EVOLUTION IN ELECTRICAL AND ELECTRONIC ENGINEERING
e-ISSN: 2756-8458

EEEE

Vol. 5 No. 1 (2024) 276-285
https://publisher.uthm.edu.my/periodicals/index.php/eeee

Univariate and Multivariate Time Series Blood Glucose Prediction with LSTM Deep Learning Model

Muhammad Shahmi Mohamed Nordin[1], Farhanahani Mahmud[2]*

[1] *Department of Electronic Engineering, Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Batu Pahat, 86400, MALAYSIA*

[2] *Microelectronics and Nanotechnology Shamsuddin Research Centre (MiNT-SRC), Universiti Tun Hussien Onn Malaysia, Batu Pahat, 86400, MALAYSIA*

*Corresponding Author: farhanah@uthm.edu.my
DOI: https://doi.org/10.30880/eeee.2024.05.01.035

Article Info

Received: 15 January 2024
Accepted: 28 February 2024
Available online: 30 April 2024

Keywords

LSTM, Type-1 diabetes, Blood glucose, Cobelli type-1 Diabetic Model, Insulin, Deep Learning

Abstract

Type-1 diabetes is a chronic autoimmune condition impacting insulin production, a crucial hormone for regulating blood sugar levels. Caused by the immune system mistakenly attacking insulin-producing beta cells in the pancreas, this leads to insulin deficiency and elevated blood sugar levels. Globally, approximately 9 million people, representing 0.1% of the population, grapple with type-1 diabetes. With technological advancements, a need arises for the blood glucose prediction model to help monitor and manage type-1 diabetes patients. Deep learning, particularly Long Short-Term Memory (LSTM) networks, proves its ability to grasp long-term dependencies in sequential data however, creating an accurate model for a time-series blood glucose prediction is a complex challenge that requires further research and exploration in the modeling. Thus, leveraging the Cobelli model to simulate blood glucose data in type-1 diabetes patients, the primary goal is to utilize an LSTM network for better prediction of glucose levels. Ten datasets containing information on blood glucose levels, insulin, and meal intake are employed to train both univariate and multivariate models. The univariate model relies solely on glucose data, while the multivariate model integrates insulin and meal intake variables. Two prediction horizons (5- and 10-minute) are utilized to assess and compare model performance. Performance evaluation includes regression analysis metrics of Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and MAE. From the result, it is found that the multivariate model has shown a better prediction performance compared to the univariate model, with the best mean error scores of 0.8777 [mg/dl] for the MAE, 0.958024 [mg/dl] for the RMSE and 1.9875 [mg/dl] for the MSE with the 5-minute prediction horizon outperformed the 10-minute prediction horizon. Based on the findings, a better understanding of designing a high-performance LSTM deep learning model for blood glucose prediction has been achieved, which could promote better diabetes treatment.

1. Introduction

With approximately 34.7 million diagnosed cases worldwide [1], diabetes disrupts glucose homeostasis, leading to chronic conditions. Monitoring blood sugar is vital for identifying metabolic disorders, assessing treatment efficacy, and preventing long-term complications [1]. Typically arising from insufficient insulin production or utilization, diabetes manifests in various types, with type-1 and type-2 being the most common. Type-1 diabetes, starting early in life, results from pancreatic cells failing to produce insulin. Symptoms include increased thirst, weight loss, fatigue, and frequent infections, requiring lifelong insulin therapy, diet management, and regular exercise [2]. Type-2 diabetes, often emerging in middle or old age, stems from insulin resistance or inadequate insulin production. Genetic and lifestyle factors like obesity and poor diet contribute to its development, sharing symptoms with type-1 diabetes [2].

Currently, diabetes cannot be fully cured but can be stabilized using drugs or insulin. Predicting blood glucose levels is crucial for timely and effective treatment. Continuous Glucose Monitoring (CGM) is a method to manage blood glucose levels in type-1 diabetics. This involves wearing a device that periodically checks glucose levels in interstitial fluid, providing a real-time record with measurements every 5 minutes, totalling 288 per day [3].

Continuous monitoring enhances treatment assessment and enables anticipation of future blood glucose levels [3]. In the era of big data, where vast volumes of data are generated in every facet of science and industry [4], innovative machine learning and artificial intelligence approaches are crucial for data analysis and interpretation [4]. Deep learning (DL), a distinct subset of Machine Learning (ML), is gaining considerable focus, characterized by neural networks with three or more layers [5]. DL's key attributes include its adaptability to features and autonomous learning from data [5]. Both ML and DL algorithms offer benefits and find applications in the diagnosis and prediction of diseases [5].

The deep learning process typically involves several stages. Initially, high-throughput data with numerous features is introduced into the learning process [5]. Subsequently, the data undergoes preprocessing to eliminate outliers and reduce dimensionality by excluding disjointed data or identifying relevant information [5]. Algorithms are then tailored to align with the study's objectives [5]. The model is tested on external data, and its performance is evaluated using various metrics such as the Receiver Operating Characteristic Curve (ROC), Area Under Curve (AUC), Mean Absolute Relative Difference (MARD), Root-Mean-Square Error (RMSE), Mean Squared Error (MSE), accuracy, precision, recall, log loss, among others [5].

