

Machine learning application development to predict blood glucose level based on real time patient data

György Eigner

Physiological Controls Research Center *Physiological Controls Research Center* *Physiological Controls Research Center*
Óbuda University *Óbuda University* *Óbuda University*
Budapest, Hungary
eigner.gyorgy@nik.uni-obuda.hu

Miklós Nagy

Physiological Controls Research Center
Óbuda University
Budapest, Hungary
nagy.miklos95@outlook.com

Levente Kovács

Physiological Controls Research Center
Óbuda University
Budapest, Hungary
kovacs@uni-obuda.hu

Abstract—The given paper details the development of a decision support system application to help for people with type 1 diabetes mellitus. The developed solution is based on supervised machine learning and it focuses to predict the future blood glucose level to support decision making of patients with conservative therapy. We applied the Tensorflow and Keras framework during our work to make our solutions embedded system compatible. We applied the AIDA diabetes simulator to generate data for the proof-of-concept. We found that our result are promising and the performance of the developed solutions are able to satisfy the requirements related the proof-of-concept.

Index Terms—Neural network, Tensorflow, Diabetes Mellitus

I. INTRODUCTION

Diabetes mellitus (DM) is one of the most common disease affecting millions of people worldwide. DM is a malfunction of the human metabolic system connected to the insulin hormone. The most widespread types of DM are type 1 DM (T1DM) and type 2 (DM), however, other kind of the malady appear in the population [1]. Today's estimation of the number of the diabetic population is appallingly high: 451 million adults within the age of 18 and 99 globally. Moreover, according to the approximations 49.7% of these people are undiagnosed. The predictions for 2045 is 693 million worldwide which is about 7% of the estimated human population [2]. Thus, DM is one of those crucial fields of engineering where such solutions have to be developed which are able to help the affected people in their daily life, help the physicians and decrease the load on healthcare.

DM is a disorder of the blood glucose regulatory system. In the case of T1DM, the body does not produce insulin as a results of a acute autoimmune reaction devastating the insulin producer β -cells of the pancreas. Insulin is the key hormone of transporting the glucose from plasma into muscle, fat, liver and other types of body cells. Thus, without insulin these cells are fasting despite the plasma is full of glucose, namely, energy. The lack of insulin causes serious physiological conditions in the short term which may result coma or death without treatment [1], [3].

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 679681).

Machine learning is one of the biggest promises of cyber-physical systems and decision support systems related to physiology. Many machine learning based application related to DM have been developed which are able to improve the therapeutic outcome [4]–[6]. It have to be noticed that most of them focusing on prediction and quantification of diabetic states [7].

In this paper we introduce the development of a machine learning based solution to predict the future blood glucose levels (BGL) focusing on the support of conservative therapy. Conservative therapy is based on daily BGL measurements by using finger pricking and manual insulin therapy by considering the measured BGL and the carbohydrate intake [2]. This therapy can be seriously improved by applying decision support system using machine learning solutions. In our application we were keen to approach reality in the sense of we have input only when the sampling happens. In this article we applied simulated patient model only to prove our concept. Real patient data will be involved in the next development phase of our research.

The paper is structured as follows. First, we introduce the data source and data generation applied in this study. Then we provide an overview of the development of the machine learning application. After, we introduce our results, findings and conclusions.

II. DEVELOPMENT PROCESS

A. Applied dataset

Our goal was to develop an application which can be applied to predict whether the BGL one hour after the measurement of the features falls into the low, normal or high BGL zones. Furthermore, we have selected the features to approach reality as much as possible – namely, selection of only a few measurable or controlled features which mostly determine the BGL evolution over time. We also considered the generality of the application as well. Namely, we were keen to apply methods which can be generally used regardless the source of data in order to increase the robustness.

We have analyzed what kind of data are necessary in order to fulfill the determined goals and we have selected the following features to be applied:

- The weight of the patient (kg).

- The time when the meal was consumed (s).
- The current glucose level (mmol/L).
- The consumed carbohydrates in g.
- The unit of short acting insulin used.
- The unit of long acting insulin used.

All of the selected features are available for a given patient during his/her daily routine. The timing of meal consumption and insulin administration can be given on an hourly basis as well.

At this point of our research we have investigated two possible sources of applicable data in accordance with the aforementioned goal:

- using of real patient data using continuous blood glucose monitoring sensor or finger prick test;
- using of virtual patient data simulating continuous blood glucose monitoring sensor or finger prick test.

