Artificial Intelligence for Diabetes Mellitus Type II: Forecasting and Anomaly Detection

Kareem Arab, Zied Bouida, and Mohamed IbnKhala
Sensor Systems and Internet of Things Laboratory,
Systems and Computer Engineering Department,
Carleton University, Ottawa, Canada
Emails: {kareem.arab, zied.bouida, mohamed.ibnkahla}@carleton.ca

***Abstract*—Diabetes Mellitus Type II (T2D) is a Chronic Disease and is the most common type of Diabetes in the world, responsible for 95% of all Diabetes patients. T2D is a very complex disease and requires a large amount of self-management from the patient in order to maintain a healthy and threat-free life style. Therefore, we develop in this paper a data analytics solution to assist in the self-management of T2D patients through several methods consisting of a rule-based system, anomaly detection, and threat forecasting.**

***Index Terms*—Artificial Intelligence, Anomaly Detection, Diabetes, Chronic Diseases, and Forecasting.**

I. Introduction

Diabetes is one of the most dangerous and complex chronic diseases on the planet. Its complex nature makes it hard to sustain a healthy life, therefore it requires vast amounts of self-management in terms of nutrition, sleep activity, physical activity, stress-management psychological-management, and many more factors that contribute to overall well-being of the patient. Diabetes claims the lives of more than two million people a year. In 1980, the number of diabetic people around the world has reached 108 million and it is currently at 422 million [1]. This number is expected to rise to 552 million by 2030 [2]. It is a known fact that the best method to achieving a healthy and threat-free life style, is through thorough and rigorous self-management. Each patient suffering from T2D is forced to go through several steps of self-management; psychological treatment, continuous medical care and self-management (physical activity, sleep activity, nutrition, . . .).

To the best of our knowledge, there has not been any published research that attempts to combine multiple machine learning techniques into a single system. There have been relevant studies on the topics of glucose forecasting using many different algorithm as well as research on the automatic detection of anomalies in blood glucose. Throughout literature, glucose forecasting has been tackled using Support Vector Machines (SVM), Neural Networks (NN) and many more machine learning techniques. One technique that attempted to tackle the forecasting of blood glucose was by using a combination of the Gaussian Processes and Data assimilation which yielded worthy glucose forecasts [4][5] yet, on its own does not contribute much to the patient's well-being. The important step here is to utilize this analysis and gain the maximum amount of information possible. Other literature we came across attempts to use the Hidden Markov Model (HMM) to automatically detect blood glucose anomalies [3].

Therefore, we intend to create a system that employs AI tools to detect and predict threats, and based on that, recommends certain changes to the everyday lifestyle, aiming for a healthier lifestyle. We do this by designing a system based on a multi-module architecture where several analytical techniques are used together to produce extremely useful information about the patient, which can then be use for suggesting healthy recommendations that can help the patient with their self-management. Our analysis was done on types of data; a) Real data from the UCI data set [6] (low-quality data) and b) generated data. The system is divided into preparation, pre-processing, analytics, evaluation and recommendation stages. The core analytics take place in the analytics engine when we employ a rule-based system, anomaly detection and threat forecasting. We make use of SVMs with Non-linear Kernels to detect blood glucose concentration anomalies and use this data to train a Recurrent Neural Network (RNN) with Long Short Term Memory (LSTM) cells to forecast glucose-based threats in correlation with other factors such as nutrition, and activity.

The proposed solution has the potential to achieve effective results because of the way it is designed. Most of the research done focuses on fitting the problem around certain algorithms, in our case it is the opposite; we designed the algorithms around our goal.

In light of the above, the contributions of this paper can be summarized as follows:

- Data generation, Preparation and Pre-processing
- Analytics Engine design
- Rule-Based System

- Glucose Anomaly Detection
- Glucose Forecasting

The remainder of this paper is organized as follows. Section II presents the details behind the system architecture. Section III explores the process of data preparation and pre-processing. Section IV dives into the technicalities of the analytics engine. In section V, we explore the mathematical formulation of the models we used. Section VI, displays our results and discussion. Finally, section VII concludes the paper and introduces our plans for future work.

II. System Architecture

The proposed system is designed around a multi-layered arrangement aimed to cover as many "special cases" as possible. Special cases can include data deficiency and bias which can cause problems moving forward with the analysis. The system progresses through a series of steps to ensure the that the overall quality of the analytics engine is held to a high standard. First, data is prepared and pre-processed to guarantee its reliability. Then, the system goes through a rule-based process that classifies the historic glucose levels into categories such as hyperglycemia or hypoglycemia. The system then presents an anomaly detection algorithm that can identify outliers in the data and uses this data to predict when these anomalies can occur again in the future and what might cause them.