Various deep learning (DL) methods are utilized for predicting blood glucose levels, with algorithms applied to control, classify, predict, and manage diabetes. Long Short-Term Memory (LSTM), a type of Recurrent Neural Network (RNN), is employed for data-driven blood glucose patterns and diabetes and hypoglycemia prediction [5]. RNN models, including LSTM, handle time-dependent sequence data by incorporating previous state information through looping in a chain structure [5]. LSTM introduces memory cells, enhancing prediction feasibility by integrating memories with inputs and allowing the network to learn order dependence [5]. The forget gate in LSTM determines which information from the old state should be retained [3]. LSTM-based networks, known for faster learning capabilities, have shown promising results in time series prediction and understanding physiological models of blood glucose behavior [3]. In this research, an LSTM-based deep neural network is employed to predict blood glucose levels based on multivariate data.

2. Problem Statement

Fig. 1 shows the general steps of the research methodology. The work begins with a comprehensive study on the Cobelli type 1 diabetic model and various deep-learning techniques. Following this, a MATLAB program analyzes and studies Cobelli model parameters, generating a blood glucose dynamics dataset. This dataset forms the basis for a simulation system designed using Python-based software with an LSTM prediction model for blood glucose level predictions. Data preparation involves processes like splitting and windowing to fit the LSTM model. The subsequent steps include univariate and multivariate modeling of the LSTM network by determining the architecture. Achieving an effective model involves considering critical parameters, including the number of layers such as LSTM and hidden layers. The depth of the LSTM network is determined by the number of layers, with more layers needed for complex tasks to capture patterns and trends. The hidden layer specifies the number of memory cells in each LSTM layer, enabling the network to learn complex datasets at the cost of increased computational requirements. According to Karsten Eckhardt et al., there are no definite ways to calculate how much layer is needed for a LSTM network to choose. Very often a trial and error will give the best result for each individual model [6]. Additional parameters like sequence length, input shape, batch size, and optimizer are crucial for determining the LSTM architecture, ensuring the model has the capacity to effectively learn and represent patterns. The trained LSTM model predicts blood glucose values, and performance analysis involves evaluating error metrics such as mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

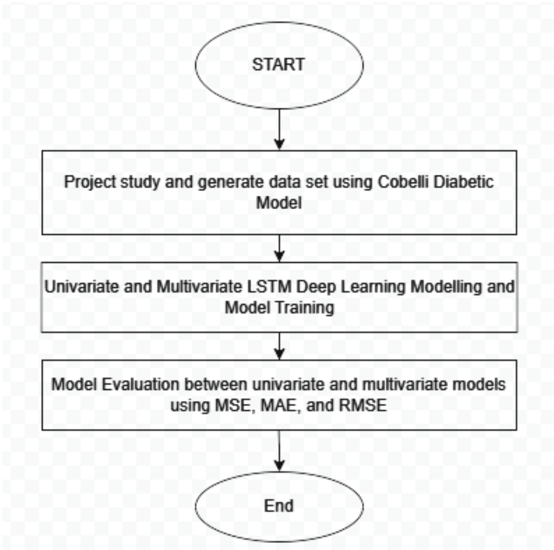


Fig. 1 *General steps of the research methodology*

2.1 The dataset Generation using Simulink Model of Open-Loop Delivery System in Cobelli Virtual Type-1 Diabetic Model

The Cobelli model was chosen over competitors like the Hovorka model for investigating type I diabetes due to its comprehensive physiological foundation, encompassing variables affecting insulin secretion, glucose absorption, and utilization. This model demonstrates improved accuracy in modeling blood glucose levels, providing a realistic response to varied inputs. Its endorsement by a recognized regulatory organization adds confidence for researchers and physicians. The model's compatibility with MATLAB simplifies its application, allowing effective simulation of blood glucose dynamics. Fig. 2 shows MATLAB Simulink model, spanning 24 hours, employs various blocks, including a diabetes block created using the S-function block to describe mathematical equations of the Cobelli model. This open-loop setup facilitates the study of the model's behavior. The Signal Builder block designs the meal disturbance and insulin inputs in the Cobelli model dataset simulation. Insulin and meal input are represented in 5-minute step input boluses for a body weight of 76.37 kg. Additional parameters include Glucose Set Point, Glucose Hyperglycemia, and Glucose Hypoglycemia. Fig. 2 illustrates the open-loop system for insulin input with five inputs for breakfast, lunch, evening snack, dinner, and night snack. Carbohydrate intake values accompany each meal, causing blood glucose levels to peak 30 minutes after consumption. The insulin input signal block regulates insulin supply, restoring blood glucose levels to normal.

