Abstract—Diabetes self-management relies on the blood glucose prediction as it allows taking suitable actions to prevent low or high blood glucose level. In this paper, we propose a deep learning neural network (NN) model for blood glucose prediction. It is a sequential one using a Long-Short-Term Memory (LSTM) layer with two fully connected layers. Several experiments were carried out over data of 10 diabetic patients to decide on the model’s parameters in order to identify the best variant of it. The performance of the proposed LSTM NN measured in terms of root mean square error (RMSE) was compared with the ones of an existing LSTM and an autoregressive (AR) models. The results show that our LSTM NN is significantly more accurate; in fact, it outperforms the existing LSTM model for all patients and outperforms the AR model in 9 over 10 patients, besides, the performance differences were assessed by the Wilcoxon statistical test. Furthermore, the mean of the RMSE of our model was 12.38 mg/dl while it was 28.84 mg/dl and 50.69 mg/dl for AR and the existing LSTM respectively.

I. INTRODUCTION

DATA mining (DM) techniques are useful tools for extracting valuable knowledge from (large) databases that helps in decision making [1], [2]. DM has been fruitfully used in different subfields of medical informatics such as diabetes [1], [3], cardiology [1], [4] and cancer [1], [5].

This paper deals with the application of DM for diabetes which is a chronic illness caused by a disorder in the glucose metabolism. There are mainly two types of diabetes: 1) Type 1 Diabetes Mellitus (T1DM) when the pancreas does not produce enough insulin, and Type 2 Diabetes Mellitus (T2DM) which results from an ineffective use of insulin [6]. If not well managed, this disease can lead to serious problems such as heart attacks, kidney damage, blindness, unconsciousness and even death [6]. The prediction of blood glucose level (BGL) is an important task in the diabetes management and self-management as it can help controlling the BGL by taking appropriate actions ahead of time [7]. To predict the BGL, the previous glucose measurements are required. The BGL can be measured: 1) Manually by self-monitoring of blood glucose (SMBG) using sticks several times a day or 2) Automatically by continuous glucose monitoring (CGM) using sensors [6], [7].

According to El Idrissi et al. [7], considerable work was done for the BGL prediction and various Data Mining approaches including statistical methods and machine learning techniques were investigated for that purpose; the most used ones are Artificial Neural Networks (NNs) and Auto Regression (AR) [7]. Recently, deep learning modeling is gaining more interest, such as LSTM NN [8] and deep NN [9].

This paper proposes a deep learning NN with one LSTM layer and two fully connected layers for the prediction of BGL using CGM data. Predicting glucose using LSTM Nns is promising [8] since LSTM Nns are successfully applied in other domains such as prediction of water quality [10], electricity consumption [11] and stock prices [12].

This work aims at: (1) Setting the parameters of the model to identify the best configuration of our LSTM NN; and (2) Assessing and comparing the accuracy of the proposed model to existing ones. Toward this aim, two research questions are discussed:

(RQ1): Can the proposed LSTM model achieve good performance?

(RQ2): Is the proposed LSTM model significantly more accurate than existing models?

This paper is structured as follows: Section II presents an overview of LSTM NNs. Section III summarizes the related work on predicting blood glucose. Data used and performance measurement are described in Section IV. Section V describes the experimental design. Results are reported and discussed in Section VI. Threats to validity of this study are presented in Section VII and finally conclusion and future work are presented in Section VIII.
II. LSTM NNs: An Overview

Deep learning in NNs is an emerging method that allows the NN to learn automatically the characteristics of data by selecting the relevant features [10], contrary to the classical NNs that require features’ selection based on domain knowledge [9].

LSTM NNs are deep recurrent NNs (RNNs) that were introduced by Hochreiter and Schmidhuber [13] to overcome the problem of exploding or vanishing gradient encountered with traditional RNNs [8]. The LSTM NNs are suitable for sequential data such as speech, video and time series as they can capture long term dependencies [14]. They consist of memory cells with a cell state which is maintained over time, and a gate structure that controls and regulates the information of the cell state.

Fig. 1 illustrates the structure of a memory cell. The index \( t \) refers to time or sequence. \( X_t \), \( Y_t \), \( h_t \), and \( C_t \) represents respectively the input, the output, the hidden vector and the cell state for \( t \).

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. Each of these gates is a NN whose input vector is a concatenation of the hidden vector of the previous cell and the input vector. Let \( W_i \), \( W_f \), \( W_o \) be the weight matrices corresponding respectively to the input, forget and output gates; and \( b_i \), \( b_f \), \( b_o \), the corresponding bias vector. \( W_i \) and \( b_i \) are the weight matrix and the bias vector used for updating the cell state.

The result of the input gate is \( i_t \), which is obtained as follows:

\[
i_t = \sigma(W_i \ast [h_{t-1}, X_t] + b_i)
\]  

To calculate \( f_t \), the forget gate uses the following equation:

\[
f_t = \sigma(W_f \ast [h_{t-1}, X_t] + b_f)
\]  

The output gate uses the equation (3) to obtain \( o_t \) and the equation (4) to get the hidden vector:

\[
io_t = \sigma(W_o \ast [h_{t-1}, X_t] + b_o)
\]  

\[
h_t = o_t \ast \tanh(C_t)
\]  

The cell state is updated as follows:

\[
C_t = f_t \ast C_{t-1} + i_t \ast \hat{C}_t
\]  

where 

\[
\hat{C}_t = \tanh(W_c \ast [h_{t-1}, X_t] + b_c)
\]

In the Equations (1 to 4) and (6), \( \sigma \) and \( \tanh \) are the activation functions, the former is the sigmoid function defined in Equation (7) and the latter is the hyperbolic tangent function defined in equation (8).

\[
\sigma(x) = \frac{e^x}{1 + e^x}
\]  

\[
\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}
\]

\( \sigma \) and \( \tanh \) are defined on the set of real numbers. \( \sigma \) ranges between 0 and 1 while \( \tanh \) ranges from -1 to 1.