Preperation
Pre-processing
Analytics
Evaluation
Recommend-ations

Fig. 1. System Progression

This system's architecture was designed around a "Fail-Safe" model where if a certain module does not work, another one takes over based on the "global confidence metric" as seen in the Analytics Engine Architecture Overview (Figure 2).

III. Data Preparation and Pre-Processing

Before any analysis can be done with the data, it has to go through a process of preparation and pre-processing, where the validity of the data is verified and certain minor fixes are made to the data in order to avoid biases.

A. Artificial Data Generation

The process of generating a data set that will serve as a replacement to real data in the field of E-Health can pose many problems. The results will be too accurate to be used for any form of comparative analysis. Yet, generated data and their results can serve as a reference as to how will the algorithms are performing, and based on the results, tweaks can be easily made to the scripts that generate this data.

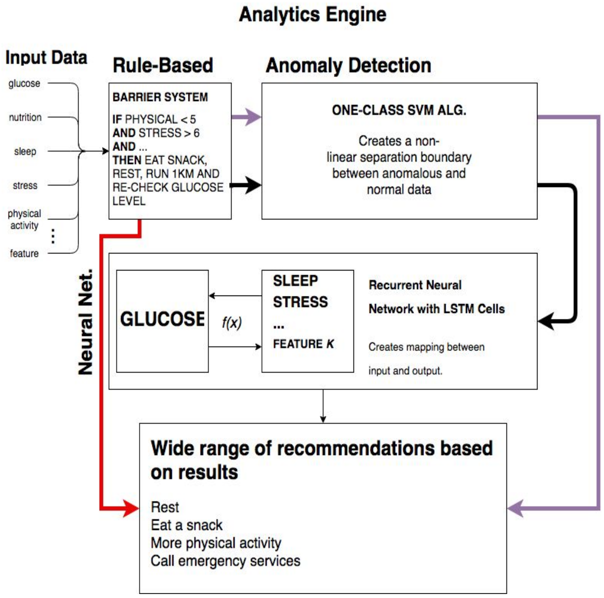


Fig. 2. Analytics Engine Overview: Red: Low quality data - Purple: Medium quality data - Black: High quality data

The first step is to create several profiles for Diabetes patients: poor, average, and ideal; where each profile represents a type of patient. Each profile contains lists of manually created data based on other literature and medical resources. The lists contain glucose concentration, diet, sleep activity, and time inputs. The second step is to generate more data by adding Gaussian Noise to the previously defined profiles.

B. Preparation

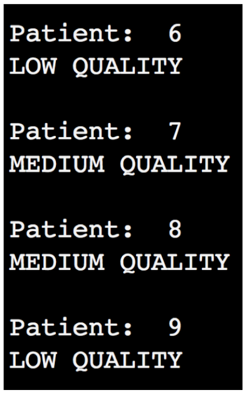
The goal of the preparation module is to check for biases and missing values in the dataset, as well as creating an object for each patient where certain attributes are assigned to. This preparation is essential for the next step, pre-processing. The data is then assigned what is known as the "global confidence metric". This metric assigns a quality level to the data; the data can have **a)** low-quality **b)** medium-quality **c)** high-quality.

C. Pre-processing

After the data has been checked for biases and other abnormalities, certain fixes can be made depending on the data quality levels. If the data is of low quality, nothing is done. If the data is of medium quality, methods such as interpolation are implemented. If the data is of high quality, it won't need any fixing. The next and final step is to normalize all the data and prepare it for the analytics engine.

IV. Analytics Engine

The Analytics Engine is the heart of the system, where the patient data is used to for different types



```
Patient:  6
LOW QUALITY

Patient:  7
MEDIUM QUALITY

Patient:  8
MEDIUM QUALITY

Patient:  9
LOW QUALITY
```

Fig. 3. Example data quality per patient

of analysis, ranging from simpler methods such as the Rule-Based System to Threat Forecasting. Each one of these modules follows a mathematically defined model (more details in the Mathematical Models section).