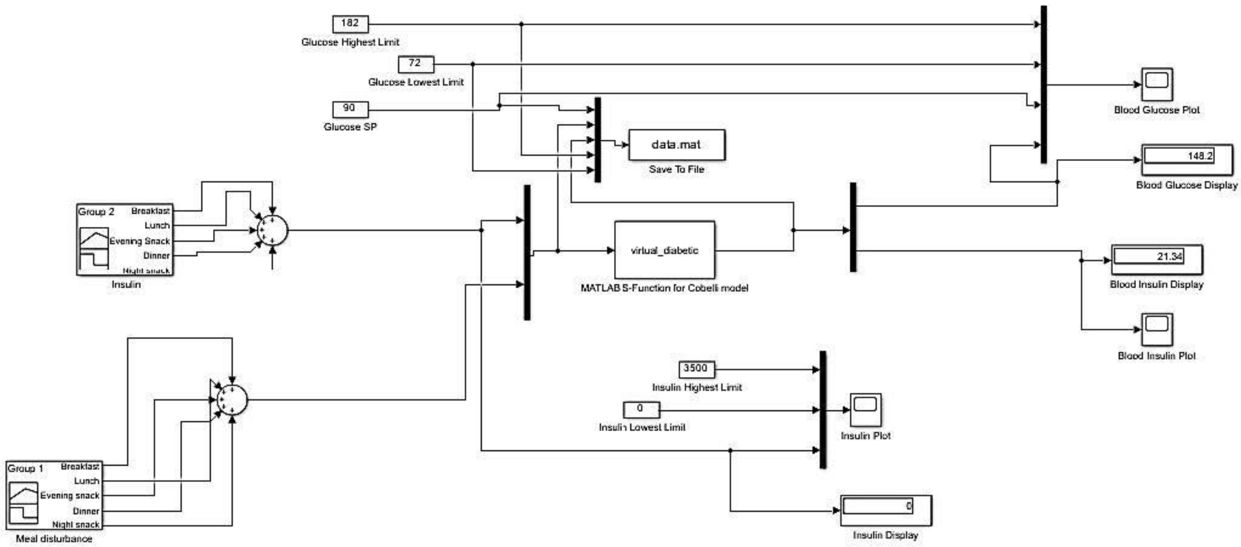


Fig. 2 *Open-Loop insulin delivery system for MATLAB Simulink design*

2.2 LSTM Deep Learning Design

One of the critical steps in ensuring that the research can proceed as smoothly as possible is designing the experiments. Fig. 3 shows the flowchart of the LSTM deep learning model design.

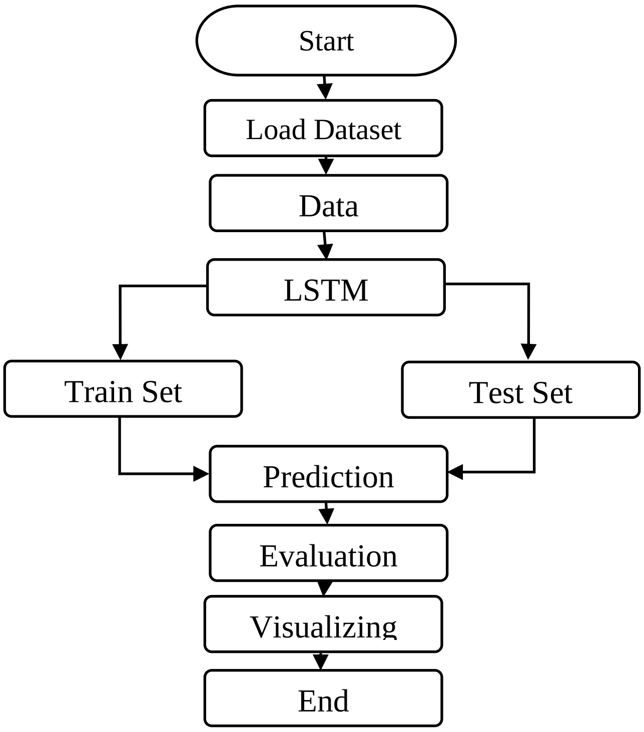


Fig. 3 *LSTM Design deep learning model*

Ten datasets generated from the Cobelli model were imported into Google Colab, a cloud-based tool. These datasets are stored in Google Drive and mounted to Colab for easy access. Data preprocessing using 'MinMaxScaler' is crucial for deep learning, normalizing input features to enhance model reliability. The 'create_multivariate_dataset' code segment transforms time series data into a multivariate format using a sliding window approach for input-output pairs. 'Lookback' determines past steps, and 'horizon' sets future steps for prediction. Data splitting into training and test sets is essential for optimizing and validating deep learning models, providing an unbiased estimate of generalization on new data.

The LSTM is a type of recurrent neural network (RNN), effectively captures long-term dependencies in sequential data. Its architecture facilitates the flow of information through memory cells and gates, making it adept at learning patterns in sequential data. Stacking multiple LSTM layers is common to increase model capacity and complexity. The 'MultivariateLSTM' model is created using information as in Table 1 where the input size of 3 (features: 'Glucose', 'Insulin', 'Meal') and one LSTM layer with 50 neurons. Hyperparameters like the number of layers and output size are tuned. Mean Squared Error (MSE) serves as the loss function, measuring the squared difference between predictions and actual targets. PyTorch's 'DataLoader' handles batching for training data, improving memory efficiency. This code segment establishes the crucial steps for setting up the LSTM architecture, vital for training the deep learning model.

