for forecasting hyperglycemia and hypoglycemia episodes.

For our thesis, we have opted to employ a hybrid deep learning model, **Convolutional Recurrent Neural Network (CRNN)** as a pivotal element in our model. As we strive to further enhance it by employing the optimal **architecture** to outcome more accurate hypo and hyperglycemia predictions. Our primary objective is to *enhance the accuracy of blood glucose levels prediction*, advancing the existing performances.

Conclusion

We conducted a state-of-the-art review in this chapter, showcasing some of the most important publications that used data mining, machine learning, deep learning, and artificial neural networks concepts and techniques to predict blood glucose levels. We read carefully over each article in these volumes, analyzing the authors' methodologies, assessing the models' output, summarizing the researchers' conclusions, and considering their suggestions for additional research. Every model exhibited its own set of strengths and limitations; some featured less-known machine learning methods like reinforcement learning, while others used more traditional algorithms like RF and SVM. Some even put forth novel deep learning strategies. Each of these methods contributed significantly to the early prediction of both hyperglycemia and hypoglycemia. Additionally, the research targets varied, encompassing different types of diabetes. Some studies focused solely on one type, such as Type 2 diabetes, Type 1 diabetes, or gestational diabetes, while others addressed multiple types, covering all three of them.

In the next chapter, we will present our approach for hypoglycemia and hyperglycemia prediction in detail.

3
Contributions

Introduction

Chronic diseases, such as diabetes, are an incurable illnesses that affect millions of people every year worldwide. Its treatment generally begins with the step of predicting blood glucose episodes, including hypoglycemia and hyperglycemia, before they occur. We are now in a situation where prediction is essential, primarily to avoid potentially very serious health complications due to inadequate diabetes management. The challenge then lies in distinctly predicting episodes of blood glucose fluctuation, but this remains to be clarified. With a multitude of variables, prediction becomes complex. The selection of the most representative characteristics and the choice of the right machine learning models for the most accurate prediction play a key role in the prediction phases.

After comparing, in the previous chapter, several works already carried out, we found that the hybrid CRNN model offers more effective results due to its ability to combine the extraction of spatial characteristics of the data with convolutional layers and the capture of temporal dependencies using recurrent layers. This combination allows the model to better understand and predict variations in glucose levels in time series, thus outperforming traditional approaches that only consider one aspect of the data.

This chapter explores the conceptual subtleties of our approach, detailing the use of the CRNN algorithm for the prediction of glycemic episodes.

3.1 Proposed approach

In this section, we describe in detail our approach to developing our prediction system related to the health field, aiming to help diabetic patients to take early action by developing an accurate

U
A
M
B

algorithm, based on individual health profiles using CRNN algorithm for prediction. The overall architecture is presented in Figure 3.1 and involves three main steps. Before detailing the steps of our proposed approach, it is essential to present the deep learning models that form the foundation of our system. These models are crucial for understanding the architecture and functionality of our hypo and hyperglycemia prediction system.

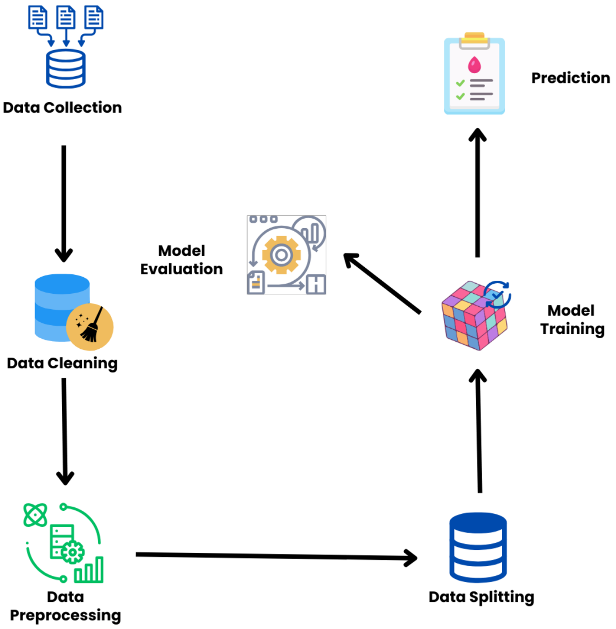


Figure 3.1: Proposed Approach

3.1.1 Deep Learning for Blood Glucose Prediction

Deep learning is a significantly strong area for forecasting blood glucose levels, so as we move forward with predictive analytics for diabetes. Common methods of deep learning are a really great improvement over existing traditional methods, simply because of their automatic potential of learning appropriate features coming out of data. By these advanced algorithms for the prediction of blood glucose levels, we aim to provide more accurate and reliable outputs to aid in the effective management of diabetes. We will describe some of these deep learning algorithms. We start by:

Recurrent Neural Networks (RNNs)

Deep learning in neural networks (NNs) is an emerging method that allows the NN to learn automatically the characteristics of data by selecting the relevant features [55], contrary to the classical NNs that require feature's selection based on domain knowledge [34].

Blood glucose, however, depends upon many factors and follows sequential patterns. Traditional deep learning architectures may not efficiently capture these long-term dependencies within the data. This is where RNNs come into play, providing a framework to model these temporal sequences effectively.

2023-2024

U
A
M
B

Definition of RNN : A special kind of artificial neural network designed to process time series or sequence-based data is the recurrent neural network (RNN). Standard feedforward neural networks are designed for independent data points only.

However, we must adjust the neural network to take into account the dependencies between the data points if the data are presented in a sequence where one data point depends on the preceding data point. In order to store the states or data from previous inputs and generate the next output in the sequence, RNNs are equipped with the idea of **"memory."**

RNN memorizes previous data. While making a decision, it takes into consideration the current input and also what it has learned from the inputs it received previously. Output from previous step is fed as input to the current step creating a feedback loop.

Let's see its basic structure in figure 3.2. Where the block 'A' captures input pattern x_t, and delivers a hidden pattern h_t as well as a final forecast value y_t. The arrow pointing in the block 'A' indicates that the information inside the block is recursively used. Once unfolded the structure, it looks like a chain of networks, as illustrated in 3.2 [26].

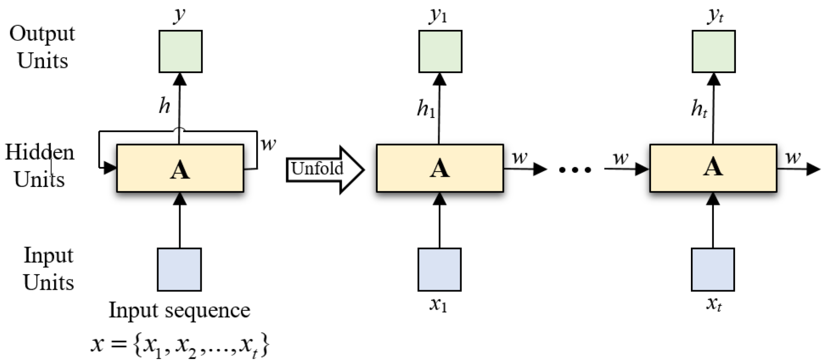


Figure 3.2: Basic illustration of an RNN

The different types of RNN are:

- One to One RNN
- One to Many RNN
- Many to One RNN
- Many to Many RNN

Vanishing and Exploding Gradients : A gradient is a partial derivative for its inputs. It measures how much the output changes with slight changes in the inputs. It, in essence, represents:

- Steeper slope = faster learning
- Near zero slope = no learning

When training a Recurrent Neural Network, gradients sometimes can become:

Exploding Gradients : When gradients grow exponentially due to large weight importance – ”too large”, thus unstable. This can be reduced by:

- Identity Initialization
- Truncated Back-Propagation
- Gradient Clipping

Vanishing Gradients : The gradients become too small, and the model stops learning or is learning very slowly. Thus, solutions include:

- Weight Initialization
- Choosing the Right Activation Function
- Using Long Short-Term Memory (LSTM) networks, which are especially good at solving the vanishing gradient problem.

These issues result : Three main points

1. Poor Performance
2. Low Accuracy
3. Long Training Periods

Strengths and Weaknesses of RNNs : Let’s see them in the summarizing table 3.1

Table 3.1: Strengths and Weaknesses of RNNs for Blood Glucose Prediction

Feature	Strengths
Sequential Data Handling	Captures sequential nature of readings, understands past influence on future
Memory of Previous Inputs	Retains information from past readings (meals, insulin)
Flexibility	Handles variable-length sequences (continuous vs. intermittent monitori
Real-Time Processing	Enables real-time prediction systems for immediate feedback

RNN Architectures : Various RNNs are being used in practice to solve machine learning problems:

Bidirectional Recurrent Neural Networks (BRNNs): Future time step inputs are utilized in BRNN to increase the network’s accuracy. Predicting the middle words of a sentence is similar to knowing its beginning and ending words.

Gated Recurrent Units (GRUs): Address the vanishing gradient problem, a common challenge in RNNs. GRUs utilize update and reset gates to control which information is retained for future predictions.

Long Short Term Memory (LSTM): To solve the vanishing gradient issue in RNNs, LSTMs were also created. LSTMs use three gates: the input, output, and forget gates. These gates determine which information to keep, similar to GRU architecture. We will get deeper in the next section.

U
A
M
B

Long Short-Term Memory (LSTM)

LSTM neural networks (NNs) are deep RNNs that Hochreiter and Schmidhuber introduced [17] to overcome the problem of exploding or vanishing gradient encountered with traditional RNNs [51]. The LSTM NNs are suitable for sequential data such as speech, video and time series as they can capture long term dependencies [15].

Definition of LSTM : An LSTM NN relies on special building blocks called memory cells. In principle, the cell state can memorize information over time due to a particular internal state. In addition, it has a sophisticated gate system for controlling information inflow and outflow from the cell state, which lets the network learn how to select relevant information and forget useless ones. Simply, the gates decide which data is important and can be useful in future and which data has to be erased.

- The memory cell contains 3 gates:

1. Input gate selects the information to be retained in the cell;
2. Forget gate decides about the information to be ignored; and
3. Output gate calculates the output and updates the hidden vector.

Squashing / Activation Functions in LSTM : Are

1. Logistic (*sigmoid*): Outputs range from 0 to 1.
2. Hyperbolic Tangent (*tanh*): Outputs range from -1 to 1.

A typical LSTM cell is illustrated as follows [58] :

The index t refers to time or sequence.
X_t : the input,
h_t : the output,
h_t^n : the hidden vector,
v_t : the input vector,
f_t : the forget gate,
i_t : the input gate,
tanh : hyperbolic tangent,
o_t : the output gate activation values, and
s_t : the memory cell state for t.

These operations are obtained as follows:

1. Forget Gate, (f_t)**:**

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \tag{3.1}$$

2023-2024

U A M B

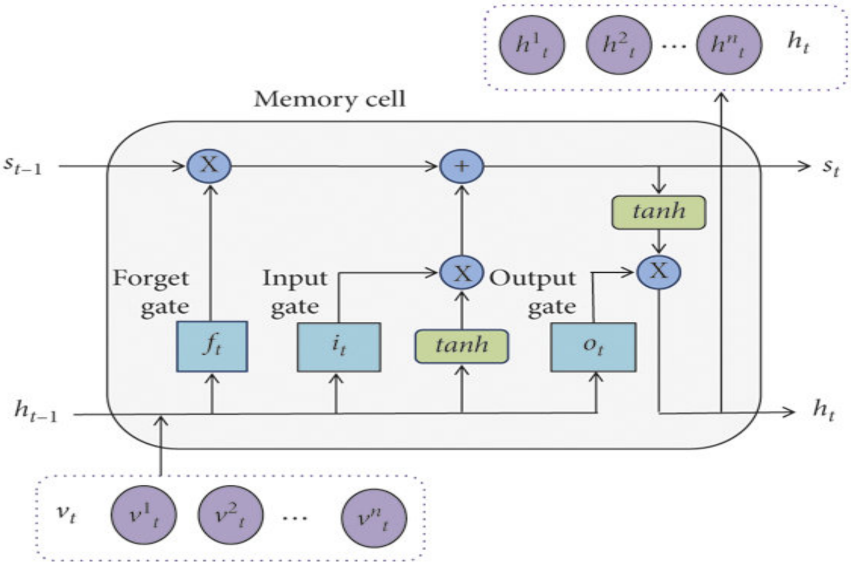


Figure 3.3: LSTM Memory Cell Structure

Where:
σ : Sigmoid activation function, this will give an output between 0 and 1,
W_f : Weight matrix for forget gate,
h_{t-1} : Hidden state from previous time step,
X_t : The current input vector,
b_f : The forget gate bias vector.

This equation tells how much information from previous memory cell state, s_{t-1} to forget. A value near 1 means more remember and a value close to 0 means more forget.

2. Input Gate:

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \tag{3.2}$$

W_i : Weight matrix of input gate,
h_{t-1} : Past hidden state,
X_t : Present input vector, and
b_i : Bias Vector of Input gate.

The output of the input gate, i_t, ranges from 0 to 1. A value close to 0 indicates blocking new information, while a value closer to 1 allows more information to flow into the cell.

3. Candidate Cell State, $\hat{C}$

$$\hat{C} = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \tag{3.3}$$

The tanh function, which produces output between -1 and 1, is applied to a linear combination of the previous hidden state (h_{t1}), current input vector (X_t), and the candidate cell state bias vector (b_c). This weight matrix W_c controls the influence of each element. The formula creates a new candidate cell state ($\hat{C}$) based on the current input and the previous hidden state.