We have decided to use virtual patients at this stage of our development process and switch to real patient data in our future work. In this way, we were able to prove our concept before validating our solution through costly clinical trials.

We have chosen to use the AIDA diabetes simulator [8], [9] to generate the necessary data. The simulator is easy to use and set and freely available. With AIDA, it is possible to create or use already defined virtual patients with different physiology. During our research we used 37 virtual patients from the 40 provided patients by AIDA. These virtual patients are based on clinical data [9]. It should be noted that the simulator uses seconds as the basis for time and the time is counted from the "virtual midnight" equivalent with 0 sec. By using the simulator it is possible to simulate one virtual day at a time, however, the frequency of measurement can be arbitrarily selected (multiple measurements can be set).

The simulator is able to use the body weight, the timing and amount of consumed carbohydrates, the types, amounts and timing of short and long acting insulins as inputs – these parameters can be set beside others. As an output, the simulator provides the BGL level. For example, two sample outputs of different simulations are shown on Figs. 1 – 2.

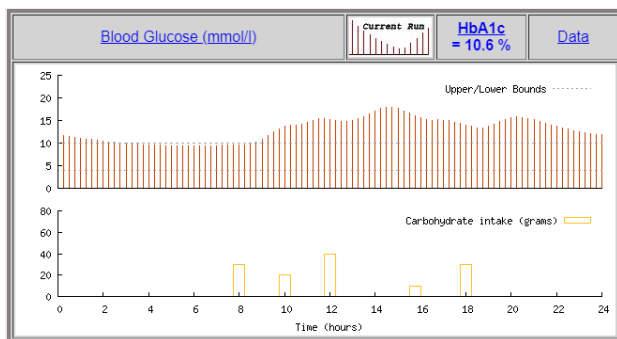


Fig. 1. AIDA simulation result - BGL and carbohydrates

The AIDA on-line simulator requires manual input for every simulation run, the whole process was automated with the help of Selenium [10], a web browser automation tool. Namely, we

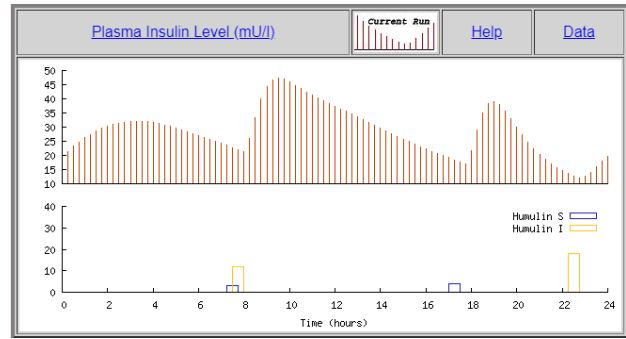


Fig. 2. AIDA simulation result - plasma insulin level and insulin input

have developed a Selenium based script environment that was able to replace the dozens of manual simulations and providing automated run instead.

With these patients, we simulated various scenarios in terms of carbohydrate input and insulin usage. The common points were the logic behind the initiation of the simulations, the data gathering and ordering and the format of the resulting database.

As it was already mentioned, each virtual patient has different habits in terms of time, carbohydrate and insulin intakes, but all of them has six meals a day (breakfast, snack, lunch, snack, dinner, snack), usually with 3 hours difference. We used these features as a start-up and run several iterations with randomized and non-randomized modifications. We randomized the carbohydrate intake in every iteration in a way we generated a random value between -10 and 10 and added to the corresponding meal. By non-randomized modification we meant, that after a certain number of iterations - 5 times going through all the patients with the randomized carbohydrate inputs - we raised the insulin intake by one unit and then repeated it. The result of one run contained a 0-24 hour observation of the patient's glucose level with measurements in every 15 minutes (Fig. 3). After the input and output parameters were known the dataset could be created. Each of the samples is handled as individual instances.

As the result, we got 3560 entries with just a couple of hours of runtime. As we wrote above, we applied the introduced circumstances during the data generation. It should be noted that we include the multiple measurements of the same patient during the one day long simulation as different.

Each entry of the generated dataset contains 8 columns consists of the 6 selected features (the time of the measurement is given in hours and seconds as well that is why 7 columns are given, however, this information is the same one) and the output to be predicted. An example of the resulting dataset can be seen in Table I which belongs to a given virtual patient at a given simulated day.