Table 1 *Hyperparameter of LSTM Model*

Categories	Hyperparameter
Optimizer	Optim.Adam
Number of input size	3
Number of LSTM layers	1
Number of hidden layers	50
Number of output size	Horizon

Training the dataset involves adjusting model parameters to minimize the difference between predictions and actual values. A list is employed to store Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) for each input ('Glucose', 'Insulin', 'Meal'). The 'epoch' is set to 4, indicating the model iterates through the data four times. The epoch setting undergoes trial and error to find the optimal number, preventing overfitting.Overfitting happens when the number of epochs used to train a neural network model is more than necessary, the training model learns the patterns that are specific to a great extent and making the model incapable of performing well on new dataset [7]. The model continues to learn and refine its parameters within the epochs.

The training process iteratively adjusts parameters to minimize the difference between predictions and actual values, enhancing the model's accuracy on unseen data. 'model.train()' initiates training mode, and 'append' stores actual and predicted values in a list for evaluation and loss calculation. 'loss = loss_fn(y_pred, y_batch)' computes the loss between predicted and actual values. Two training methods are employed to compare accuracy: univariate and multivariate. Univariate uses only 'Glucose' as input, predicting its future value. Multivariate utilizes 'Glucose', 'Insulin', and 'Meal' as inputs, predicting 'Glucose' values. Different time horizons (5 and 10 minutes) are set for each model to assess performance. Evaluation metrics (RMSE, MAE, MSE) will gauge the model's accuracy.

The LSTM model using Python Google Collab was employed to predict the blood glucose levels. Fig. 4 illustrates the variables for the model input such as Glucose, Insulin and Meal from Dataset 1. The table gives better visualisations for all datasets. The graph explains the input used in the LSTM model. On the top left was the time input in blue which on the y-axis represents the time in second for total of 1440 second consisting 24 hours of period and the x-axis is the representation of the data step which consists of 28802 numbers of data. For every data there is interval of 0.05 second, means that 0.05 interval x 28802 data = 1440 second (24 hours). The top right of the graph illustrates the value of glucose in red and the bottom left of the graph represents the value of insulin in green while at the bottom right of the graph represents the meal in purple taken by the patient. As the whole, the graphs show the changes in blood glucose in milligram per deciliter (mg/dL).

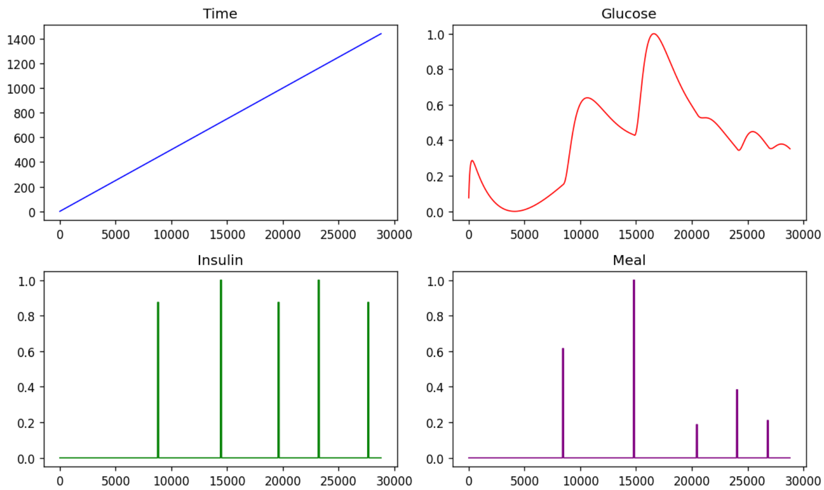


Fig. 4 *Simulation result of normal range blood glucose level.*

3. Result and discussion

The simulation results of the Cobelli model, a 24-hour open-loop glucose-insulin regulation system for type-1 diabetes management, are discussed. Ten datasets, simulating hyperglycemia and normal blood glucose levels, are included, illustrating the model's output for blood insulin, glucose levels, and insulin administration. The subsequent section delves into the results of deep learning Long Short-Term Memory (LSTM) models. The impact of epochs on model training is analyzed, and model performance for univariate and multivariate cases is evaluated. Comparison is made for prediction horizons of 5 minutes and 10 minutes using various metrics such as MSE, RMSE, and MAE. The effectiveness of LSTM models, especially the multivariate version, in predicting blood glucose levels is highlighted by the findings. Due to page limitations, only the waveform results from Dataset 1 are shown in this paper.

3.1 Cobelli Model Simulation Results

The Cobelli model is a 24-hour simulation of an open-loop glucose-insulin regulation system, depicting the management of type-1 diabetes patients with insulin therapy. The model considers five meals with varying carbohydrate values and consumption rates, yielding results for blood insulin, glucose levels, and insulin administration. Ten datasets were created, five for hyperglycemia and five for normal blood glucose levels. Fig. 5 illustrates the simulation of normal blood glucose levels, with the red line indicating the hyperglycemia threshold. When the green line surpasses it, hyperglycemia occurs, signifying insufficient insulin. The black line represents the set point value, and the blue line signifies the threshold for hypoglycemia, indicating blood glucose below the standard range.