4. Cell State, s_t

$$s_t = f_t s_{t-1} + i_t \hat{C} \tag{3.4}$$

The cell state is updated by combining information from the previous cell state (controlled by the forget gate) and the current information (controlled by the input gate). Essentially, the forget gate decides which past information to retain, while the input gate controls how much new information is integrated.

5. Output Gate, o_t

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \tag{3.5}$$

Output gate value (o_t) is computed using the sigmoid function () applied to a linear combination of the previous hidden state (h_{t-1}), current input vector (X_t), and bias vector (b_o). If one desires some elements to have more influence, this can be achieved through the weight matrix W_o.

How much of the current cell state (s_t) will be used as output (h_t) from the LSTM unit during this time step will be decided by this formula.

Convolutional Neural Networks (CNNs)

In our goal of developing a deep learning model for predicting blood glucose levels in diabetic patients, we now explore the basics behind Convolutional Neural Networks, ConvNet or CNNs for short. Traditionally, CNNs are known to be at the very core of image recognition tasks, as their convolutional layers excel in capturing spatial relationships between pixels. Their power, however, goes beyond images. Over the last few years, CNNs have been applied to a great extent in attempts for time series forecasting and have demonstrated impressive performance in feature extraction from sequential data.

This capability exploits the fact that time series data is treated like one-dimensional images. Therefore, CNNs can learn very complex patterns and temporal dependencies in sequential data. This makes them good for any kind of task related to forecasting, much more powerful than traditional methods such as ARIMA or exponential smoothing. Indeed, these traditional techniques are often unable to capture the non-linear relationships and intricate patterns that may be in the data.

In the context of blood glucose prediction for patients with diabetes, the ability of CNNs to process large amounts of data, recognize complex patterns, and develop accurate predictions makes them applicable in the case under consideration.

The section bellow will discuss in detail their architecture and considerations that need to be considered for their use in our particular application of blood glucose levels prediction.

UAMB

Definition of CNNs : CNNs are a type of deep learning algorithms particularly made to process data that contains a grid structure, generally associated with images. They are characterized by the use of convolutional layers for applying filters on input data, which automatically and adaptively learns and extracts spatial hierarchies of features. That makes them effective at visual tasks and sequential data analysis[25].

CNN Architecture : Building up from the intrinsic strengths of CNNs in time series forecasting, we can now go on to see how that takes the form of an architecture for blood glucose level prediction. The typical CNN architecture for this task includes a sequence of layers with the purposes of feature extraction and accurate prediction.

Core Layers :

- **Convolutional Layers:** These are the heart of the CNN, designed to convolve filters, much like sliding windows, over the input data for the identification of local patterns and dependencies across the blood glucose measurements and other features of interest, such as insulin intake, carbohydrate consumption, physical activities, sleep patterns or others. All these filters are specialized in extracting different features to add up for a richer understanding of the underlying relationships.

  Pooling layers : With convolutional layers comes the pooling layers, which reduce the dimensionality of the data. This prevents overfitting; a common challenge in machine learning, and allows the system to be more computationally efficient. Commonly used techniques include max-pooling and average-pooling, in which most information is summarized for a local region of the data.

- **Fully Connected Layers :** These are the last layers of the CNN, where every neuron is connected to every neuron in the previous and next layers. These layers take care of high-level features that are extracted by the convolutional layers only to be integrated at the very end. So, in our case, they put information on blood glucose measurements, the intake of insulin, or other general confounding factors together to come up with the final prediction for future blood glucose readings. It may be designed for an output layer to return a single value, such as an hour's predicted blood sugar, or a sequence of values, such as the next few hours of blood sugar levels to be predicted.

A Schematic diagram of a basic CNN architecture model for time series data forecasting is presented in 3.4 [43].

Multivariate CNN Models : Blood glucose levels are influenced by many factors. Multivariate CNNs will handle these complex inter-relationships. Such models extend every vital architecture, taking multiple time series as input. It will, therefore, let the CNN capture relationships between features and not only patterns within a feature, such as blood sugar readings. For instance, it would learn how specific intake of carbohydrates affects further

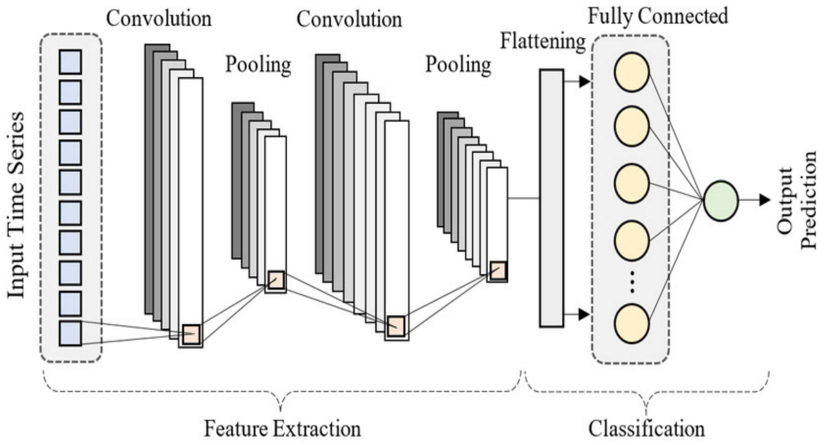


Figure 3.4: Basic CNN Architecture for Time Series Forecasting.

blood sugar levels concerning past insulin intake of the patient.

By implementing this multivariate capability, the CNN is then able to gain full insight into the factors influencing blood glucose levels to result in a more accurate prediction.

Hybrid Model : Convolutional Recurrent Neural Network (CRNN)

The Convolutional Neural Network and the Recurrent Neural Network are combined into a powerful hybrid deep learning model, called the Convolutional Recurrent Neural Network (CRNN), specifically created for time series forecasting tasks. It exploits the power of the two following established architectures:

- **CNNs :** One major use of convolutional neural networks is in learning extraction for sequential data's features. CRNN follows exactly on this because, within the model, connections of a CNN will learn relevant features in a time series input in a self-supervised manner.

- **RNNs, mainly LSTMs :** RNNs are used for modeling temporal dependencies that are part of the series. LSTMs deal with the problem of vanishing gradients in RNNs so that one can model long-term dependencies.

This will allow all **spatial features** to be extracted by CNNs—and **temporal relationships**, captured by LSTMs—fairly against multivariate time series data. Accordingly, complex forecasting problems, such as blood glucose level prediction, CRNN can be well-suited for complex forecasting areas.

3.1.2 Proposed approach steps

1. First step: "Data collection"

Data collection is the foundation of any machine learning project. The quality and relevance of the data directly impact the performance and effectiveness of the resulting model. In result, data collection is our initial step in hyperglycemia/hypoglycemia prediction and diabetes management, serving as a cornerstone for accurate model predictions. Adding to that, having an important row number is good, but understanding it is better. The comprehensiveness of the collected medical data serve as fundamental factors influencing the precision of predictive models and making good decisions.

In this study, the dataset used, is recently publicly available and sourced from Mendeley Data. It's under the name of HUPA-UCM Diabetes Dataset [16] , it encompasses 25 individuals, 14 various medical and demographic predictive variables.

Comprehensively, data collection for hyper/hypoglycemia for diabetes patients prediction, encompassing clinical and nonclinical factors associated with diabetes risk plays a pivotal role in building accurate prediction models. In essence, the more complete and representative of the data is the target population, the more reliable the model is in its predictions. Here are some key aspects of data collection for predicting hyper/hypoglycemia; we gathered all the features studies in our research's state or art collection, we have arrived to class them according their affect to the glycemia level:

Strongly Affects:

- HbA1c level: This is a key indicator of long-term blood sugar control. Higher HbA1c indicates a greater risk of both hyperglycemia and hypoglycemia.
- Physical Activity: Regular exercise helps manage blood sugar levels, reducing the risk of both hyperglycemia and hypoglycemia.
- Physical health: Illness and injuries can elevate blood sugar levels, leading to hyperglycemia.
- Consume Vegetables: Vegetables are generally low in carbohydrates and can help regulate blood sugar, reducing the risk of both hyperglycemia and hypoglycemia.
- Consume Fruit: Fruits contain carbohydrates, and while some fruits are good for blood sugar management, excessive fruit intake can contribute to hyperglycemia.

May Affect:

- Body mass index (BMI): Being obese raises the risk of hyperglycemia by increasing insulin resistance.
- Smoking history: Smoking can cause insulin sensitivity problems and blood vessel damage, which can result in hyperglycemia.
- High Cholesterol: Although not a direct cause, elevated cholesterol levels may increase the likelihood of developing various disorders that trigger high blood sugar levels.
- Mental health: Stress and depression can have an impact on eating habits and exercise levels, which can then have an indirect impact on blood sugar regulation.

UAMB

Less Likely to Affect:

- Age
- Gender
- Hypertension (high blood pressure)
- Heart disease
- Cholesterol level
- Stroke
- General health perception
- Cost of healthcare
- Health insurance coverage

2. Second step ”Pre-processing and cleaning”

Following data collection, data pre-processing and cleaning takes center stage. Pre-processing step transforms raw data into a format suitable for machine learning algorithms, aims to enhance data quality and address issues that can hinder model performance. Real-world data often contains inconsistencies such as missing values, outliers, and redundancies. These imperfections can lead to inaccurate or misleading models. Pre-processing techniques tackle these challenges, ensuring the data is clean, consistent, and ready for robust analysis.

Common pre-processing tasks include:

Missing Value Imputation: Techniques like mean/median imputation or more sophisticated methods fill in missing entries strategically.

Outlier Treatment: Outliers can be addressed through techniques like winsorization (capping extreme values) or removal if justified.

Data Cleaning: This involves identifying and correcting inconsistencies, formatting errors, and potentially removing duplicates.

Data Scaling/Normalization: Scaling ensures all features are on a similar range, preventing features with larger scales from dominating the model. Normalization transforms data to a specific range (often 0-1) for improved model convergence.

Data Transformation: This encompasses various techniques like encoding categorical variables, binning continuous variables, and creating new features based on existing ones.

2023-2024

3. Third step Prediction process

The main objective of this phase is to predict glucose levels in patients using the CRNN model, a powerful machine learning algorithm that combines the strengths of convolutional and recurrent neural networks (CNN+RNN). The CRNN model excels at capturing complex patterns in time series data, making it very powerful for regression and classification tasks. In fact the prediction made by CRNN can fall into two categories:

Regression: used when the prediction involves continuous values. For example, predicting blood glucose levels over time.

Classification: used when the prediction involves discrete labels or classes. For example, predicting if a patient is in a state of hypoglycemia, normoglycemia, or hyperglycemia.

In this model, convolutional layers first extract meaningful features from the input sequences. These features are then passed to the recurrent layers, which learn the temporal dynamics of glucose levels. The final prediction is made by a dense layer, which produces the predicted glucose level. This continuous value is then classified into discrete categories (Hypoglycemia, Normal, Hyperglycemia) based on predefined thresholds,ensuring accurate and exploitable information for patient management.

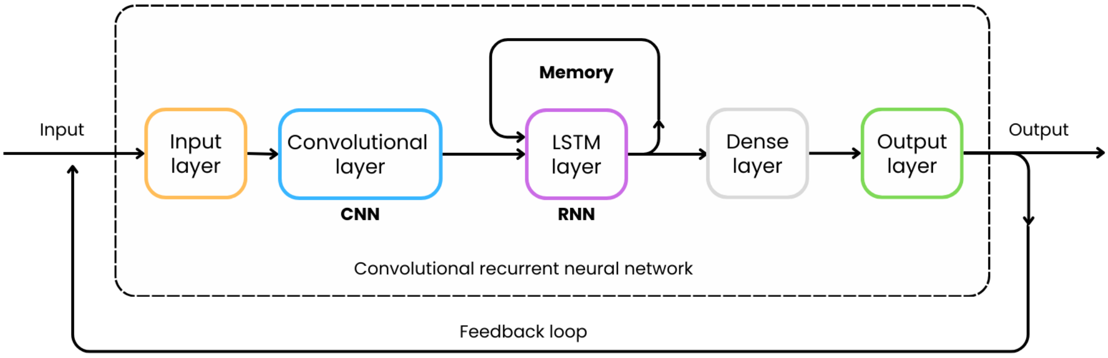


Figure 3.5: Our Proposed CRNN Model Architecture

Generally, the process follows this structure:
Phase 1: CRNN Model Construction

1. Data Preprocessing:

   - **Normalization:** Scale input data to a suitable range (0 to 1).
   - **Reshaping:** Organize data into a format suitable for the CRNN, typically a 3D array (samples, timesteps, features).

2. Feature Extraction:

   - **Convolutional Layers:** Apply convolutional layers to extract spatial features from the input data.

- **Pooling Layers:** Reduce the dimensionality of feature maps while retaining important information, it is optional.

3. Sequence Learning:

   - **Recurrent Layers:** Use LSTM layers to capture temporal dependencies and patterns in the sequential data.

4. Prediction Layer:

   - **Fully Connected Layers:** Connect the output of recurrent layers to fully connected layers.
   - **Activation Function:** Use the ReLU activation function to introduce non-linearity.

5. Output Layer:

   - **Linear Activation:** Apply a linear activation function in the output layer to produce continuous values for regression tasks.
   - **Loss Function:** Use Mean Squared Error (MSE) for regression tasks.
   - **Optimizer:** Choose an optimizer Adam to update model weights during training.