In the realization of the final dataset we have applied normalization in case of all features. Hence, all of the six features have fallen between 0 and 1 – which is a suitable input for the planned neural network based solution.

Time	BGL
0	14.97555
0.25	14.97555
0.5	14.65713
0.75	14.09202
1	13.39083
1.25	12.64035
1.5	11.87754
1.75	11.17869
2	10.56508
2.25	10.00348
2.5	9.489319
2.75	9.030949
3	8.632641
3.25	8.295779
3.5	8.019724
3.75	7.791038
4	7.60156
4.25	7.449335
4.5	7.319413
4.75	7.22585
5	7.165158

Fig. 3. Example export of AIDA of the first 5 hours of the day

Weight	Time (h)	Time (s)	BGL (mmol/L)	CHO (g)	SAI (U)	LAI (U)	BGL _{T+1} (mmol/L)
68	7:30	27000	7.376	40	6	0	5.891
68	10:00	36000	4.252	19	0	0	4.978
68	12:00	43200	7.733	46	4	4	7.176
68	16:00	57600	5.243	15	0	0	6.046
68	19:00	68400	8.839	34	3	0	8.874
68	22:00	79200	10.41	13	0	0	11.804

TABLE I

EXAMPLE ENTRY OF THE DATASET. BGL: BLOOD GLUCOSE LEVEL; CHO: CARBOHYDRATE; SAI: SHORT ACTING INSULIN; LAI: LONG ACTING INSULIN; BGL_{T+1}: BLOOD GLUCOSE LEVEL ONE HOUR LATER

B. Outcome of the prediction

The main idea behind this research is to get advantage the computation power of modern machine learning based solutions and try to solve the glucose prediction as a classification problem. In this interpretation the possible outcome of the prediction is whether the glucose level is in the range of low, normal or high values as follows:

- Low: $BGL < 5.0$ mmol/L.
- Normal: $5.0 \text{ mmol/L} \leq BGL < 8.0$ mmol/L.
- High: $8.0 \text{ mmol/L} \leq BGL$.

These values and boundaries are usually applied in the case of manual therapy and when there is no distinguishing between the pre- and post-prandial phases [1].

In our framework, the selected categories have applied as the output of the neural network – thus, the output of the net was "Low", "Normal" and "High" and the probability of prediction.

C. Development framework

To implement the desired model we used Tensorflow [11] (v1.13.1), an open source machine learning framework. Tensorflow provides a high level API implementation for Keras [12] (v2.2.4), an open source library to create neural networks.

One of our aim was to develop a model which can be applied in embedded framework as well – due to our goal was to include it into mobile application. Beside its many other advantages, Tensorflow is a very convenient tool to satisfy this requirement. By using the Tensorflow Lite, which exists

exactly for this purpose, we have the possibility to integrate our already trained model into smartphone applications.

D. Developed neural network solution

During our development process we have investigated several network configuration. In accordance with our further aims we were keen to find the most simple network structure which is accurate enough but its usage does not require too much resources in an embedded framework, however.

Based on our investigations we have chosen a 3 layered, fully connected neural network as it can be seen in Fig 4.

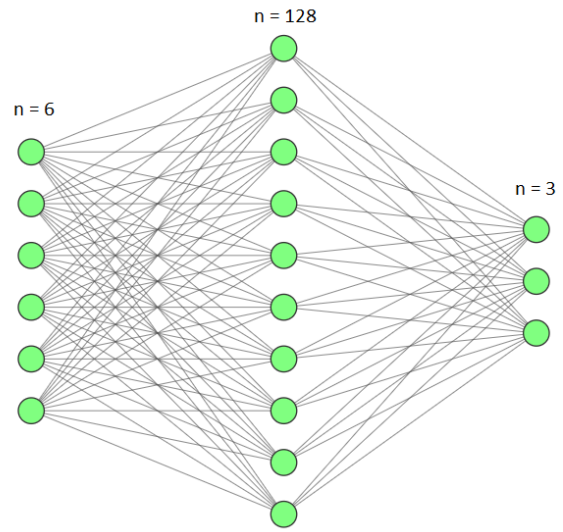


Fig. 4. The structure of the neural network

The input layer has 6 neurons, each represents the previously discussed features (weight, time, current glucose level, carbohydrate input, short acting insulin input, long acting insulin input). The hidden middle layer has 128 neurons with ReLu (rectified linear unit) activation function. As Figure 5 shows, this function provides a 0 output in case of a negative input and it returns the value itself for non-negative inputs. Using of ReLu in the hidden layers does provide many benefits due to its computational simplicity, linear behavior, representational sparsity and easier training [13]. The output layer has 3 neurons, according to the 3 expected classes. These neurons uses the softmax activation function which returns a probability between 0 and 1 (Fig 5).