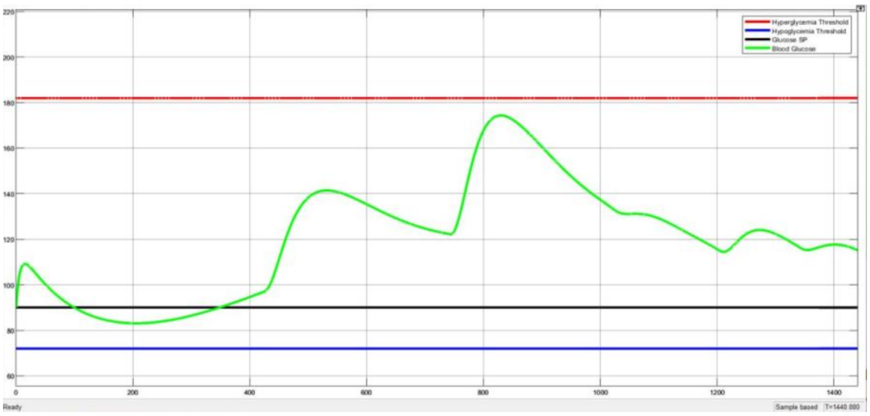
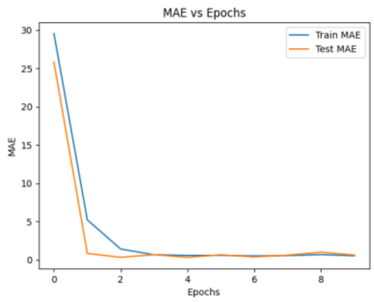
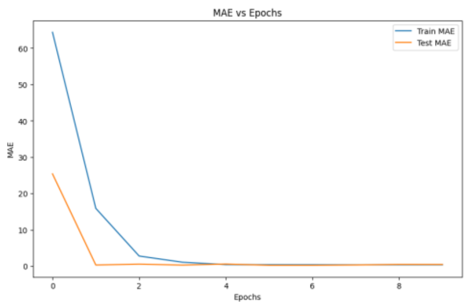
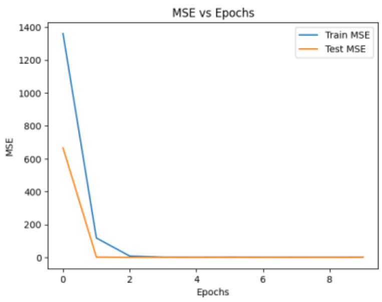
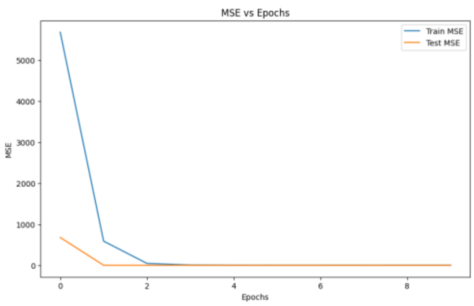
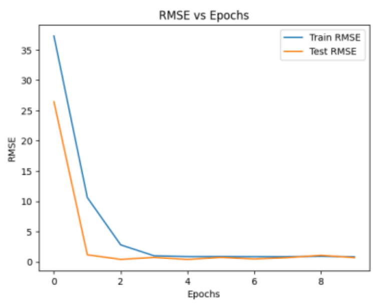
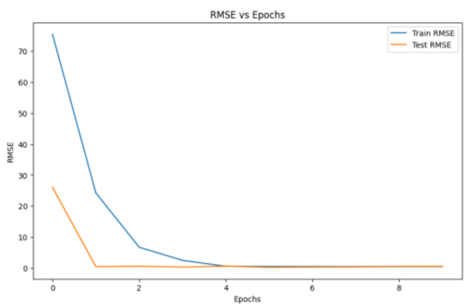


Fig. 5 *Simulation result of normal range blood glucose level.*

Table 2 *Graphs of epoch analysis for univariate and multivariate LSTM model to find the best-fitted epoch for each of the models at 5-minute prediction horizon*