6. Compile Model:

   - **Loss Function:** Use Mean Squared Error (MSE) for regression tasks.
   - **Optimizer:** Choose an optimizer Adam to update model weights during training.

Phase 2: Prediction Generation

1. Input New Data:

   - **Preprocess Data:** Normalize and reshape the testing data similarly to the training data.

2. Generate Predictions:

   - **Forward Pass:** Pass the preprocessed data through the trained CRNN model.
   - **Output:** Obtain continuous predicted values from the output layer.

3. Post-processing (if needed):

   - **Rescale Predictions:** Convert the normalized predictions back to the original scale if necessary.
   - **Evaluation:** Compare predictions with true values using appropriate metrics (RMSE).

At the conclusion of these phases, a decision is made by evaluating the continuous values predicted by the model relative to predefined thresholds. In the context of our study, the decision involves determining whether the predicted values are within the range of normal glucose levels, or whether they indicate hypoglycemia or hyperglycemia.

Conclusion

In this chapter, we have introduced our approach to BGL prediction using advanced CRNN deep learning technique. Our system encompasses several crucial steps, including data collection, model training with CRNN, and prediction. Additionally, we explained the algorithm structure employed, demonstrating how it efficiently processes data and generates accurate predictions.

Our blood sugar prediction system represents a significant advancement in the health field. Its primary goal is to promote early detection and intervention for diabetic patient, leading to improved overall public health outcomes.

In the next chapter, we will take our research to the next level by implementing and rigorously evaluating our approach in the context of blood glucose levels prediction. Additionally, we will present the tools and development environment used in our study.

In conclusion, our BGL prediction system stands as a vital tool in the quest for proactive healthcare management. By applying cutting-edge technology to predict and prevent blood sugar fluctuations, our goal is to provide diabetic patients with the information needed to make informed decisions and, therefore, alleviate the burden of this prevalent chronic disease.

4
Experiment and Evaluation

Introduction

The overarching goal of this project is to create an advanced predictive model tailored for the early detection of hyperglycemia and hypoglycemia in T1D patients. To achieve this objective, we have chosen to leverage the Covolutional Recurrent Neural Network model, a highly regarded hybrid model renowned for its exceptional accuracy and resilience in handling complex time series data analysis.

Creating a hyperglycemia and hypoglycemia prediction system that performs excellently in term of accuracy carries profound implications for the field of diabetes management. It has the potential to substantially enhance the quality of care provided to diabetic people at risk of developing high or low blood glucose level and causing serious health complications if is unchecked earlier. By harnessing the synergistic power of cutting-edge ensemble learning, our objective is to equip healthcare professionals with a valuable and precise tool. This tool will facilitate early identification and intervention, contributing significantly to the overall well-being of T1D patients during crucial periods in their life, every good moment is precious.
Ultimately, our project aspires to revolutionize diabetes care by offering an accurate tool for early detection of blood glucose swings in T1D patients, leading to better healthtech outcomes.

This chapter is dedicated to providing a description of two interesting datasets, widely used in diabetes research. We will then discuss the development environment details, followed by an explanation of the model's algorithm and we conclude with a performance analysis of our proposed model.

4.1 Data Description

4.1.1 Description

The HUPA-UCM Dataset [16] has been created to support and ease studies Kick-only aimed at the prediction of blood glucose level, analysis of hypoglycemic and hyperglycemic events, and exploring the relationships between different physiologic variables and glucose values. It contains data from 25 T1D patients, of which there are continuous glucose monitoring and insulin delivery data, as well as physiological data, gathered for several weeks. The ages of the participants were between 18 and 65 years.

From this period, years were subject to anonymous assignment of ID numbers. Each patient either had an insulin pump therapy or multiple daily injections with CGM. They had patients perform their assessment on glucose using the FreeStyle Libre 2 CGM sensors, which measured variations in blood glucose every 15 minutes and Fitbit Ionic smart watches that provided data on **physical activity levels, heart rate, sleeping quality** during that period of data collection. Structured reporting was used to monitor the **intake of insulin and carbohydrates.**

First made available in April 2024, the HUPA-UCM Dataset [16]is intended to facilitate a range of scientific investigations and research initiatives. The collection consists of:

- **Continuous Glucose Monitoring (CGM) Data**: Recorded every 15 minutes using FreeStyle Libre 2 sensors.
- **Insulin Data**: Including basal and bolus insulin doses, recorded at 5-minute intervals.
- **Physical Activity Data**: Steps, calories burned, and heart rate data collected from Fitbit Ionic smartwatches.
- **Carbohydrate Intake**: Self-reported and estimated by participants, recorded at 5-minute intervals.

Key Variables

- **Participants**:
  - **Number**: 25 individuals with type 1 diabetes.
  - **Clinical Characteristics**: 52% female, average age 39.23 ± 11.84 years, average weight 69.06 ± 14.12 kg, average height 169.04 ± 10.41 cm, average HbA1c 7.37 ± 0.82%, average disease duration 17.8 ± 10.5 years.
  - **Therapies**: 56% on insulin pump therapy (CSII), 44% on multiple daily injections (MDI).

- **Data Types Collected**:
  - **Continuous Glucose Monitoring (CGM)**: Data collected every 15 minutes using FreeStyle Libre 2.

  - **Physical Activity**: Data on steps, calories burned, heart rate, and sleep quality collected via Fitbit Ionic smartwatches.
  - **Insulin**: Administration of insulin (basal and bolus) and carbohydrate consumption.
  - **File Format**: Data organized in CSV files for each patient, with recordings every 5 minutes.

- **Variables Included**:
  - **time**: Timestamp of data recording (format yyyy-MM-dd'T'HH:mm:ss).
  - **glucose**: Blood glucose value (mg/dL).
  - **calories**: Calories burned during the time interval.
  - **heart_rate**: Heart rate.
  - **steps**: Number of steps taken during the time interval.
  - **basal_rate**: Basal insulin infusions during the time interval.
  - **bolus_volume_delivered**: Bolus insulin injections during the time interval.
  - **carb_input**: Carbohydrate intake during the time interval (1 portion = 10 g).
- **Output**:

Classifying a time series of glucose readings highlights areas that may call for certain intervention measures or lifestyle adjustments to maintain good health. We can deduce that the primary output will be the three classes: *'Hyperglycemia', 'Hypoglycemia', and 'Normal'*, whereby one can have a pattern of changes in glucose over time.

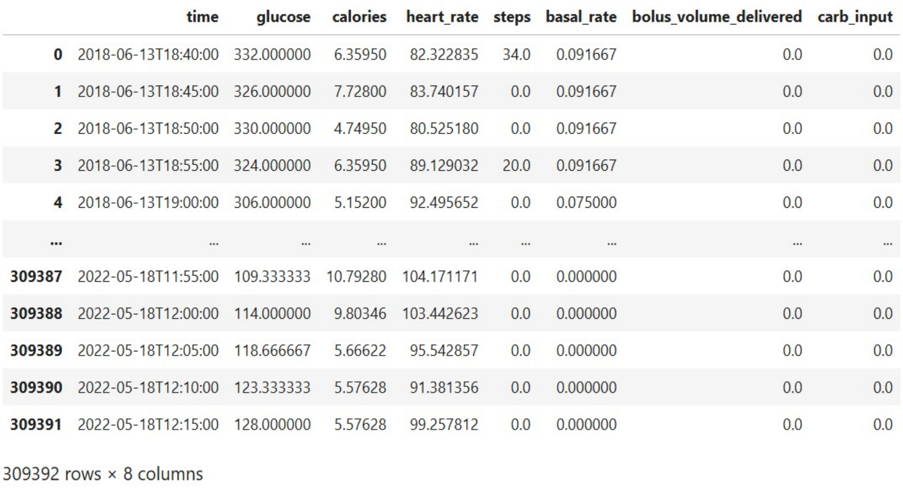
The classification will be based on predefined thresholds: above 180 mg/dL is classified as **hyperglycemia**, otherwise below 70 mg/dL are classified as **hypoglycemia**, which represents a dangerous-low blood sugar level and needs immediate attention. Those readings within these limits are considered **normal**.

4.1.2 Exploratory Data Analysis (EDA)

Overview of the dataset

First of all let's have an overview on the HUPA-UCM diabetes dataset[16] showed in the figure 4.1:

After combining the 25 patients CSVs in one dataframe, we get a dataset composed of 309,392 rows and 8 columns. Some of the captured information includes time, glucose levels, calories burned, heart rate, steps taken, basal insulin rate, bolus volume delivered, and carbohydrate input. There are uniquely different rows of entries, each timestamped, thereby giving the elaborate record of health metrics of a person over some time. The data has a varying range of glucose levels together with corresponding parameters of health, hence allowing an in-depth trend and pattern analysis.



	time	glucose	calories	heart_rate	steps	basal_rate	bolus_volume_delivered	carb_input
0	2018-06-13T18:40:00	332.000000	6.35950	82.322835	34.0	0.091667	0.0	0.0
1	2018-06-13T18:45:00	326.000000	7.72800	83.740157	0.0	0.091667	0.0	0.0
2	2018-06-13T18:50:00	330.000000	4.74950	80.525180	0.0	0.091667	0.0	0.0
3	2018-06-13T18:55:00	324.000000	6.35950	89.129032	20.0	0.091667	0.0	0.0
4	2018-06-13T19:00:00	306.000000	5.15200	92.495652	0.0	0.075000	0.0	0.0
...	...	...	...	...	...	...	...	...
309387	2022-05-18T11:55:00	109.333333	10.79280	104.171171	0.0	0.000000	0.0	0.0
309388	2022-05-18T12:00:00	114.000000	9.80346	103.442623	0.0	0.000000	0.0	0.0
309389	2022-05-18T12:05:00	118.666667	5.66622	95.542857	0.0	0.000000	0.0	0.0
309390	2022-05-18T12:10:00	123.333333	5.57628	91.381356	0.0	0.000000	0.0	0.0
309391	2022-05-18T12:15:00	128.000000	5.57628	99.257812	0.0	0.000000	0.0	0.0

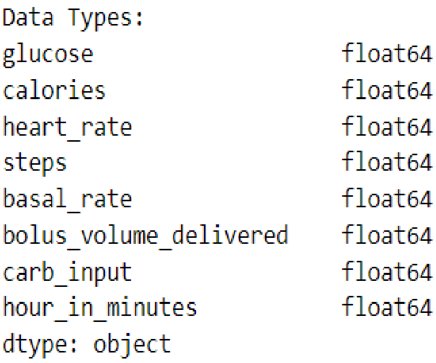
309392 rows × 8 columns

Figure 4.1: Overview of The Dataframe

Statistical summary of the Data Frame

First, we examine the 'time' variable in Figure 4.1. It is formatted as ”yyyy-MM-dd HH:mm:ss”, combining both date and time. To standardize the data type, we convert this timestamp to minutes, resulting in a float data type for all entries.

Figure 4.2 shows the data types of all variables after this conversion.



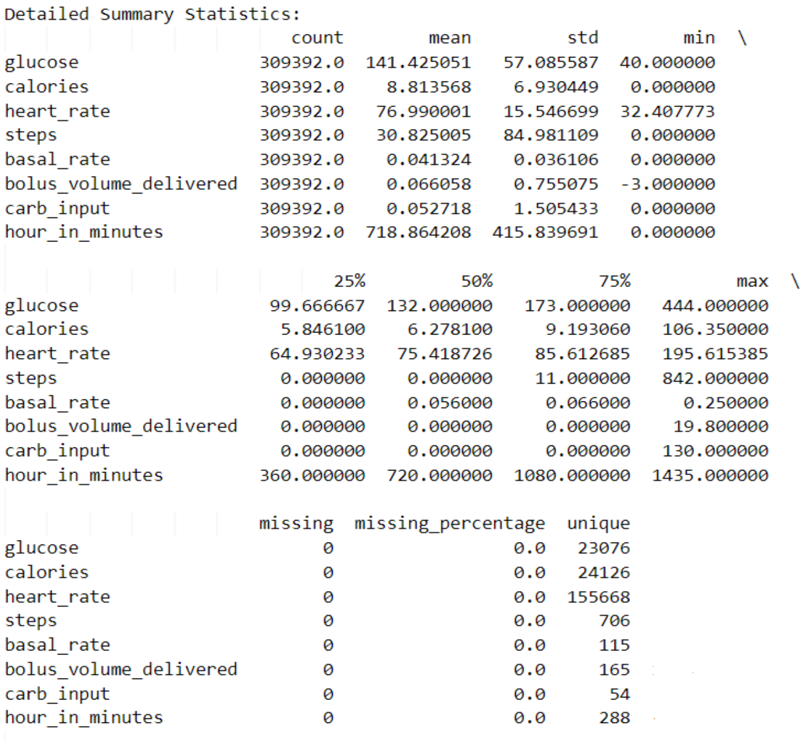
```
Data Types:
glucose                   float64
calories                  float64
heart_rate                float64
steps                     float64
basal_rate                float64
bolus_volume_delivered    float64
carb_input                float64
hour_in_minutes           float64
dtype: object
```

Figure 4.2: Data Types

Next, we generate a summary of the dataset, as shown in Figure 4.3. This summary provides a quick quantitative overview, including fundamental statistical measures such as mean, standard deviation (std), minimum (min), and other relevant metrics. To obtain these statistics, we typically use the Pandas library with the ”describe” method.