A. *Rule-Based System*

The proposed rule-based system is intended to classify different levels of glucose as well as other features and based on this classification, makes a decision on what should be done as a result. The system is based on the idea of ”barriers”,

IF glucose concentration level is less than 80 mg/dL
AND/OR feeling hungry,
AND/OR reduced physical activity/fatigue
THEN relax, eat, check blood glucose again.

where, in this example, the glucose level was a barrier and a certain action(s) was taken to resolve the problem.

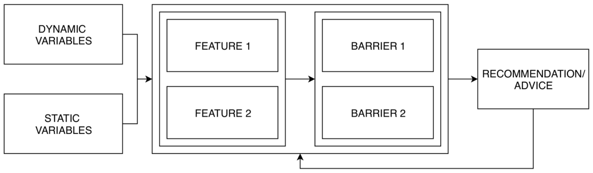


Fig. 4. Rule-Based System

This is a very simple system, that can be used for either **a)** obvious glucose readings that exceed certain medically approved thresholds, or **b)** data that is a accompanied with a low-confidence metric and cannot be used for more complex analysis.

B. *Anomaly Detection*

The anomaly detection method used in our system is based on supervised learning models known as Support Vector Machines. This technique creates a separation barrier between the “normal” values in a dataset and the anomalies. This works by detecting a high density area in a two-dimensional grid, then applying a non-linear barrier to separate the outlying values.

C. *Threat Forecasting*

The anomaly detection analysis done in the second stage of the Analytics Engine is very effective yet is not enough. Detected anomalies are very valuable and can be utilized to produce very interesting results. We take the these anomalies (glucose data) as well as other input data such as nutrition, physical activity, sleep activity, etc. and we feed them into a Recurrent Neural Network with Long Short Term Memory cells in order to forecast if such anomalies can occur again and when that would happen. The activation function used in this RNN model is the sigmoid as detailed in the next Section.

The network’s input layer consists of glucose concentration readings, physical activity, sleep activity, nutrition, stress levels, weight, age; but not limited to these inputs. The output only produces glucose readings. Using the data attained from the anomaly detection module, we create a vector with each anomaly with its corresponding lifestyle-data (physical, sleep, stress, nutrition, etc.) at the same time stamps and we use these vectors as inputs for our network. We then use existing corresponding outputs to train the parameters of the RNN in order to use them for future forecasting.

We approximate the following function

$$y = f(\gamma; x) \tag{1}$$

through back-propagation through time (BPTT) where we minimize the cost function, yielding an effectively approximated γ parameter that can be used with future inputs to forecast unknown output glucose levels.

V. Mathematical Formulation

A. *SVM with Non-linear Kernel (RBF)*

Since the data is linearly separable, the SVM creates a non-linear decision boundary by projecting the data points onto a higher dimension, *d+1*, through a non-linear function where the highest density of data points resides in the absolute minima region of the d-dimensional space. Afterwards, a d-dimensional hyperplane is utilized to separate the data points. This induces a non-linear decision boundary.

- Project some points from $\mathbb{R}^d$ to some space $\mathcal{H}$:

$$\Omega : \mathbb{R}^d \rightarrow \mathcal{H} \tag{2}$$

- Select kernel K such that

$$K(x_i, x_j) = \Omega(\mathbf{x}_i)\Omega(\mathbf{x}_j) \tag{3}$$

- Here we use the RBF (Gaussian Kernel)