	Univariate	Multivariate
MAE		
MSE		
RMSE		

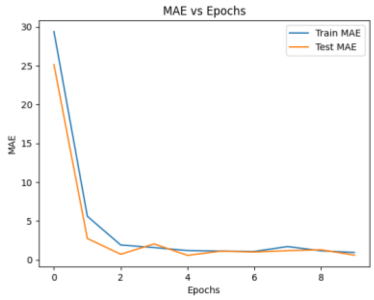
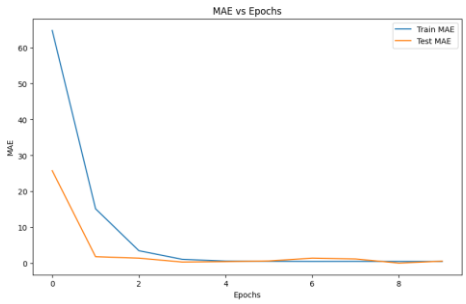
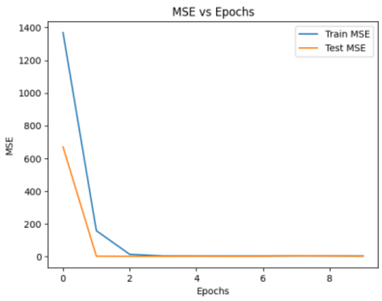
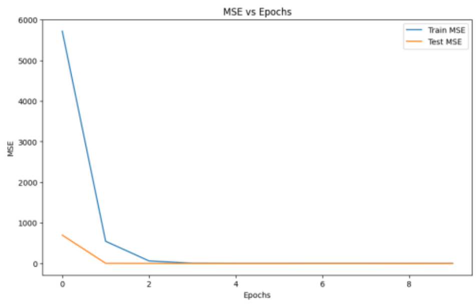
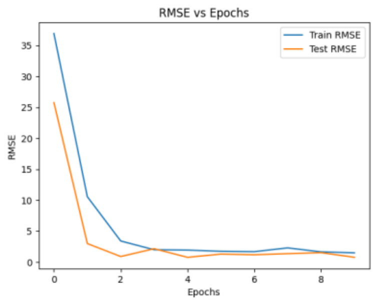
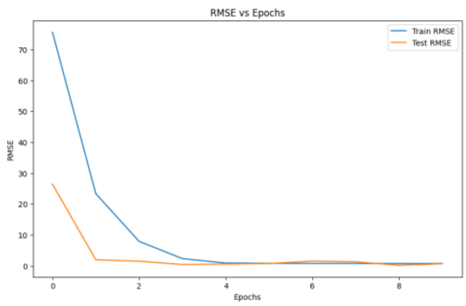
3.2 Deep Learning LSTM Results

3.2.1 Epoch analysis of the univariate and multivariate LSTM models

The evaluation metrics MSE, RMSE, and MAE were employed to analyse the impact of epochs on the LSTM model's training and testing, aiding in determining the optimal epoch for predicting blood glucose levels. This analysis is crucial for identifying potential underfitting or overfitting issues. Two sets of models were considered: univariate (based on glucose data only) and multivariate (based on glucose, insulin, and meal data) for 5-minute and 10-minute prediction horizons.

For both univariate and multivariate models, the number of epochs was determined through epochs analysis by setting the epoch from 0 to 9. Table 2 and Table 3 present graphs of MSE, RMSE, and MAE vs. the number of epochs for the univariate at 5-minute and 10-minute prediction horizons, respectively. According to Table 2 and 3 for univariate and multivariate models respectively, RMSE, MSE, and MAE decrease across epochs for both training and test sets, indicating continuous learning and improvement of the model over time and approximately from the epoch of 2 in both cases of 5-minute and 10-minute prediction horizon, the error scores started to reach a plateau. Thus, based on this analysis, the prediction of blood glucose levels for the univariate and multivariate LSTM models were conducted at epoch of 3.

Table 3 *Graphs of epoch analysis for univariate and multivariate LSTM model to find the best-fitted epoch for each of the models at 10-minute prediction horizon*

	Univariate	**Multivariate**
MAE		
MSE		
RMSE		

3.2.2 Performance evaluation of the univariate and multivariate LSTM models

This section provides the result for the prediction of blood glucose univariate and multivariate LSTM model for 5-minute and 10-minutes prediction horizons. Univariate means that the model only takes Glucose as it input and predicts the values of the future value of the glucose. The multivariate aspect indicates that the model takes Glucose, Insulin, and Meal as inputs to predict future glucose values. Two different time horizons which are 5-minutes and 10-minutes was applied to assess the performance between two sets of the horizon at epoch of 3.

The result was then plotted to provide a better understanding of the result. Table 4 shows the result of both predictions for datasets with 10-minute and 5-minute predictions horizon for the univariate LSTM model and Table 5 details the predictions for both 10-minute and 5-minute horizons for the multivariate LSTM model. The graph shows the predicted value (red line) plotted against the actual glucose (blue line) for both the training and test datasets. The model generally captures the overall trend and the actual glucose value especially for the 5-minute horizon, but there are some notable disparities. For the 10-minute time horizon, the predicted value still follows the trend with good accuracy but there is larger disparity than the 5-minute horizon. This is likely because it is more challenging for the model to predict further into the future. indicating that the model had effectively captured the pattern and trend of the actual value.