A detailed summary statistics shown in figure 4.3:

Statistical summary interpretation



```
Detailed Summary Statistics:
                           count        mean         std        min  \
glucose                 309392.0  141.425051   57.085587  40.000000
calories                309392.0    8.813568    6.930449   0.000000
heart_rate              309392.0   76.990001   15.546699  32.407773
steps                   309392.0   30.825005   84.981109   0.000000
basal_rate              309392.0    0.041324    0.036106   0.000000
bolus_volume_delivered  309392.0    0.066058    0.755075  -3.000000
carb_input              309392.0    0.052718    1.505433   0.000000
hour_in_minutes         309392.0  718.864208  415.839691   0.000000

                              25%         50%          75%          max  \
glucose                 99.666667  132.000000   173.000000   444.000000
calories                 5.846100    6.278100     9.193060   106.350000
heart_rate              64.930233   75.418726    85.612685   195.615385
steps                    0.000000    0.000000    11.000000   842.000000
basal_rate               0.000000    0.056000     0.066000     0.250000
bolus_volume_delivered   0.000000    0.000000     0.000000    19.800000
carb_input               0.000000    0.000000     0.000000   130.000000
hour_in_minutes        360.000000  720.000000  1080.000000  1435.000000

                        missing  missing_percentage  unique
glucose                       0                 0.0   23076
calories                      0                 0.0   24126
heart_rate                    0                 0.0  155668
steps                         0                 0.0     706
basal_rate                    0                 0.0     115
bolus_volume_delivered        0                 0.0     165
carb_input                    0                 0.0      54
hour_in_minutes               0                 0.0     288
```

Figure 4.3: Detailed Summary Statistics

In short, with 309,392 entries, the dataset provides a robust sample size for analysis.
- The mean glucose level is 141.43 mg/dL with a standard deviation of 57.09, indicating variability in glucose levels among the recorded instances.
- Calories burned average at 8.81, heart rate at 76.99 bpm, and steps taken at 30.83, each showing significant ranges and standard deviations, reflecting diverse activity levels.
- Basal rate, bolus volume delivered, and carb input show lower averages but with notable variability, highlighting the fluctuations in insulin administration and carbohydrate intake.
- Importantly, there are no missing values in the dataset, ensuring a complete and reliable foundation for subsequent modeling efforts.

Data Distribution Visualization

- **Histograms of numerical variables**:

Figure 4.4 represents the histograms of numerical variables that show the frequency distribution of all the numeric type columns:

Our numerical variables histograms interpretation :

- There are fewer instances of higher glucose levels, indicating that higher glucose levels are less common in the dataset.
- The calories burned show a highly right-skewed distribution, with most values concentrated between 0 and 20 calories. This indicates that in many instances, the calories burned are quite low, with a few instances of higher calorie burn.

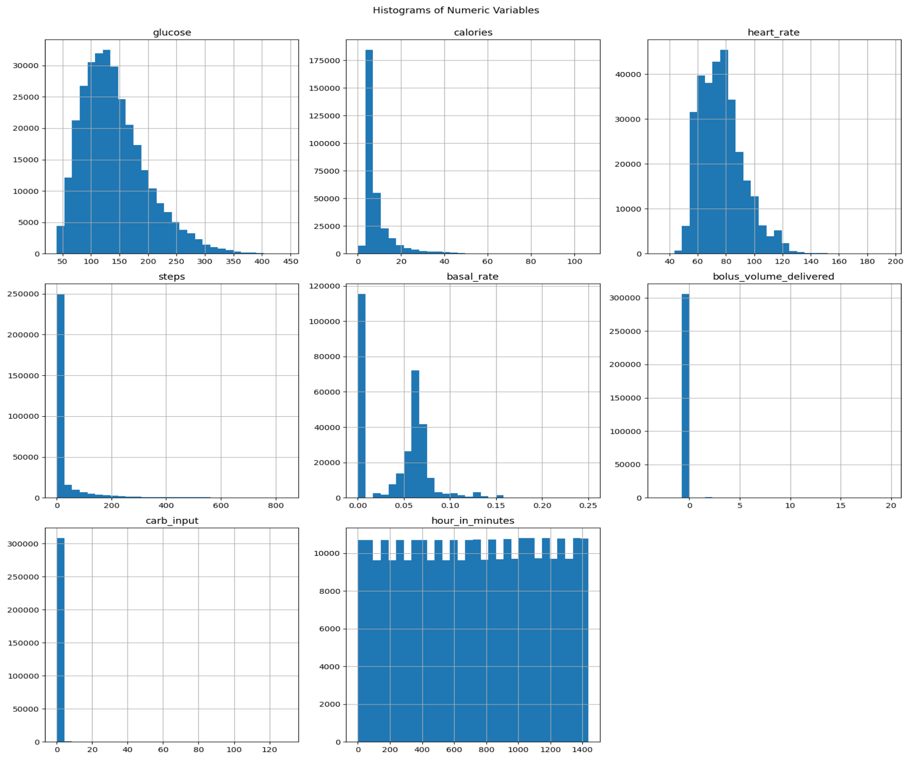


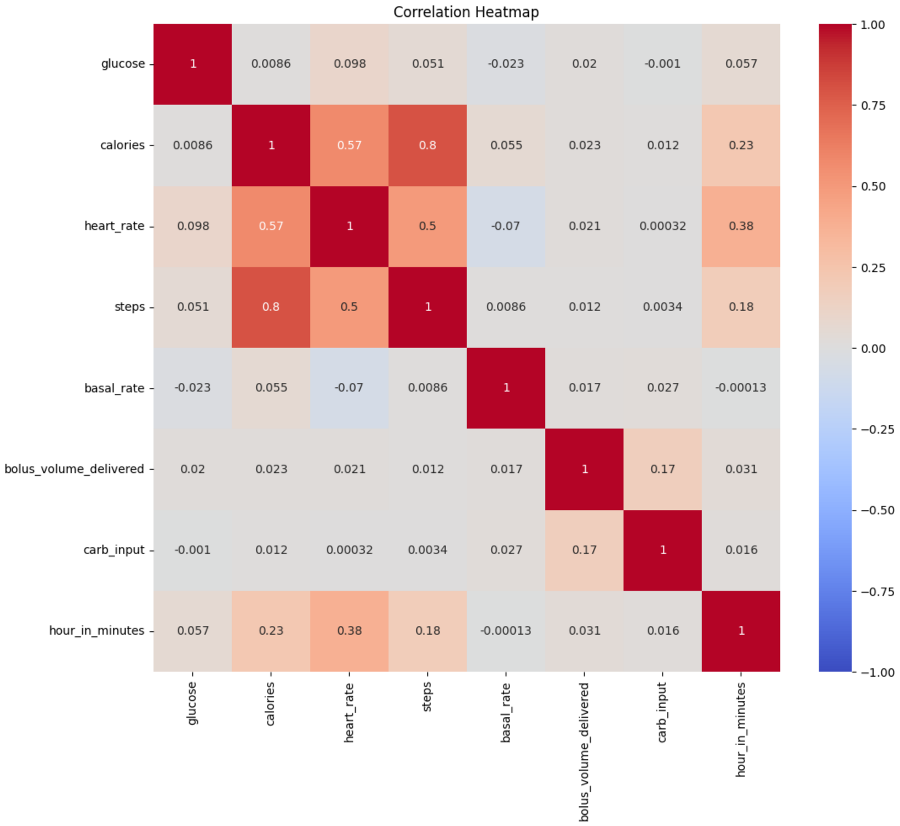
Figure 4.4: Histograms of Numerical Variables

- The heart_rate data is approximately normally distributed, with a peak around 70 to 80 beats per minute (bpm). Most values lie between 50 and 120 bpm, indicating the typical resting and active heart rates.
- The steps data shows a large number of instances having between 0 and 100 steps and there are fewer instances of higher step counts.
- The basal_rate shows a distribution concentrated around 0.05 to 0.15 units.
- In the bolus_volume_delivered visual, the vast majority of values are clustered around 0 units. This indicates that in many instances, no bolus insulin was delivered, with a few instances of higher bolus volumes.
- There are fewer instances of higher carb inputs, suggesting that higher carbohydrate intake is less common.
- The hour_in_minutes data shows a uniform distribution, indicating that the data points are evenly distributed throughout the 24-hours. This approve data collection across different times of the day.

Overall, these histograms provide a clear view of the distributions of various features in the dataset, highlighting the skewness and common ranges for each variable.

- **Correlation Matrix**:

A correlation matrix is a table that displays the correlation coefficients between many variables. Each cell in the table shows the correlation between two variables. Correlation coefficients quantify the strength and direction of a linear relationship between two features. Figure 4.5 represents a Heatmap, a correlation matrix visualization tool, where we see :



	glucose	calories	heart_rate	steps	basal_rate	bolus_volume_delivered	carb_input	hour_in_minutes
glucose	1	0.0086	0.098	0.051	-0.023	0.02	-0.001	0.057
calories	0.0086	1	0.57	0.8	0.055	0.023	0.012	0.23
heart_rate	0.098	0.57	1	0.5	-0.07	0.021	0.00032	0.38
steps	0.051	0.8	0.5	1	0.0086	0.012	0.0034	0.18
basal_rate	-0.023	0.055	-0.07	0.0086	1	0.017	0.027	-0.00013
bolus_volume_delivered	0.02	0.023	0.021	0.012	0.017	1	0.17	0.031
carb_input	-0.001	0.012	0.00032	0.0034	0.027	0.17	1	0.016
hour_in_minutes	0.057	0.23	0.38	0.18	-0.00013	0.031	0.016	1

Figure 4.5: Correlation Heatmap

Correlation analysis interpretation :

Glucose

- Shows very weak correlations with all other features.
- The highest correlation is with *heart_rate* (0.098), but it's still very weak.

Calories

- Has a moderate positive correlation with *heart_rate* (0.57) and a strong positive correlation with *steps* (0.8).
- Indicates that as the number of calories burned increases, both the heart rate and the number of steps taken tend to increase.

Heart Rate

- Moderately positively correlated with *calories* (0.57) and *steps* (0.5).
- Also shows a weak positive correlation with *heure_en_minutes* (0.38), suggesting that heart rate tends to be higher at certain times.

Steps

- Strongly positively correlated with *calories* (0.8) and moderately with *heart_rate* (0.5).
- Indicates that taking more steps is strongly associated with burning more calories and having a higher heart rate.

U
A
M
B

Basal Rate

- Shows negligible correlations with other features.
- The highest (but still weak) correlation is with *calories* (0.055).

Bolus Volume Delivered

- Shows very weak correlations with all other features.
- The highest correlation is with *carb_input* (0.17), suggesting a weak relationship between the amount of bolus insulin delivered and carbohydrate intake.

Carb Input

- Weakly correlated with *bolus_volume_delivered* (0.17).
- Indicates a slight increase in carb input is associated with an increase in bolus volume delivered.

Hours in Minutes

- Weak to moderate positive correlations with *heart_rate* (0.38) and *calories* (0.23).
- Suggests that certain times of the day might be associated with higher heart rates and more calories burned.

Overall, the results of the correlation analysis indicate that variables such as the number of steps and calories burned, although closely related to each other, do not have a strong correlation with glucose levels. This observation is essential for the modeling phase, as it drives us to adopt an approach that takes into account temporal and sequential dependencies to capture the complex dynamics influencing glucose levels.

Data cleaning

- **Not available Number (NaN) values removal**

As mentionned in the previous sections, it was essential to preprocess the database before utilizing it.

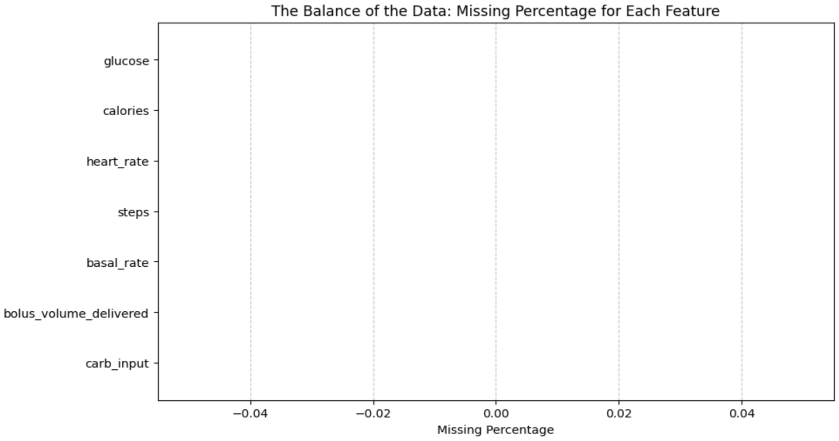


Figure 4.6: Missing Percentage Plot

As illustrated in 4.6, we have a clean data, no missing values in all our features.

2023-2024

U
A
M
B

4.2 Software, libraries and hardware

This section presents the main software, libraries and hardware that supported our current research on hyperglycemia and hypoglycemia prediction with deep learning.

4.2.1 Software

Python : Python, a versatile, open-source, high-level programming language known for its readability, extensive libraries, and large developer community. It was created in 1989 by Guido van Rossum in the Netherlands. The name Python comes from a tribute to the TV series Monty Python's Flying Circus, of which G. van Rossum is a fan. The first public version of the language was published in 1991.[7]

Python offers a robust foundation for developing and deploying deep learning models. It supports different programming paradigms like object-oriented, imperative, functional, and procedural. It was among the pioneering languages offering libraries and machine learning tools. In our implementation, we rely on several frequently used libraries [33].