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}} \tag{4}$$

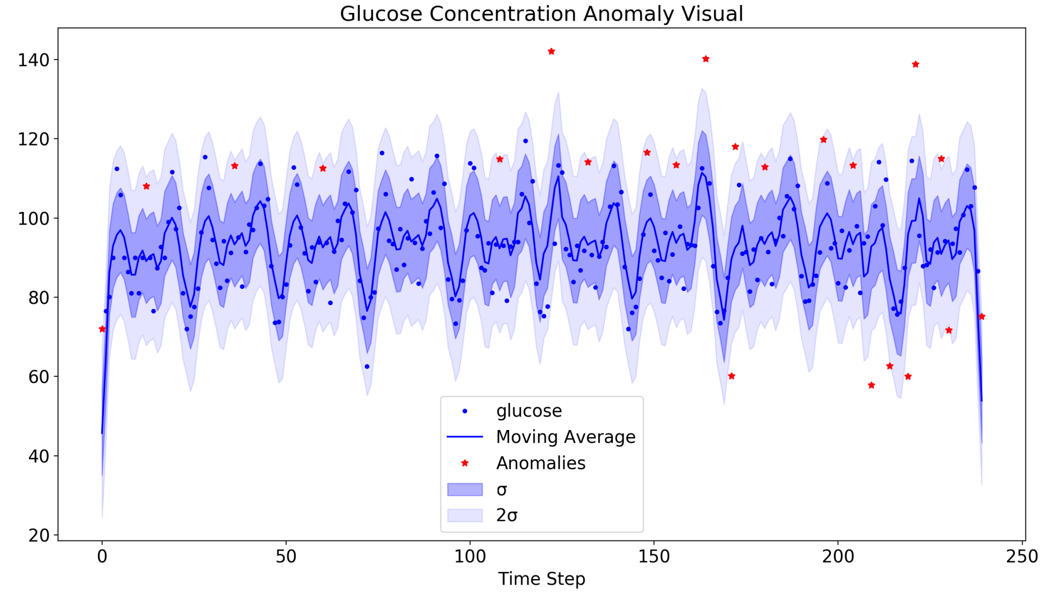


Fig. 5. Anomaly Detection Visualization

- Separation is still linear, yet in a different infinite dimensional space.
- Project the non linear kernel into the space

$$\operatorname{sgn}\sum_{i=1}^{N_s}\alpha_i y_i K(\mathbf{s}_i,\mathbf{x})+b \tag{5}$$

B. *Recurrent Neural Network with LSTM Cells*

Artificial Neural Networks (ANN) are widely used in classification and forecasting problems across many disciplines. Recurrent Neural Networks with LSTM Cells (RNN) are a specific type of ANN. The was RNNs are designed, is by having a "recurrent" element to them, where each RNN block has a set time delay in order to use the output from the previous RNN block as an input. This allows for the induction of a memory gradient (i.e. retaining information through time).

1) Back-Propagation and Vanishing Gradient: Backpropagation is used to calculate the error gradients with respect to their weights by unfolding the Recurrent Neural Network in time and differentiating to backpropagate. Yet, deep neural networks, specifically those of the same nature as RNNs with activation functions such as the sigmoid

$$\phi(x)=\frac{1}{1+e^{-x}} \tag{6}$$

or tanh

$$\phi(x)=\tanh(x) \tag{7}$$

retain an element of extreme temporal depth causing a problem known as the vanishing gradient where the network is not able to retain information through time.

Here ϕ is any activation function, x_t is the input, s_t is the local output (hidden layer) and o_t is the output.

$$s_{t+1}=\phi(Ws_t+Ux_{t+1}+b_s) \tag{8}$$

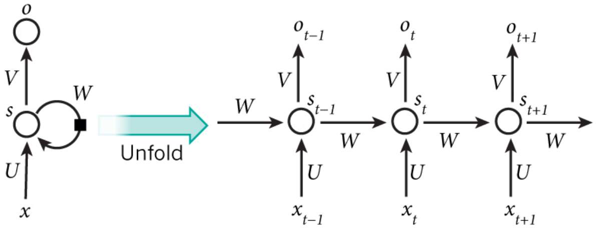


Fig. 6. An RNN being unrolled into a full network. http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

2) Long Short Term Memory: A solution to the vanishing gradient problem can be achieved through using Long Short Term Memory Cells with the following layers.

a) Forget Gate Layer: The forget gate layer mainly looks at the h_{t-1} and x_t variables and returns a number between 0 and 1 using the sigmoid layer in order to decide on what information should be discarded from the cell.