E. Training method

To train our model we used supervised learning. Supervised learning means that the training data already contains the desired outputs then again the unsupervised learning where the training data is unlabelled and the computer is left to find out some kind of hidden structure by itself [13].

During our work we applied built-in solutions of Keras for training and optimization of the network, namely, backpropagation and the Adam optimizer which is a sort of an extension

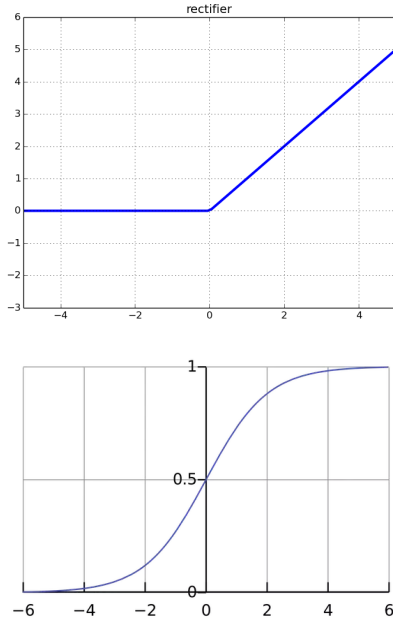


Fig. 5. ReLU (upper diagram) and Sigmoid (lower diagram) activation functions.

of regular stochastic gradient descent method [14]. Adam is a good choice from that point of view as well to support our intention to use as less computational capacity as possible even during the training.

The distribution of our 3560 data entry was the following: 2800 entry for the actual training and 760 for testing (all values has been scaled and normalized to fall between 0 and 1).

III. RESULTS AND DISCUSSION

The confusion matrix (Figure 6) and the scatter charts (Figure 7 - 8) show how our model performed on our 760 test data (on scatter plots Glucose class 1.0 means "Low", 2.0 means "Normal" and 3.0 means "High"):

In Figs 7-8 the test have cut apart into two parts for the better representation. It is visible in both figures that the prediction was quite accurate in case of the "Low" cases, only a few false prediction appeared. In case of the "Normal" and "High" values the results are not easy to decide the absolute precision of the predictions. Furthermore, the high BGL are over-represented in the test set randomly which make the evaluation harder.

Based on the confusion matrix, the values of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) for each class are shown on Table II.

To evaluate the performance of the model, the following metrics were applied:

$$\text{Accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{Total}}, \quad (1)$$

$$\text{Specificity} = \frac{\text{True negatives}}{\text{True negatives} + \text{False positives}}, \quad (2)$$

BGL predictor Confusion Matrix - Test Data

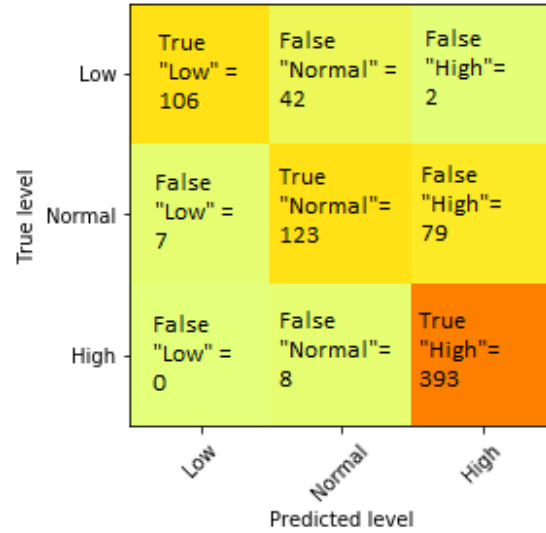


Fig. 6. Confusion matrix of the trained model

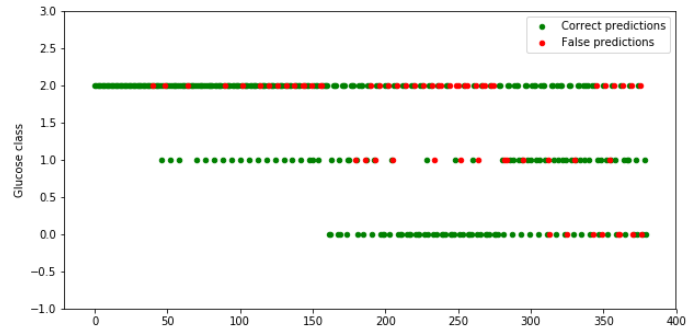


Fig. 7. Correct and false predictions - 1. Horizontal axis means the number of the sample while the vertical axis means the predicted group.