Kaggle Environment : Kaggle as they say is "Your Home for Data Science". Kaggle is a web platform that hosts the largest Data Science community in the world, with over 536,000 active members in 194 countries and receives nearly 150,000 submissions per month, providing powerful tools and resources to help achieve all advancements in data science. Similar to Datascientest, Kaggle offers a customizable, no-setup Jupyter Notebooks environment. Free access to GPUs and a vast amount of data and code published by the community is available. Within Kaggle, you will find all the code and data you need to complete your data science projects. There are more than 50,000 public datasets and 400,000 public notebooks available to everyone[4].

4.2.2 Deep Learning Libraries

TensorFlow: Developed by Google and freely available at [8], TensorFlow is an open-sourced software library for performing numerical computation using data flow graphs. TensorFlow is mainly used in the domain of large-scale computations and machine learning tasks. Being powerful, it offers tools and resources for building, training, and deploying deep learning models[8].

Keras: Is an open-source library that provides a Python interface for artificial neural networks. Keras was first independent software, then integrated into the TensorFlow library, and later supporting more[52].

4.2.3 Machine Learning Libraries

Scikit-learn: This bundle of machine learning algorithms is[**?**]:
- Simple and efficient tools for predictive data analysis.
- Accessible to everybody, and reusable in various contexts.
- Built on NumPy, SciPy, and matplotlib includes everything from data preprocessing,

UAMB

feature engineering to model selection, evaluation, and visualization.
- Open source, commercially usable - Berkeley Source Distribution (BSD) license. [?]

Though Scikit initially concerns itself with areas of classical machine learning.

4.2.4 Data Analysis Libraries

NumPy: The base for scientific computing with Python [6], efficiently provides multidimensional arrays and operations, functions for linear algebra, and random number generation; it forms the basis for many other scientific Python packages, like Scikit-learn [?] and TensorFlow [8].

Pandas: Built for data manipulation and analysis [40], pandas offers high-performance, easy-to-use data structures and data manipulation of the said structures - Series and DataFrames for working with structured/tabular data. It's ideal for data cleaning and transformation building in EDA stage.

Matplotlib: General purpose plotting library for Python [?], Matplotlib provides an object-oriented interface to create a wide range of visualizations, everything from simple line plots and bar charts to complex statistical plots and histograms. Effective visualizations are needed to understand the data, the model's performance but also to be able to communicate results.

Seaborn: abbreviated as *sns*, seaborn is a Python data visualization library based on Matplotlib. It provides a high-level interface for creating informative and attractive statistical graphics. Seaborn is particularly useful for visualizing complex datasets and statistical relationships in a concise and aesthetically pleasing manner [56].

4.2.5 Hardware

Personal computer 1 Machine:

- Type: HP.
  Processeur: Intel(R) Core(TM) i5-4200U CPU @ 1.60GHz 2.30 GHz.
- Random access memory (RAM): 8,00 Go.
- Exploitation system: 64 bits, processor x64.
- Operating system: Windows 10 Professional.

Personal computer 2 Machine:

- Type: DELL Latitude E7470.
  Processor: Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40 GHz.
- RAM: 8.00 GB.

- ○ Exploitation system: 64 bits, processor x64.
- ○ Operating system: Windows 10 Professional.

4.3 Model Development and Evaluation

4.3.1 CRNN Model Algorithm

We will outline the parts of our CRNN model code, including the **training, optimizer, and fit** phases, as well as the our **optimal hyperparameters** that enabled the performance which we will present in the next sections of evaluation:

```
    # Divide data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
    =0.2, random_state=42)

    # Define and train the CRNN model
model = Sequential()
```

This line of *model* = *Sequential()* makes an instance of a Sequential model. A Sequential model in Keras refers to a linear stack of layers where you create a layer and then add another to the previous one.

```
model.add(Conv1D(filters = 64, kernel_size = 3, activation = 'relu',
    input_shape = (seq_length, scaled_features.shape[1] + 1))) #+1 for
     the glucose column
```

This adds a 1Dimentionnal Convolutional layer to the model:

- filters = 64: The layer shall learn 64 different filters.
- kernel_size = 3 : each filter will see 3 time steps at a time.
- activation = 'relu': it uses ReLU, stands for the Rectified Linear Unit activation function. input_shape = (seq_length, scaled_features.shape[1] + 1): This specifies the shape of input data.

```
    model.add(MaxPooling1D(pool_size=2))
```

This adds a Max Pooling layer, which reduces the spatial dimensions of the output from the previous layer. *pool_size* = *2* means it will take the maximum value over each 2 adjacent values.

```
    model.add(LSTM(50, return_sequences=True))
```

It adds an LSTM layer, with a number of 50 units as parameters. The returning of sequences, *return_sequences = True*, from this layer, that is, the full sequence of outputs for every input sequence, is necessary to be considered in the case of stacking LSTM layers.

```
    model.add(LSTM(50))
```

This will also add another LSTM layer, with 50 units. This layer will return the last output only for each of the input sequences.

```
    model.add(Dense(1))
```

This adds a fully connected (Dense) layer with 1 unit, which will be the output of the model.

```
    # Compile the model with Adam optimizer and mean squared error loss
    model.compile(optimizer=Adam(learning_rate=0.001), loss=’mean_squared_error’)
```

It will compile the model, setting the applied optimizer to *Adam* with a learning rate of 0.001, and the loss function to the MSE

Adam Optimizer is chosen to adjust model weights and minimize the loss function effectively.

MSE (Mean Squared Error) is utilized as the measure of prediction accuracy, ensuring the model minimizes the squared differences between predicted and actual values during training.

```
# validation_split: fraction of training data to use as validation data
model.fit(X_train, y_train, epochs=50, batch_size=32, validation_split=0.2)
```

This trains the model on the data. It uses:

- *X_train* and *y_train* as the training data.
- 50 epochs (full passes through the dataset).
- A batch size of 32 (number of samples processed before the model is updated).
- 20/80% splitting, 20% of the training data is used for validation.

In summary, the combination of CNN and LSTM layers allows the model to capture both short-term patterns and long-term trends in BGLs as we mentioned above. This architecture aims to create a model that can effectively learn from our glucose time series data, ”HUPA-UCM Diabetes dataset,” to predict future events for hyperglycemia and hypoglycemia.

These architectural choices and hyperparameters were selected after long experimentation with various configurations to optimize performance. We have tried several combinations of hyperparameters iteratively to finally get the one best for our prediction task. Figures 4.8, 4.9 summarize our various findings when adjusting the values of the chosen hyperparameters.

4.3.2 Evaluation metrics

The main evaluation techniques for forecasting goals are listed in this table**??**, then we will provide the metrics to be employed in our evaluation:

Category	**Metric**
Error Metrics	Mean Squared Error (MSE) Average squared difference between predicted and actual values. Lower MSE indicates better performance.
	Root Mean Squared Error (RMSE) Square root of MSE; error in the same units as the original data. Easier to interpret.
	Mean Absolute Error (MAE) Average absolute difference between predicted and actual values. Less sensitive to outliers than MSE.
	Mean Absolute Percentage Error (MAPE) Average error as a percentage of the actual values. Useful for comparing across datasets with varying scales.
Loss Functions	Binary Cross-Entropy (BCE) Commonly used for binary classification tasks (e.g., hyper vs. normoglycemia). Measures difference between predicted probability and true binary labels.
	Categorical Cross-Entropy (CE) Extension of BCE for multi-class classification (e.g., predicting different blood sugar levels). Average CE across all classes.
Correlation Metrics	Pearson Correlation Coefficient (PCC) Measures linear correlation between predicted and actual values. 1: perfect positive correlation, 0: no correlation, -1: perfect negative correlation.
	Spearman Rank Correlation Coefficient (SRCC) Similar to PCC but assesses monotonic relationship between rankings. Less sensitive to outliers than PCC.
Informative Metrics	F1-Score Harmonic mean of precision and recall, useful for imbalanced datasets. Balances true positive identification and avoiding false positives.
	Area Under the ROC Curve (AUC) Probability that the model ranks a random positive instance higher than a random negative instance. 1: perfect performance, 0.5: random guessing.

Table 4.1: Key Performance Metrics for Forecasting

In our study, since a single indicator is insufficient to offer a thorough analysis, five criteria are used to assess the testing performance and measuring the error between predicted BGLs and original values.

Specifically, we employ the :

- **Mean Squared Error (MSE)**: MSE measures the average of the squares of the errors—that is, the average squared difference between the estimated values ($\hat{y}_i$) and the actual value (y_i). A smaller MSE value indicates better model performance. The formula for MSE is depicted in Equation (4.2).

- **Root Mean Squared Error (RMSE)**: RMSE is the square root of the average of the squares of the errors. It provides an aggregate measure of the magnitude of the errors in a set of predictions. The formula for RMSE is depicted in Equation (4.3).

- **Mean Absolute Error (MAE)**: MAE measures the average of the absolute differences between the predicted values ($\hat{y}_i$) and the actual values (y_i). It is a measure of the average magnitude of the errors in a set of predictions, without considering their direction. A smaller MAE value indicates better model performance. The formula for MAE is depicted in Equation (4.1).

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|, \tag{4.1}$$

U
A
M
B

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \tag{4.2}$$

And,

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{4.3}$$

Where N denotes the total number of BG points in the dataset, $\hat{y}_i$ is the prediction value and y_t is the original (actual) value.

- **R-squared (R²) Score**: Tells us how well the model explains the variance within our data; that gives an idea of the goodness of fit our model has. Equation (4.4) shows how the R² score is calculated.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}i)^2}{\sum i = 1^N (y_i - \bar{y})^2}, \tag{4.4}$$

Where $\bar{y}$ is the mean of the actual values.
The R² score is a value between 0 and 1, where 1 is perfect prediction. The higher the R² score, the more fitted the model is to the data.

- **F1 Score**: The F1 score is a measure of a model's accuracy, calculated as the harmonic mean of precision and recall. It balances the trade-off between precision and recall and gives a score between 0 and 1. A higher F1 score indicates better model performance. The formula for the F1 score is depicted in Equation (4.5).

$$\text{F1 Score} = 2 \cdot \frac{\text{True Positives}}{2 \cdot \text{True Positives} + \text{False Positives} + \text{False Negatives}} \tag{4.5}$$

  - **True Positives (TP)**: The number of correct predictions that an instance is positive.
  - **False Positives (FP)**: The number of incorrect predictions that an instance is positive.
  - **False Negatives (FN)**: The number of incorrect predictions that an instance is negative.

4.3.3 CRNN Model Evaluation

In the previous studies of Kezhi Li et al. [27] and Francesca Iacono Et al.[19], it was CRNN algorithm outperformed the other approaches in 2020 and Francesca Iacono Et al.[19] study, they found that LSTM was their optimal model in BGLs prediction. Nevertheless, to enhance and ensure the CRNN model's performance noting that LSTMs are powerfull, we opted to employ a new real dataset [16] of BG records of T1D patients, using a new CRNN model architecture.

U
A
M
B

In addition, we tried training the model on cloud accessible GPU of Kaggle plateform [4]. It is noteworthy that we applied the CRNN model to a recently released public dataset (in april, 2024) that has not yet been employed for CRNN modeling by other researchers, described in the preceding titles.

Combining CNNs + RNNs, particularly "LSTM", can offer several advantages in many applications, especilly in deep learning and prediction tasks. This combination, often referred to as Convolutional LSTM or ConvLSTM, covering the strengths of both architectures. Here are some key advantages:

1. Feature extraction and temporal dynamics:
   - CNN: Excels at extracting spatial features from the input data.
   - LSTM: Good at capturing time dependencies and long-term patterns in sequential data.
   - Combined: It can effectively handle the spatiotemporal data; it is very much fit for tasks that have spatial and temporal elements.
2. Handling multidimensional and multivariate series of time data.
3. Fewer parameters: CNN layers can lower the dimensionality of the input fed into LSTM layers; this might lead to fewer parameters as compared to an approach only based on LSTMs.
4. Feature representation: CNNs learn hierarchical features which are a better input to LSTMs in modeling their temporality.
5. Flexibility, like using equential data.
6. Variable-length input handling: The combination, like standard LSTMs, also can handle variable-length inputs.
7. Capturing both global and local patterns: CNNs may capture the local patterns, while LSTMs are capable of capturing longer-range dependencies.
8. Better generalization: Spatial and temporal feature learning provide a combination that can be leveraged to enroll better generalization of the model on unseen data.

These benefits make CNN+LSTM combinations particularly useful in areas like time series forecasting with spatial components, for our context, in complex medical data analysis like glucose level prediction.

To evaluate the effectiveness of our CRNN model architecture and it's chosen hyperparameters, we conducted a thorough analysis using the Kaggle environment [4]. Our evaluation process involved using the HUPA-UCM Diabetes Dataset, which we divided into two distinct sets: one consisting of 80% of the data for training and the other containing 20% for testing.

The evaluation consisted of two main phases:

Prediction with CRNN model with different parameters

In this phase, we employed the Convolutional Recurrent Neural Network algorithm to make predictions on the main dataset. Here is our principal CRNN architecture

1. Input: Sequential data with shape (seq_length, number_of_features + 1), where the +1 accounts for the glucose column.
2. Conv1D layer:
   - 64 filters to learn various local patterns
   - Kernel size of 3 to look at 3 time steps at once
   - ReLU activation for non-linearity
3. MaxPooling1D layer:
   - Pool size of 2 to reduce spatial dimensions and computational load
4. Two LSTM layers:
   - 50 units each for capturing long-term dependencies.
   - First layer returns sequences, allowing stacking.
5. Dense output layer:
   - unit for predicting future glucose levels.