Table 4 *Prediction of blood glucose univariate LSTM model for 5 and 10-minute prediction horizons*

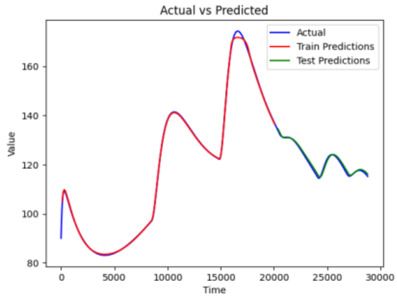
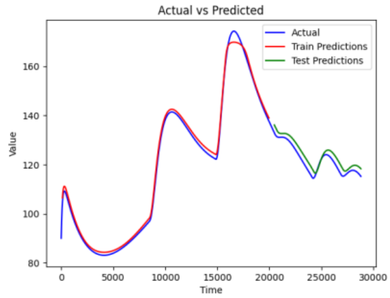
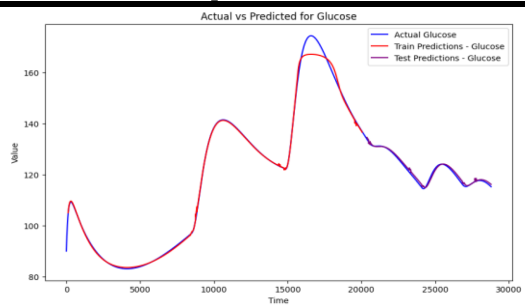
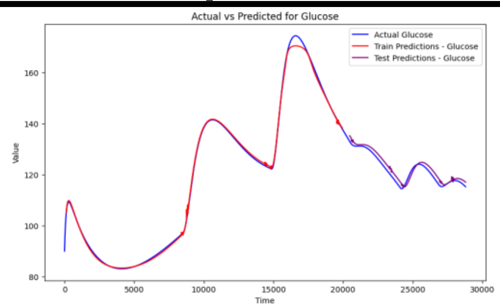
Dataset	5-minute prediction horizon	10-minute prediction horizon
1		

Table 5 *Prediction of blood glucose multivariate LSTM model for 5 and 10-minute prediction horizons*

Dataset	5-minute prediction horizon	10-minute prediction horizon
1		

3.2.3 Performance evaluation of the univariate and multivariate LSTM models

Evaluating the performance of LSTM models for the glucose prediction is crucial to ensure the accuracy and generalisability in real-situation applications. The metrics of MSE, RMSE and MAE were also used to provide valuable insight into how effectively the model predicts unseen data. Lower values of the metrics such as MSE, RMSE and MAE indicate better alignment between predicted and actual glucose values, implying a better model accuracy. MAE quantifies the average absolute disparity between the predicted and actual blood glucose values. MSE calculates the average squared difference between the predicted and actual blood glucose values. RMSE is the square root of MSE, which offers a more understandable metric in the same units as the predicted values.

Table 6 *Values of MAE, MSE, and RMSE for univariate with 5-minute prediction*

Dataset	MAE		MSE		RMSE	
	Train	Test	Train	Test	Train	Test
1	0.6733	0.3916	0.9943	0.1753	1.1015	0.4773
2	0.7851	0.3360	2.0887	0.6059	1.3045	0.4290
3	0.5909	0.2291	1.1962	0.1453	1.0213	0.2911
4	1.0412	1.3606	1.2276	0.3350	1.3460	1.4210
5	1.1033	1.4612	0.9655	0.1862	1.3112	1.5106
6	1.4010	1.6548	6.9462	0.7082	2.3123	1.7536
7	1.1407	0.4885	17.3023	20.0373	1.7505	0.5986
8	0.7290	1.3950	1.4045	1.2158	1.1779	2.1810
9	0.5390	1.8706	0.7932	2.3763	0.8136	2.0934
10	0.9841	5.1796	3.2213	6.2597	1.4621	5.2194
Mean	0.8988	1.4367	3.614	3.2045	1.3601	1.5975

Table 7 *Values of MAE. MSE, and RMSE for multivariate with 5-minute prediction horizon*

Dataset	MAE		MSE		RMSE	
	Train	Test	Train	Test	Train	Test
1	1.1808	0.2503	8.0696	0.1045	2.8407	0.323337
2	0.8702	0.1468	3.7578	0.0363	1.9385	0.190420
3	0.8135	0.7305	3.3119	0.5445	1.8199	0.737934
4	0.7168	0.1881	2.4360	0.0528	1.5608	0.229709
5	0.7938	0.5477	2.6571	0.3130	1.6301	0.559457
6	1.6934	0.6566	16.4891	0.5049	4.0607	0.710592
7	1.5652	1.0825	14.5005	1.8826	3.8080	1.372090
8	0.9285	0.3778	3.9518	0.1550	1.9879	0.393727
9	0.7953	0.9996	2.4391	1.4771	1.5618	1.215378
10	1.5212	3.7970	10.8179	14.8040	3.2891	3.847595
Mean	1.0879	0.8777	6.8431	1.9875	2.4498	0.958024

Table 6 and Table 7 show all the values of RMSE, MSE and MAE for 5- minute prediction horizon using both the univariate and multivariate models. These metrics are used to measure the difference between the model's prediction and the actual values of the targeted variable. The data was taken at the final selection of epoch at 3 for both models which includes the train data and test data for performance analysis. Comparing the multivariate and univariate, it appears that multivariate model has better performance than the univariate model. This could be because of the increased number of the predictor variable.