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}, \quad (3)$$

$$\text{TPR(Recall)} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}, \quad (4)$$

$$\text{FPR} = \frac{\text{False positives}}{\text{True negatives} + \text{False positives}}, \quad (5)$$

Since it is a multi-class model, our approach was the One vs All methodology [15] and then the calculation of the average.

For accuracy, the model achieved 0.879, which basically shows us how often the model is correct in general. It should be noted the prediction capability is good in this sense. The specificity was 0.891, which represents the correctly identified negatives. Precision, which shows what proportion of the predicted positives is truly positive, was 0.817. In terms of TPR, the outcome of the model was 0.759, and 0.109 for FPR. TPR represents the correctly identified positives and FPR represents the incorrectly identified negatives. It's worth to

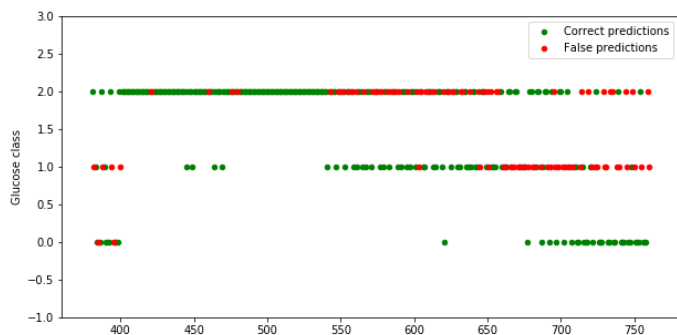


Fig. 8. Correct and false predictions - 2. Horizontal axis means the number of the sample while the vertical axis means the predicted group.

	Low	Normal	High
TP	106	123	393
FP	7	50	81
TN	603	501	278
FN	44	86	8

TABLE II
THE AVERAGED METRICS

mention that F1 score could be calculated from precision and recall, but since it doesn't take the different misclassification costs into account, the result could be misleading (classifying glucose level as high when it's normal cannot do much harm compared to classifying glucose level as normal when it's actually low).

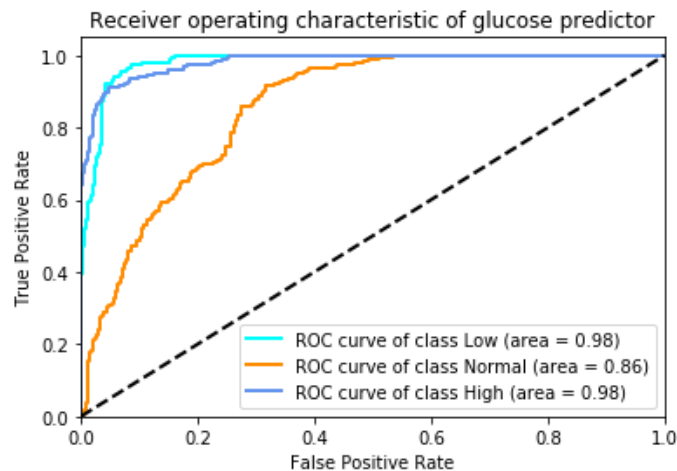


Fig. 9. ROC curve of the three classes

After the observation of the given confusion matrix, metrics and the ROC curve (Fig 9 – 10), it's already visible that the model performs better on classifying "Low" and "High" glucose levels. For class "Normal" the model does have smaller uncertainty in the prediction, especially, when it comes to True Positive Rate, which is just slightly above 50%. The individual TPR for class "Normal" was 0.589. It can be also seen on the ROC curve that the AOC value for class "Normal" is significantly lower compared to the other two. The other