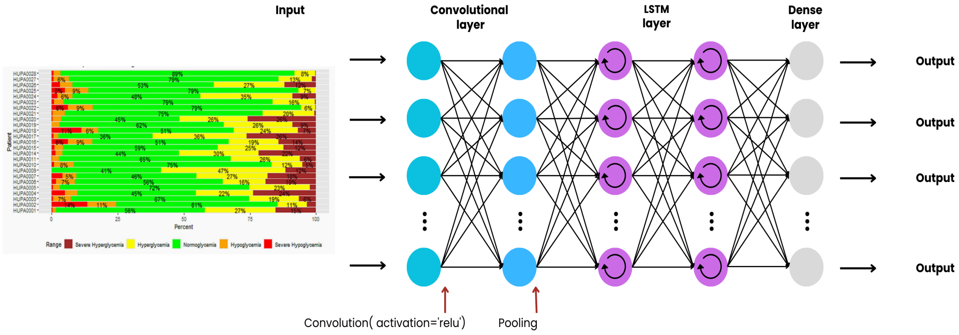
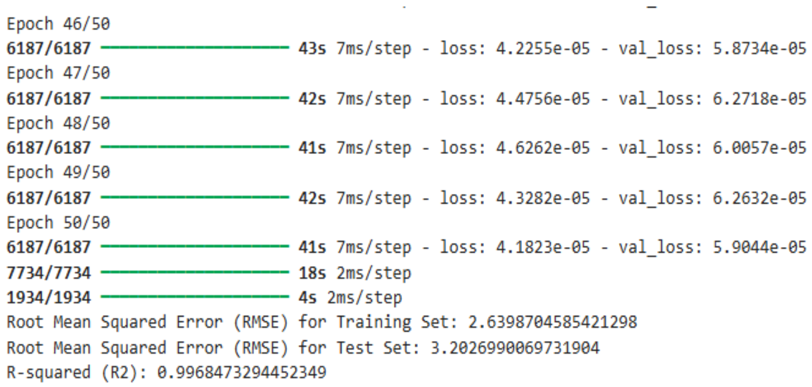


Figure 4.7: Prediction with CRNN Model

Then we follow up with two experiments with two different hyperparameters for each:

1. **First Experiment** using :
   - Learning rate: 0.001
   - Loss function: MSE
   - Epochs: 50
   - Batch size: 32



```
Epoch 46/50
6187/6187 ━━━━━━━━━━━━━━━━━━━━ 43s 7ms/step - loss: 4.2255e-05 - val_loss: 5.8734e-05
Epoch 47/50
6187/6187 ━━━━━━━━━━━━━━━━━━━━ 42s 7ms/step - loss: 4.4756e-05 - val_loss: 6.2718e-05
Epoch 48/50
6187/6187 ━━━━━━━━━━━━━━━━━━━━ 41s 7ms/step - loss: 4.6262e-05 - val_loss: 6.0057e-05
Epoch 49/50
6187/6187 ━━━━━━━━━━━━━━━━━━━━ 42s 7ms/step - loss: 4.3282e-05 - val_loss: 6.2632e-05
Epoch 50/50
6187/6187 ━━━━━━━━━━━━━━━━━━━━ 41s 7ms/step - loss: 4.1823e-05 - val_loss: 5.9044e-05
7734/7734 ━━━━━━━━━━━━━━━━━━━━ 18s 2ms/step
1934/1934 ━━━━━━━━━━━━━━━━━━━━ 4s 2ms/step
Root Mean Squared Error (RMSE) for Training Set: 2.6398704585421298
Root Mean Squared Error (RMSE) for Test Set: 3.2026990069731904
R-squared (R2): 0.9968473294452349
```

Figure 4.8: CRNN Model Predictions Results ”RMSE” on HUPA-UCM Dataset with 50 Epochs

- Validation split: 0.2

And these are the performances we get, see figure 4.8:

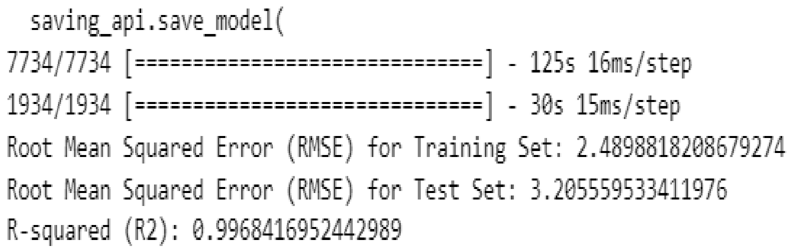
The obtained results suggest that our model is well-fitted to the data and capable of predicting glucose levels with high accuracy, evidenced by :

- A low RMSE of 3.20 indicating precise glucose level predictions. Furthermore, there is a negligible difference between the RMSE of the test set and the training set, indicating that there is no overfitting or underfitting.
- A high $R^2 = 99.6\%$ signifies that our model effectively captures relationships between features and the target variable (glucose).

2. **Second Experiment** using :

   - Learning rate: 0.001
   - Loss function: MSE
   - Epochs: 100
   - Batch size: 32
   - Validation split: 0.2

These are the performances we get with Epochs = 100, see figures 4.9 and 4.10:



```
  saving_api.save_model(
7734/7734 [==============================] - 125s 16ms/step
1934/1934 [==============================] - 30s 15ms/step
Root Mean Squared Error (RMSE) for Training Set: 2.4898818208679274
Root Mean Squared Error (RMSE) for Test Set: 3.205559533411976
R-squared (R2): 0.9968416952442989
```

Figure 4.9: CRNN Model Predictions Results on HUPA-UCM Dataset with 100 Epochs

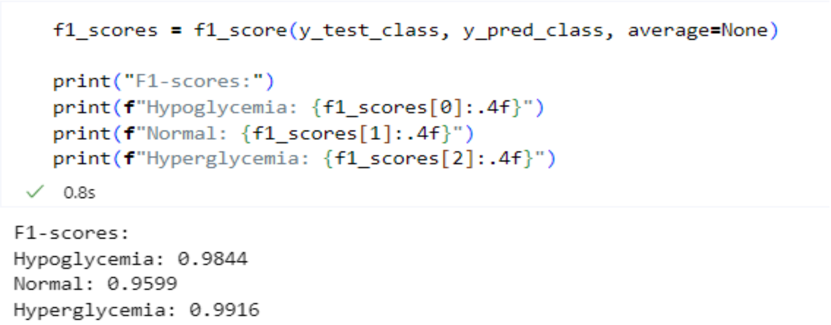
In our second experiment we obtained the following results :

- RMSE = 3.20
- R^2 = 99.6%

Comparison between the two experiences and Observations

- **Training Set RMSE:** In the second experiment, our model achieved a slightly lower training set RMSE (2.48) compared to the first experiment (2.63). This indicates that our model slightly improved its ability to fit the training data with this longer training configuration (100 epochs instead of 50).
- **Testing Set RMSE:** The testing set RMSE remained unchanged between the two experiments (3.20). This suggests that our model maintains a consistent accuracy when predicting test data, regardless of the number of epochs used.
- **R-squared (R2):** The R2 value remains high and identical (0.99) in both experiments, demonstrating that our model continues to effectively capture relationships between features and the target variable (glucose).

Thus, extending the training to 100 epochs in the second experiment did not lead to a significant improvement in testing set RMSE compared to the first experiment with 50 epochs. However, it did slightly enhance performance on the training set. This indicates that our model achieves sufficient convergence with 50 epochs for the task of predicting glucose levels, without requiring a significant increase in the number of epochs.



```
f1_scores = f1_score(y_test_class, y_pred_class, average=None)

print("F1-scores:")
print(f"Hypoglycemia: {f1_scores[0]:.4f}")
print(f"Normal: {f1_scores[1]:.4f}")
print(f"Hyperglycemia: {f1_scores[2]:.4f}")
```

✓ 0.8s

```
F1-scores:
Hypoglycemia: 0.9844
Normal: 0.9599
Hyperglycemia: 0.9916
```

Figure 4.10: CRNN Model Predictions Results ”F1-score” on HUPA-UCM Dataset with 100 Epochs

- The model performed well on all three classes, with F1-scores above 0.95 for each class.

4.4 Real-time Processing in BGL Prediction

In this work, the system is represented through a Convolutional Recurrent Neural Network model that will help it in doing the estimation of future glucose levels based on trained historical glucose information. The model can process sequential data and is therefore appropriate for the time-series prediction, which is necessary in monitoring such dynamic variations in glucose levels throughout the day.

For that, if data is captured and transmitted every 5 minutes, it is essential to manage this data stream efficiently to update our CRNN model regularly and sustainably. Here is a detailed plan for handling real-time predictions with updates every 5 minutes:

Data Capture: Continuous monitoring devices capture glucose levels, heart rate, and other relevant parameters every 5 minutes.

Preprocessing: The incoming data is immediately cleaned and preprocessed. This includes handling missing values, outlier detection, and normalization to ensure consistency with the training data.

Prediction Pipeline: The preprocessed data is fed into the prediction model. Given the temporal nature of the data, a model capable of handling sequences, such as a CRNN, is ideal.

Specify the classification: Based on the predictions, alerts are generated for potential hypo or hyperglycemia events. This feedback is crucial for immediate medical interventions.

Model Retraining: The system periodically retrains the model using a combination of **historical and recent** data to adapt to any changes in the patient's condition or behavior.

4.4.1 Implementation

Data Capture

Using function to simulate data capture every 5 minutes

Temporary Storage

Initialize new_data with existing data

```
new_data = data.copy()
```

In the main loop, **new_data_chunk** represents the newly captured data, which is appended to **new_data**, a DataFrame initially copied from our data. This variable accumulates all data points captured over time. Sorting new_data by timestamp ensures that data is processed in chronological order, which is essential for time-series data analysis and prediction tasks.

```
while True:
    new_data_chunk = capture_new_data()
    new_data = pd.concat([new_data, new_data_chunk], ignore_index=True)
    new_data['time'] = pd.to_datetime(new_data['time'])# time column to
     datetime format
    new_data.sort_values('time', inplace=True)# Sort by timestamp
    ...
```

Real-time Prediction

During each iteration of the main loop, the code prepares **scaled_data**, **scaler_glucose** , and **scaler_features** using the **prepare_data()**function. These variables scale and transform the data to fit the model's input requirements. The model then predicts glucose levels using the most recent sequence of data (sequence), providing real-time predictions **(y_pred)** and inverse transforms them back to original units for interpretation. **y_pred_list** and **y_true_list**.

```python
scaled_data, scaler_glucose, scaler_features = prepare_data(new_data)
for i in range(len(new_data_chunk)):
    if len(new_data) < seq_length:
        continue
    sequence = scaled_data[-seq_length:].reshape(1, seq_length, scaled_data.shape[1])
    y_pred_scaled = model.predict(sequence)
    y_pred = scaler_glucose.inverse_transform(y_pred_scaled.reshape(-1, 1))

    y_pred_list.append(y_pred[0][0])
    y_true_list.append(new_data.iloc[-1]['glucose'])
```

Periodic Model Update

Every **update_interval** iterations, typically set to capture a day's worth of data, the model is retrained with updated **X_train** and **y_train** sequences. This retraining keeps the model uptodate with the latest data trends and variations in glucose levels. The model architecture, including convolutional and LSTM layers, is defined and compiled with an optimizer and loss function. Training occurs on a subset of the data **(X_train, y_train)** split for validation **(X_test, y_test)**. Finally, the trained model is saved for future use **('crnn_model.h5')**.

```python
update_interval = 288  # 24 hours of data captured every 5 minutes

if len(new_data) % update_interval == 0:
    scaled_data, scaler_glucose, scaler_features = prepare_data(new_data)
    X, y = create_sequences(scaled_data, seq_length)
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    #CRNN model
    model = Sequential()
    model.add(Conv1D(filters=64, kernel_size=3, activation='relu', input_shape=(seq_length, scaled_data.shape[1])))
    model.add(MaxPooling1D(pool_size=2))
    model.add(LSTM(50, return_sequences=True))
    model.add(LSTM(50))
    model.add(Dense(1))

    model.compile(optimizer=Adam(learning_rate=0.001), loss='mean_squared_error')
    model.fit(X_train, y_train, epochs=5, batch_size=32, validation_split=0.2)
    model.save('crnn_model.h5')
```

Minute Pause

To simulate real-time data flow, the main loop includes a 5-minute pause **(timeṡleep(300))** between each iteration. This delay mimics the interval between data captures and model updates in our real-time prediction system. It ensures that the system operates at a realistic pace and allows continuous monitoring and adaptation to changes in glucose levels and other physiological parameters.

```
time.sleep(300)
```

4.4.2 Simulation test

To test this program, we need to simulate a data stream. We created a dataset named **HUPA0028P1000ligne.csv**, which contains the first 1000 rows of the dataset from patient **HUPA0028**.

Using our pre-trained CRNN model and the data from `HUPA0028P1000ligne.csv`:

1. **Loading and Preparing Simulated Data**: We progressively read the data from `HUPA0028P1000ligne.csv`, prepare it, and use it to make predictions at each time step.

2. **Using our Model for Real-Time Predictions**: At each time step, we use the model to predict future glucose levels.