Meanwhile, Table 8 and Table 9 shows the evaluation metric (MAE, MSE and RMSE) of the univariate and multivariate for the 10-minute prediction horizon. The error scores in the multivariate are also smaller compared to the univariate model. Moreover, from the results in Table 6 to Table 9, it can be observed that the prediction with 5-minute prediction horizon is better that those from the 10-minute prediction horizon, for both models.

Table 8 *Values of MAE, MSE, and RMSE for Univariate with 10-minute prediction horizon*

Dataset	MAE		MSE		RMSE	
	Train	Test	Train	Test	Train	Test
1	1.1899	0.7281	4.3138	4.6489	1.9122	0.9107
2	1.5591	2.1578	3.2962	1.5582	2.0258	2.2624
3	1.1686	0.6869	3.5282	0.3102	1.8055	0.8642
4	1.1770	0.9883	3.4733	1.4030	1.8734	1.1543
5	1.1029	0.7791	4.0890	0.5460	1.6565	0.9741
6	2.0077	3.0550	10.2641	8.2113	3.0217	3.1771
7	1.5325	1.9106	7.8796	1.8897	2.5247	2.5980
8	1.5148	1.6745	5.9218	5.5305	2.1023	1.9235
9	1.1990	2.3467	1.9598	8.4862	1.6228	2.7147
10	1.3484	8.5585	4.7426	38.1772	2.1876	8.7180
Mean	1.38	2.2886	4.9468	7.0761	2.0733	2.5297

Table 9 *Values of MAE, MSE, and RMSE for Multivariate with 10-minute prediction horizon*

Dataset	MAE		MSE		RMSE	
	Train	Test	Train	Test	Train	Test
1	1.0381	0.7119	3.6606	0.5332	1.9133	0.730185
2	1.0084	1.3325	3.4875	1.7979	1.8675	1.340845
3	0.8894	0.5503	3.6402	0.3422	1.9079	0.584973
4	1.1141	0.3531	4.6729	0.1591	2.1617	0.398927
5	0.9316	0.8360	3.0405	0.8032	1.7437	0.896235
6	2.1686	5.8326	22.5883	35.5949	4.7527	5.966144
7	1.6119	1.3572	11.2511	2.7852	3.3543	1.668880
8	1.0892	0.6774	4.1998	0.5246	2.0493	0.724300
9	1.2218	2.5857	5.2904	9.2137	2.3001	3.035413
10	1.9035	7.5694	15.6290	58.6318	3.9534	7.657138
Mean	1.2977	2.1806	7.746	11.0386	2.6004	2.300304

4. Conclusion

In conclusion, this study presents an effective deep learning algorithm based on the LSTM for time-series blood glucose prediction in univariate and multivariate input models at 5-minute and 10-minute prediction horizon, focusing on type-1 diabetes patients. The models were trained and tested using ten datasets generated from the Cobelli type 1 diabetic model and were evaluated using performance metrics of MSE, MAE, and RMSE. which have provided a quantitative insight into the precision of the deep learning models. Considering the overall performance across both the "Train" and "Test" sets, the 5-minute forecasting consistently outperformed the 10-minute in terms of MAE, MSE, and RMSE and the multivariate seems more reliable in predicting the blood glucose levels. Essentially, the success of this work helps to add contributions to the field of diabetes management, potentially to be implemented in real world applications.

Acknowledgement

This research is supported by Universiti Tun Hussein Onn Malaysia (UTHM) through Tier 1 Grant Research (Q361).

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The author confirms sole responsibility for the following: study conception and design, data collection, analysis and interpretation of results, and manuscript preparation.

References

[1] Song, W., et al. *Predicting Blood Glucose Levels with EMD and LSTM Based CGM Data*. in *2019 6th International Conference on Systems and Informatics (ICSAI)*. 2019

[2] Celik, M.G. and S. Varli. *Deep Learning Approaches for Type-1 Diabetes: Blood Glucose Prediction*. in *2022 7th International Conference on Computer Science and Engineering (UBMK)*. 2022.

[3] Sun, Q., et al. *Predicting Blood Glucose with an LSTM and Bi-LSTM Based Deep Neural Network*. in *2018 14th Symposium on Neural Networks and Applications (NEUREL)*. 2018.

[4] Emmert-Streib, F., et al., *An Introductory Review of Deep Learning for Prediction Models with Big Data.* Frontiers in Artificial Intelligence, 2020. 3.

[5] Afsaneh, E., et al., Recent applications of machine learning and deep learning models in the prediction, diagnosis, and management of diabetes: a comprehensive review. Diabetology & Metabolic Syndrome, 2022. 14(1): p. 196.

[6] Eckhardt, K. *Choosing the right Hyperparamater for a simple LSTM using keras*. 2018 [cited 2023 1/10/2023]; Available from: https://towardsdatascience.com/choosing-the-right-hyperparameters-for-a-simple-lstm-using-keras-f8e9ed76f046.

[7] manmayi. *Choose optimal number of epochs to train a neural network in Keras*. 2023 [cited 2023; Available from: https://www.geeksforgeeks.org/choose-optimal-number-of-epochs-to-train-a-neural-network-in-keras/.