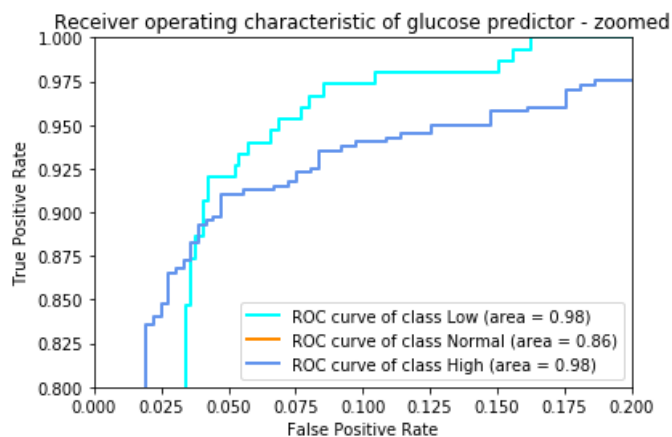


Fig. 10. ROC curve of the three classes - zoomed in

weak point of the performance is the relatively high FPR for class "High", which scored 0.226 individually. Even though the model's overall accuracy was above 80% but we found that further investigation and improvement will be necessary in our future work to be sure to avoid the danger of false classification of "Normal" BGL.

IV. CONCLUSIONS AND FUTURE WORK

We have developed a neural network based model which is able to predict whether the patients have low, normal or high glucose levels 60 minutes after the measurements (sampling). We have found that the developed model is acceptably accurate in case of predicting the low and high BGL with respect to the testing, however, the prediction accuracy of normal BGL should be improved. Nevertheless, the resulting model can be applied as a part of a decision support system.

In our future work, we will be working on to increase the performance of the developed neural network by trying new optimization algorithms and different activation functions. Besides, as it was mentioned earlier, our main focus is to create a reliable predictor for real-world usage, that's why we kept in mind the ability to integrate our model to mobile devices. With parallel to the research, a native Android application development has started and proved the feasibility of our solution. Additionally, to the predictor function, the app is supposed to work as a digital logger for insulin, meals, etc., and has the ability to visualize various statistics about our intakes and glucose level. Also, the testing the developed solutions in clinical studies will be the part of our further research.

REFERENCES

- [1] R. I. Holt, C. Cockram, A. Flyvbjerg, and B. J. Goldstein, *Textbook of diabetes*. Chichester, UK: John Wiley & Sons, 2017.
- [2] N. H. Cho, J. E. Shaw, S. Karuranga, Y. Huang, J. D. da Rocha Fernandes, A. W. Ohlogge, and B. Malanda, "IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045," *Diabetes Research and Clinical Practice*, vol. 138, pp. 271–281, Apr. 2018.

- [3] R. De Keyser, D. Copot, and C. Ionescu, "Estimation of patient sensitivity to drug effect during propofol hypnosis," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, 2015, pp. 2487–2491.
- [4] A. HAYERI, "Predicting future glucose fluctuations using machine learning and wearable sensor data," 2018.
- [5] E. Daskalaki, P. Diem, and S. G. Mougiakakou, "Model-free machine learning in biomedicine: Feasibility study in type 1 diabetes," *PloS one*, vol. 11, no. 7, p. e0158722, 2016.
- [6] A. Z. Woldaregay, E. Årsand, T. Botsis, D. Albers, L. Mamykina, and G. Hartvigsen, "Data-driven blood glucose pattern classification and anomalies detection: Machine-learning applications in type 1 diabetes," *Journal of medical Internet research*, vol. 21, no. 5, p. e11030, 2019.
- [7] I. Contreras and J. Vehi, "Artificial intelligence for diabetes management and decision support: literature review," *Journal of medical Internet research*, vol. 20, no. 5, p. e10775, 2018.
- [8] E. Lehmann, T. Deutsch, E. Carson, and P. Sönksen, "Aida: an interactive diabetes advisor," *Computer methods and programs in biomedicine*, vol. 41, no. 3-4, pp. 183–203, 1994.
- [9] A. Team. (2011) AIDA diabetes simulator. [Online]. Available: <http://http://www.2aida.org/>
- [10] S. HQ. (2019) Selenium Browser Automation. [Online]. Available: <https://www.seleniumhq.org/>
- [11] G. B. Team. (2019) Tensorflow. [Online]. Available: <http://https://www.tensorflow.org/>
- [12] F. Chollet *et al.*, "Keras," <https://keras.io>, 2015.
- [13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [15] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.