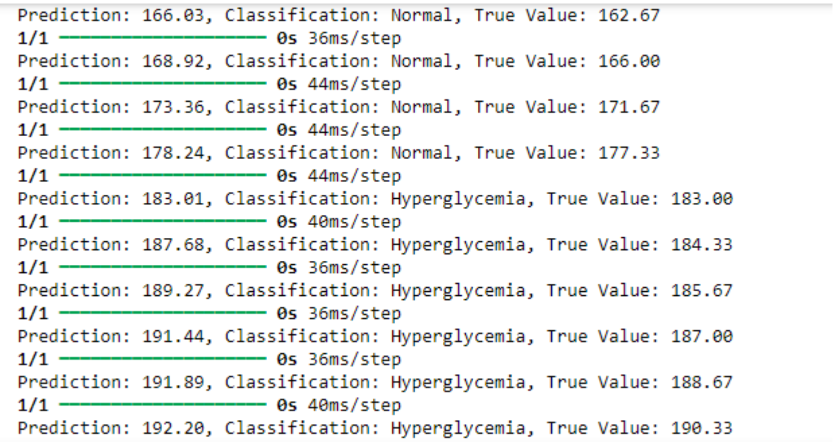
3. **Classifying Predictions**: We classify the predictions into hypoglycemia, hyperglycemia, or normal.

4. **Displaying the real Glucose Value ’True Value’**: We display the true glucose value to compare the classification result with the actual value.

```
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import load_model
import time
from sklearn.metrics import mean_squared_error, mean_absolute_error,
    classification_report

# Load the model
model = load_model('crnn_model.h5')

# Load the data
data = pd.read_csv('HUPA0028P_1000firstlignes.csv', delimiter=';')
# Time in minutes
data['hour'] = data['time'].apply(lambda x: x.split('T')[1])

# Convert time to minutes as float
def hour_to_minutes(hour):
    h, m, s = map(int, hour.split(':'))
    return h * 60 + m + s / 60
```

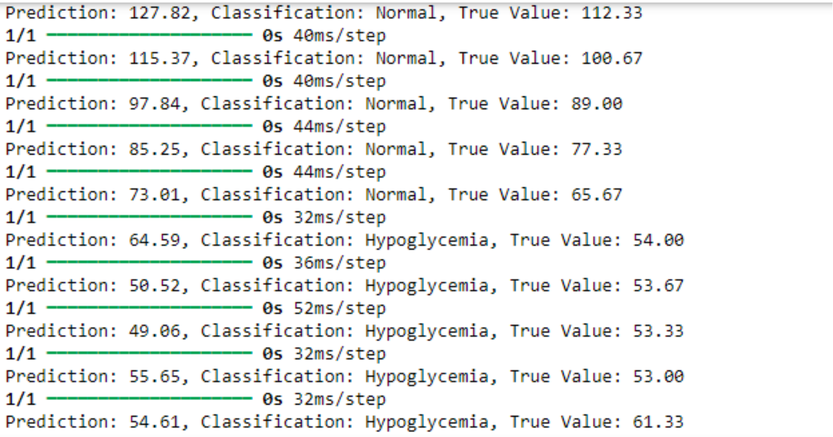
```

data['hour_in_minutes'] = data['hour'].apply(hour_in_minutes)

# Drop the 'time' column
data = data.drop(columns=['time'])
# Separate features and target
features = data.drop(columns=['glucose'])
glucose = data[['glucose']]

# Scale the features
scaler_features = MinMaxScaler()
scaled_features = scaler_features.fit_transform(features)

# Scale the glucose column
scaler_glucose = MinMaxScaler()
scaled_glucose = scaler_glucose.fit_transform(glucose)

# Combine the scaled features and target
scaled_data = np.hstack((scaled_glucose, scaled_features))

# Sequences of 60 time steps
def create_sequences(data, seq_length):
    xs, ys = [], []
    for i in range(len(data) - seq_length):
        x = data[i:i + seq_length]
        y = data[i + seq_length, 0]  # Predict the glucose level
        xs.append(x)
        ys.append(y)
    return np.array(xs), np.array(ys)

seq_length = 60

# Define the glucose level classifier
def classify_glucose_levels(values, hyper_threshold = 180, hypo_threshold = 70):
    classifications = []
    for value in values:
        if value > hyper_threshold:
            classifications.append('Hyperglycemia')
        elif value < hypo_threshold:
            classifications.append('Hypoglycemia')
        else:
            classifications.append('Normal')
    return classifications

# To store predictions and true values
y_pred_list = []
y_true_list = []

# Simulate real-time data stream
for i in range(seq_length, len(scaled_data)):
    # Create a sequence of data
    sequence = scaled_data[i-seq_length:i].reshape(1, seq_length, scaled_data.shape[1])

    # Prediction
    y_pred_scaled = model.predict(sequence)
    y_pred = scaler_glucose.inverse_transform(y_pred_scaled.reshape(-1, 1))

    # Add predictions and true values to the lists
    y_pred_list.append(y_pred[0][0])
    y_true_list.append(glucose.iloc[i, 0])

    # Display prediction result and classification
    y_pred_class = classify_glucose_levels(y_pred)
    print(f"Prediction: {y_pred[0][0]:.2f}, Classification: {y_pred_class[0]}, True Value: {glucose.iloc[i, 0]:.2f}")

    # Alert logic
    if y_pred_class[0] in ['Hyperglycemia', 'Hypoglycemia']:
        print(f"ALERT: {y_pred_class[0]} detected at {pd.Timestamp.now()}")

    # Simulate a 1-second pause for code to execute rapidly
    time.sleep(1)

# Performance metrics
mse = mean_squared_error(y_true_list, y_pred_list)
mae = mean_absolute_error(y_true_list, y_pred_list)

print(f"Mean Squared Error: {mse}")
print(f"Mean Absolute Error: {mae}")

# Classify true glucose levels
y_true_class = classify_glucose_levels(np.array(y_true_list).reshape(-1, 1))

# Classification report
report = classification_report(y_true_class, classify_glucose_levels(np.array(y_pred_list).reshape(-1, 1)))
print(report)
```

After executing this code we obtained the following results:



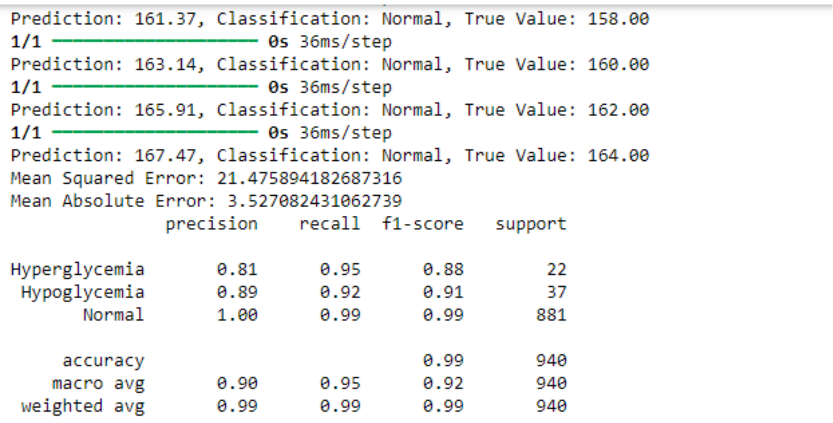
```
Prediction: 166.03, Classification: Normal, True Value: 162.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 168.92, Classification: Normal, True Value: 166.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 44ms/step
Prediction: 173.36, Classification: Normal, True Value: 171.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 44ms/step
Prediction: 178.24, Classification: Normal, True Value: 177.33
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 44ms/step
Prediction: 183.01, Classification: Hyperglycemia, True Value: 183.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 40ms/step
Prediction: 187.68, Classification: Hyperglycemia, True Value: 184.33
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 189.27, Classification: Hyperglycemia, True Value: 185.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 191.44, Classification: Hyperglycemia, True Value: 187.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 191.89, Classification: Hyperglycemia, True Value: 188.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 40ms/step
Prediction: 192.20, Classification: Hyperglycemia, True Value: 190.33
```

Figure 4.11: Hyperglycemia Detection



```
Prediction: 127.82, Classification: Normal, True Value: 112.33
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 40ms/step
Prediction: 115.37, Classification: Normal, True Value: 100.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 40ms/step
Prediction: 97.84, Classification: Normal, True Value: 89.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 44ms/step
Prediction: 85.25, Classification: Normal, True Value: 77.33
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 44ms/step
Prediction: 73.01, Classification: Normal, True Value: 65.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 32ms/step
Prediction: 64.59, Classification: Hypoglycemia, True Value: 54.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 50.52, Classification: Hypoglycemia, True Value: 53.67
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 52ms/step
Prediction: 49.06, Classification: Hypoglycemia, True Value: 53.33
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 32ms/step
Prediction: 55.65, Classification: Hypoglycemia, True Value: 53.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 32ms/step
Prediction: 54.61, Classification: Hypoglycemia, True Value: 61.33
```

Figure 4.12: Hypoglycemia Detection



```
Prediction: 161.37, Classification: Normal, True Value: 158.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 163.14, Classification: Normal, True Value: 160.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 165.91, Classification: Normal, True Value: 162.00
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
Prediction: 167.47, Classification: Normal, True Value: 164.00
Mean Squared Error: 21.475894182687316
Mean Absolute Error: 3.527082431062739
               precision    recall  f1-score   support

Hyperglycemia       0.81      0.95      0.88        22
 Hypoglycemia       0.89      0.92      0.91        37
       Normal       1.00      0.99      0.99       881

     accuracy                           0.99       940
    macro avg       0.90      0.95      0.92       940
 weighted avg       0.99      0.99      0.99       940
```

Figure 4.13: Real Time Results

The model shows very high performance, particularly in classifying normal glucose levels. It also performs very well in identifying hyperglycemic and hypoglycemic states, with high precision, recall, and F1 scores. The overall accuracy of 99% is excellent, indicating that the model is reliable for predicting glucose levels.

In this chapter, we have conducted an extensive review of experiment and results from the existing literature on blood glucose prediction. Additionally, we have implemented a CRNN model architecture on a new dataset that has not previously been tested with CRNN modeling, achieving the highest level of accuracy recorded thus far.

In this study, we adopted a different CRNN model architecture with the aid of a new large dataset in order to compare our results with the recent findings from this paper [27]. Our primary goal was to attain the highest level of accuracy in the early prediction of hypoglycemia and hyperglycemia events, as well as implementing a real time system. The potentials of the hybrid model in using CNN and RNN —especially LSTM—have been able to show robustness in solving deep learning challenges across a very wide spectrum.

To evaluate the performance of our model, we used a dataset with several variables, including: *'time', 'glucose', 'calories', 'heart_rate', 'steps', 'basal_rate','bolus_volume_delivered', 'carb_input'* [16], to make accurate predictions and real time processing for low and high blood sugar episodes concerning T1D patients. We developed a Python program for the real time processing of the CRNN model where the simulated data captured and transmitted every five minutes, as well as a generated feedback classification as a trigger for patient's hypo or hyperglycemia events.

After achieving an excellent RMSE value of 3.20 on our original data, our findings consistently demonstrated that the combined CRNN model outperformed as well on simulated data, resulting RMSE value equals to 4.63 and F1-score of: hypoglycemia = 91%, normal = 99% and hyperglycemia = 88%. The process is a feedback system, so it learns constantly after each 24 hours data generated and the model gets improved continuously.

Based on these highly promising findings, we assert that our method to have further potential applications in various diseases prediction technologies belonging to a wide range of domains. Application of machine learning, specifically deep learning, to medical diagnostics has been becoming crucial in recent times. It can certainly expedite the diagnosis process and patient triage, especially in disease prediction.

In the following general conclusion, we will draw final conclusions based on the project and present our future visions and perspectives.

General Conclusion and Perspectives

Diabetes is one of the most serious global health issues, affecting millions of people worldwide. It crosses the lives of people in several countries, the number of these cases has increased in millions of people gradually in time, which demonstrates that diabetes is now a global disease. This has created an international research initiative with the aim of the collective fight against the disease. Our work is part of an initiative that has been implemented research and one of various means of addressing this worldwide issue. We opened by pointing out some of the primary contributors in the field of diabetes prediction, focusing on blood glucose level prediction.

Our approach involved employing a CRNN (Convolutional Recurrent Neural Network) model on a large, newly available dataset ”HUPA-UCM Diabetes Dataset[16]” that had not been previously used for CRNN modeling. Our target was to spot out the glucose highs and lows with high accuracy, which could lead to better daily management of diabetes. Both the primary CRNN architecture and its real time implementation were developed and carefully tested with different hyperparameters for accurate performance.

The hybrid model with both CNN and LSTM layers had robust capabilities to capture spatial and temporal dependencies in data, attaining the highest accuracy level ever recorded. It used variables like time, glucose, calories, heart rate, steps, basal rate, bolus volume delivered, and carb input. Using these variables, our model could easily make accurate predictions and process real-time data for T1D patients. This was enhanced by a Python program that enabled real-time processing for the patient and thus enabled feedback implementation to learn continuously on the generated data within 24 hours.

Our model demonstrated excellent performance with an RMSE value of 2.48 on the training set, and 3.20 on the test set, significantly outperforming previous studies, in contrast to the [27] in particular. In addition, it achieved an remarquable RMSE value of 4.63 on the simulated data. These results underscore the potential of our CRNN model in improving diabetes management through accurate and timely predictions of glycemic events.

Real-time processing refers to the capability whereby data input is immediately processed into results upon receipt in a system, five minutes in our case. When we use machine learning in our case of glucose-level prediction with real-time processing, it is a constant feed of the information about the patient’s glucose levels to the machine that manages the selection of correct interventions for hypoglycemia and hyperglycemia. The CRNN model is designed to operate on sequential data, enabling more accurate time-series predictions of glucose level variations throughout the day. This is crucial for tracking dynamic fluctuations in glucose levels over time.