$$f_t=\sigma(W_f\cdot[h_{t-1},x_t]+b_f) \tag{9}$$

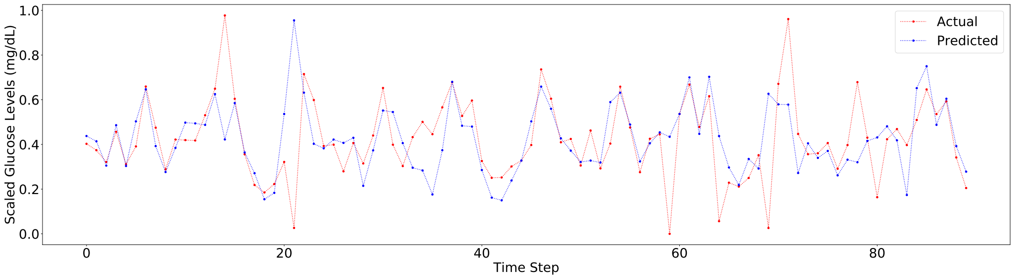


Fig. 7. Threat forecasting using RNN and generated data

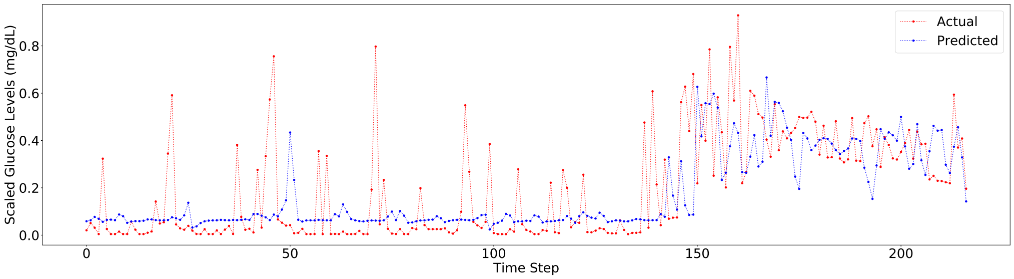


Fig. 8. Threat forecasting using RNN and UCI Dataset

b) Input Gate Layer: The next step is the input gate layer which decides on which information stays in the cells state. Computations are made at two different distinct steps; **a)** which values will be updated and **b)** creation of new candidate values through the tanh layer.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{10}$$

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{11}$$

c) Output or Sigmoid Gate Layer: Finally, we output an altered version of the previous cell state as the final output.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{12}$$

$$h_t = o_t * \tanh(C_t) \tag{13}$$

VI. Results and Discussions

After working with all the different modules mentioned above, we tested the system using generated data due to the lack of open-sourced glucose readings, and yielded the following results:

Fig. 5 is a visualization of the glucose concentration data for one patient. The anomaly detection algorithm successfully identified any anomalies (high or low) occurring in the data.

The results obtained from the anomaly detection module clearly identify and outline all the occurring dangers in terms of glucose concentration. This is highly valuable as this data can be used for two operations. Firstly, to classify this data into a medical diagnosis, and secondly, to use this data for further analysis. This is done through creating a mapping between other factors and trying to understand what caused these anomalies. Finally, using this data we also try to forecast when this might occur again and why.

As for data forecasting, two types of data sets were used: artificial generated data and UCI data from the Center for Machine Learning and Intelligent Systems.

The performance of the RNN on both types of data varies greatly and this is due to the algorithms ability to analyze the patterns and signatures that occur with the signals. Since the generated data was artificial, it was much easier for the RNN to train on the historic data presented to it and as a result, the glucose forecast yielded much higher accuracy as can be seen in Fig. 7.

In Figs. 7 and 8, direct user inputs as well as anomaly detection data are mapped to each other and predictions are made as to how the glucose concentration might look in the future. Based on this analysis a recommender system can be used to manage the patient's glucose level.

VII. Conclusion and Future Work

In this paper, we develop machine learning algorithms for glucose level forecasting and anomaly detection. Due to the shortage of the available data, we generated a data set that takes different factors affecting Diabetes into consideration. Based on the existing data, we first detected outliers and then implemented glucose forecasting algorithms. The next step is to allocate a highly dense database that can be used for extensive analysis and testing. We also aim at implementing reinforcement learning algorithms that can automate the entire process of patient-system interactions, completely removing the rule-based system.

References

[1] World Health Organization. Global report on diabetes. *Diabetes research and clinical practice.*

[2] DR Whiting, L. Guariguata, C. Weil, and J. Shaw. IDF diabetes atlas: global estimates of the prevalence of diabetes for 2011 and 2030. *Diabetes research and clinical practice, 94(3):311321, 2011.*

[3] Y. Zhu, “Automatic detection of anomalies in blood glucose using a machine learning approach,” in Journal of Communications and Networks, vol. 13, no. 2, pp. 125-131, April 2011.

[4] Albers DJ, Levine M, Gluckman B, Ginsberg H, Hripcsak G, Mamykina L (2017) Personalized glucose forecasting for type 2 diabetes using data assimilation.

[5] P. Kotanko, H. Heiss, Z. Trajanoski, P. Wach and F. Skrabal, “Blood glucose forecasting in patients with insulin dependent diabetes mellitus with the Universal Process Modeling Algorithm,” 1992 14th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Paris, 1992, pp. 898-899.

[6] C. L. Blake, C. J. Merz. UCI repository of machine learning databases. University of California, Irvine, Department of Information and Computer Sciences. 1998.