This real-time processing capability is essential for managing diabetes effectively because it proactively initiates immediate responses to possibly dangerous variations in blood sugar levels, potentially averting major health consequences. By utilizing deep learning to provide predictive insight and improve patient outcomes, integrating such a system with CGM equipment would mark a substantial improvement in the treatment of diabetes.

In conclusion, our research has demonstrated the effectiveness of using dL methods into medical diagnostics, namely in the area of diabetes event prediction. But rather than being a goal unto itself, this project is a stage in the continuous battle against diabetes. This project may go in the following paths in the future:

- Developing a comprehensive real-time glucose monitoring system for diabetic patients, which integrates a wearable device (like a smartwatch with a glucose sensor), a mobile application, and cloud-based data storage. The wearable device collects glucose data and transmits it to the mobile application, which then uploads the data to secure cloud storage. Our advanced hybrid CRNN model analyzes the data in real-time to detect abnormal glucose levels and send immediate alerts to the person through the mobile application. This system ensures continuous monitoring, timely intervention, personalized health recommendations, and improved patient outcomes.

- Creating a database of patient glucose level information in Algeria to facilitate national diabetes research and care.

Through a combination of modern technology and machine learning innovations our aim is to make diabetes treatment more efficient and less health taxing for individuals suffering with the condition. Building on these findings, we want to pursue more extensive applications and all-encompassing solutions in medical diagnostics realm in our future initiatives.

References

[1] Diabetes mellitus type 1 pancreass failure. https://www.shutterstock.com/image-illustration/diabetes-mellitus-type-1-pancreass-failure-2216776955.

[2] Difference between bladder and gallbladder — compare the difference between similar terms.

[3] Glucose levels in the blood diagram hyperglycemia hypoglycemia diabetes. https://depositphotos.com/vector/glucose-levels-in-the-blood-diagram-hyperglycemia-hypoglycemia-diabetes-215510166.html.

[4] Kaggle: Your home for data science. https://www.kaggle.com, Accessed: 2024-06-15.

[5] Learn more about diabetes — ohsu. https://www.ohsu.edu/schnitzer-diabetes-center/glossary/.

[6] Numpy. https://numpy.org/, Accessed: 2024-04-20.

[7] Python. https://www.python.org/, Accessed: 2024-04-17.

[8] Tensorflow. https://www.tensorflow.org/, Accessed: 2024-06-15.

[9] Admin. Difference between bladder and gallbladder — compare the difference between similar terms. *Difference Between* (2013).

[10] Alfian, G., Syafrudin, M., Ijaz, M. F., Syaekhoni, M. A., Fitriyani, N. L., and Rhee, J. A personalized healthcare monitoring system for diabetic patients by utilizing ble-based sensors and real-time data processing. *Sensors 18*, 7 (2018), 2183.

[11] Bliss, M. *The Discovery of Insulin*. University of Chicago Press, 1982.

[12] Dave, D., DeSalvo, D. J., Haridas, B., McKay, S., Shenoy, A., Koh, C. J., Lawley, M., and Erraguntla, M. Feature-based machine learning model for real-time hypoglycemia prediction. *Journal of Diabetes Science and Technology 15*, 4 (2021), 842–855.

[13] Duckworth, C., Guy, M. J., Kumaran, A., O'Kane, A. A., Ayobi, A., Chapman, A., Marshall, P., and Boniface, M. Explainable machine learning for real-time hypoglycemia and hyperglycemia prediction and personalized control recommendations. *Journal of Diabetes Science and Technology 18*, 1 (2024), 113–123.

[14] Dudukcu, H. V., Taskiran, M., and Yildirim, T. Blood glucose prediction with deep neural networks using weighted decision level fusion. *Biocybernetics and Biomedical Engineering 41*, 3 (2021), 1208–1223.

[15] Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems 28*, 10 (Oct 2017), 2222–2232.

[16] Hidalgo, J. Ignacio; Alvarado, Jorge; Botella, Marta; Aramendi, Aranzazu; Velasco, J. Manuel; Garnica, Oscar. HUPA-UCM Diabetes Dataset, 2024. https://data.mendeley.com/datasets/3hbcscwz44/1.

[17] Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural Computation 9*, 8 (1997), 1735–1780.

[18] Hunter, J. D., Droettboom, M., Casillas, F., et al. Matplotlib: Plotting with python, 2007. https://matplotlib.org, Accessed: 2024-04-20.

[19] Iacono, F., Magni, L., and Toffanin, C. Personalized lstm-based alarm systems for hypoglycemia and hyperglycemia prevention. *Biomedical Signal Processing and Control 86* (2023), 105167.

[20] International Diabetes Federation. About diabetes, 2023. https://www.idf.org/aboutdiabetes/what-is-diabetes/, Accessed: 2024-04-10.

[21] Kahn, M. Diabetes. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5T59G.

[22] Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., and Chouvarda, I. Machine learning and data mining methods in diabetes research. *Computational and structural biotechnology journal 15* (2017), 104–116.

[23] Kodama, S., Fujihara, K., Shiozaki, H., Horikawa, C., Yamada, M. H., Sato, T., Yaguchi, Y., Yamamoto, M., Kitazawa, M., Iwanaga, M., et al. Ability of current machine learning algorithms to predict and detect hypoglycemia in patients with diabetes mellitus: meta-analysis. *JMIR diabetes 6*, 1 (2021), e22458.

[24] Kurenkov, A. A brief history of neural nets and deep learning. *Skynet Today* (2020).

[25] LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature 521*, 7553 (2015), 436–444.

[26] Li, G., Wang, H., Zhang, S., Xin, J., and Liu, H. Recurrent neural networks based photovoltaic power forecasting approach. *Energies 12*, 13 (2019), 2538.

[27] Li, K., Daniels, J., Liu, C., Herrero, P., and Georgiou, P. Convolutional recurrent neural networks for glucose prediction. *IEEE journal of biomedical and health informatics 24*, 2 (2019), 603–613.

[28] Mahesh, B. Machine learning algorithms - a review. *International Journal of Science and Research (IJSR) 9*, 1 (January 2020). ResearchGate Impact Factor (2018): 0.28 — SJIF (2018): 7.426. Licensed Under Creative Commons Attribution CC BY.

[29] Makroum, M. A., Adda, M., Bouzouane, A., and Ibrahim, H. Machine learning and smart devices for diabetes management: Systematic review. *Sensors (Basel) 22*, 5 (2022), 1843. https://www.mdpi.com/1424-8220/22/5/1843.

[30] Mansour, A., Amroun, K., and Habbas, Z. Artificial intelligence and its application in insulin bolus calculators. In *Advances in Multidisciplinary Medical Technologies Engineering, Modeling and Findings: Proceedings of the ICHSMT 2019* (2021), Springer, pp. 55–76.

[31] Marling, C., and Bunescu, R. The ohiot1dm dataset for blood glucose level prediction: Update 2020. In *CEUR workshop proceedings* (2020), vol. 2675, NIH Public Access, p. 71.

[32] Mayo, M., Chepulis, L., and Paul, R. G. Glycemic-aware metrics and oversampling techniques for predicting blood glucose levels using machine learning. *PloS one 14*, 12 (2019), e0225613.

[33] McKinney, W. *Python for Data Analysis: Second Edition*. O'Reilly Media, Inc., 2017.

[34] Mhaskar, H. N., Pereverzyev, S. V., and van der Walt, M. D. A deep learning approach to diabetic blood glucose prediction. *Frontiers in Applied Mathematics and Statistics 3*, July (2017), 1–11.

[35] Mohammed, Nassim, B. H., Yassamina, Tabet, Z., Meriem, S., Zoubeyr, A., Amina, Y., Houssam, B., Hakima, F., Khawla, K., Nawel, B.-N., Ali, L., and Kaouel, M. Factors associated with non-adherence to insulin in type 1 and type 2 diabetes mellitus patients in western region of algeria, tlemcen: a cross-sectional study. *Pan African Medical Journal 41*, 172 (03 2022).

[36] of Disease Collaborative Network, G. B. Global burden of disease study 2019. results, 2020. https://vizhub.healthdata.org/gbd-results/.

[37] Olusanya, M. O., Ogunsakin, R. E., Ghai, M., and Adeleke, M. A. Accuracy of machine learning classification models for the prediction of type 2 diabetes mellitus: A systematic survey and meta-analysis approach. *International Journal of Environmental Research and Public Health 19*, 21 (2022), 14280.

[38] Ong, K. L., Stafford, L. K., McLaughlin, S. A., Boyko, E. J., Vollset, S. E., Smith, A. E., Dalton, B. E., Duprey, J., Cruz, J. A., Hagins, H., Lindstedt, P. A., Aali, A., Abate, Y. H., Abate, M. D., and Collaborators, G. . D. Global, regional, and national burden of diabetes from 1990 to 2021, with projections of prevalence to 2050: a systematic analysis for the global burden of disease study 2021. *The Lancet 402*, 10397 (2023), 203–234.

[39] Organization, W. H. Diabetes, 2023. https://www.who.int/news-room/fact-sheets/detail/diabetes.

[40] pandas development team, T. pandas: Python data analysis library. https://pandas.pydata.org/, Accessed: 2024-04-20.

[41] Prez, F., Granger, B. E., Zafar, A., et al. Jupyter notebook, 2003. https://jupyter.org/, Accessed: 2024-04-30.

[42] Rabby, M. F., Tu, Y., Hossen, M. I., Lee, I., Maida, A. S., and Hei, X. Stacked lstm based deep recurrent neural network with kalman smoothing for blood glucose prediction. *BMC Medical Informatics and Decision Making 21* (2021), 1–15.

[43] Rahman, M. M., Shakeri, M., Tiong, S. K., Khatun, F., Amin, N., Pasupuleti, J., and Hasan, M. K. Prospective methodologies in hybrid renewable energy systems for energy prediction using artificial neural networks. *Sustainability 13*, 4 (2021), 2393.

[44] Rghioui, A., Lloret, J., Harane, M., and Oumnad, A. A smart glucose monitoring system for diabetic patient. *Electronics 9*, 4 (2020), 678.

[45] Ruan, Y., Bellot, A., Moysova, Z., Tan, G. D., Lumb, A., Davies, J., Van Der Schaar, M., and Rea, R. Predicting the risk of inpatient hypoglycemia with machine learning using electronic health records. *Diabetes care 43*, 7 (2020), 1504–1511.

[46] scikit-learn developers. scikit-learn: machine learning in python.

[47] Sherkhane, P. D., Ratnaparkhi, N. S., and Rudrawar, S. S. Machine learning concept, algorithms and applications.

[48] Shi, M., Yang, A., Lau, E. S., Luk, A. O., Ma, R. C., Kong, A. P., Wong, R. S., Chan, J. C., Chan, J. C., and Chow, E. A novel electronic health record-based, machine-learning model to predict severe hypoglycemia leading to hospitalizations in older adults with diabetes: A territory-wide cohort and modeling study. *PLoS medicine 21*, 4 (2024), e1004369.

[49] Shuvo, M. M. H., and Islam, S. K. Deep multitask learning by stacked long short-term memory for predicting personalized blood glucose concentration. *IEEE Journal of Biomedical and Health Informatics 27*, 3 (2023), 1612–1623.

[50] Skiadopoulos, D. *Diabetes mellitus*. PhD thesis, Charles University, 2013. https://urlz.fr/rbht.

[51] Sun, Q., Jankovic, M. V., Bally, L., and Mougiakakou, S. G. Predicting blood glucose with an lstm and bi-lstm based deep neural network. In *Symposium on Neural Networks and Applications (NEUREL)* (Belgrade, 2018), pp. 1–5.

[52] Team, K. Keras: Deep learning for humans. https://keras.io/, Accessed: 2024-06-15.

[53] Tena, F., Garnica, O., Lanchares, J., and Hidalgo, J. I. Ensemble models of cutting-edge deep neural networks for blood glucose prediction in patients with diabetes. *Sensors 21*, 21 (2021), 7090.

[54] Wang, G., Liu, X., Ying, Z., Yang, G., Chen, Z., Liu, Z., Zhang, M., Yan, H., Lu, Y., Gao, Y., et al. Optimized glycemic control of type 2 diabetes with reinforcement learning: a proof-of-concept trial. *Nature Medicine 29*, 10 (2023), 2633–2642.

[55] Wang, Y., Zhou, J., Chen, K., Wang, Y., and Liu, L. Water quality prediction method based on lstm neural network. In *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)* (Nanjing, China, 2017), IEEE, pp. 1–5.

[56] Waskom, M. *seaborn: statistical data visualization*, 2021. https://seaborn.pydata.org/, Accessed: 2024-04-20.

[57] Witte, H., Nakas, C., Bally, L., Leichtle, A. B., et al. Machine learning prediction of hypoglycemia and hyperglycemia from electronic health records: algorithm development and validation. *JMIR formative research 6*, 7 (2022), e36176.

[58] Yang, C., Zhai, J., and Tao, G. Deep learning for price movement prediction using convolutional neural network and long short-term memory. *Future Internet 12*, 7 (2020), 121.

[59] Yang, T., Yu, X., Ma, N., Wu, R., and Li, H. An autonomous channel deep learning framework for blood glucose prediction. *Applied Soft Computing 120* (2022), 108636.