III. RELATED WORK

El Idrissi et al. [7] investigated by the mean of a systematic map and review the use of predictive techniques in data mining for the diabetes self-management. The study summarized and reviewed 38 studies published from 2000 to April 2017 with regards to: 1) publication’s year and source, 2) diabetes’ type, 3) investigated clinical tasks, 4) the used predictive techniques and 5) their performance.

The main findings of [7] are the following:

1. There is a growing interest to the use of DM predictive techniques in the last decade. Conferences and Journals are used for publication, nevertheless, the main publication channel is conferences by 76.32% of the selected studies, while just 23.68% were published in journals.
2. T1DM gained more attention for research than T2DM with 84.21% of the selected studies.
3. Considerable work was done for the BGL prediction comparatively to other clinical tasks, in fact, 57.89% of the considered papers investigated BGL prediction.
4. Various DM predictive techniques were investigated, and the most used ones are NNs and AR.
5. NNs and AR models yield the highest accuracies. However, none of the used DM predictive techniques is dominant over the others.

Table I reports the findings of a set of selected studies from the review of El Idrissi et al. [7], in addition to the two recent studies [8], [9]. From Table I, we note that:

1. Various techniques are investigated to the BGL prediction problem: statistical methods such as AR [15], [16] and Kalman Filter (KF) [17]; machine learning methods such as Artificial NNs [18]-[20] and Support Vector Regression (SVR) [21], [22]; and recently deep learning techniques [8], [9].
TABLE I. LITERATURE OVERVIEW OF THE BGL PREDICTION

<table>
<thead>
<tr>
<th>Reference</th>
<th>Technique</th>
<th>Diabetes type</th>
<th>Input Data</th>
<th>Findings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lu et al., 2010 [15]</td>
<td>AR</td>
<td>T1DM</td>
<td>CGM</td>
<td>AR models yield accurate BGL prediction with short length signals; it is not required to consider exogenous inputs explicitly nor all frequency bands of the glucose signals.</td>
</tr>
<tr>
<td>Novara et al., 2016 [16]</td>
<td>AR</td>
<td>T1DM</td>
<td>CGM</td>
<td>A blind identification using AR technique was proposed to predict the BGL and recover the unmeasured inputs.</td>
</tr>
<tr>
<td>Wang et al., 2013 [17]</td>
<td>KF</td>
<td>T1DM</td>
<td>CGM</td>
<td>The model based on an extended KF achieves mostly reliable BGL predictions, and made significant improvement compared to zero-order hold.</td>
</tr>
<tr>
<td>Zarkogianni et al. 2011 [18]</td>
<td>RNNs</td>
<td>T1DM</td>
<td>CGM</td>
<td>The proposed RNN model to simulate the blood glucose–insulin metabolism makes it possible to personalize the system and to handle the environment’s variations with efficiency.</td>
</tr>
<tr>
<td>Allam et al., 2011 [19]</td>
<td>Feedforward NNs</td>
<td>T1DM</td>
<td>CGM</td>
<td>For short prediction horizon, feedforward NN based model gets accurate BGL prediction without time lagging.</td>
</tr>
<tr>
<td>Mathiyazhagan &amp; Schechter, 2014 [20]</td>
<td>Fuzzy NNs</td>
<td>T1DM</td>
<td>CGM</td>
<td>This study proposes a soft computing approach which tolerates imprecision by using a fuzzy NN to predict the BGL.</td>
</tr>
<tr>
<td>Bunescu et al., 2013 [21]</td>
<td>SVR</td>
<td>T1DM</td>
<td>CGM</td>
<td>The incorporation of physiological features into an existing SVR model for BGL prediction made a significant performance enhancement.</td>
</tr>
<tr>
<td>Georga et al., 2010 [22]</td>
<td>SVR</td>
<td>T1DM</td>
<td>CGM</td>
<td>Predicting BGL is possible by compartmental models and SVR with a satisfactory accuracy and a clinical acceptability.</td>
</tr>
<tr>
<td>Sun et al, 2018 [8]</td>
<td>LSTM NN</td>
<td>T1DM</td>
<td>CGM</td>
<td>An LSTM network with one LSTM layer followed by one bi-directional LSTM layer and several fully connected layers were proposed for BGL prediction. This LSTM model outperformed the baseline methods ARIMA and SVR.</td>
</tr>
<tr>
<td>Mhaskar et al, 2017 [9]</td>
<td>Deep NN</td>
<td>T1DM</td>
<td>CGM</td>
<td>The proposed deep NN for BGL prediction outperforms shallow NN.</td>
</tr>
</tbody>
</table>

2. Different types of Artificial NNs were explored: RNNs [18], feedforward NNs [19], fuzzy NNs [20], and deep NNs [8], [9].

3. Studies are using data coming from CGM for T1DM patients.

IV. DATA DESCRIPTION AND PERFORMANCE MEASUREMENT

The data set used in this study is the historical data set DirecNetInpatientAccuracyStudy provided by Diabetes Research in Children Network (DirecNet) [23]. It holds collected data of 110 T1DM patients particularly the recorded data from CGM devices which give the BGLs at intervals of 5 minutes. As we did not specify any prerequisite for our model, we considered randomly a subset of 10 patients. However, a pre-processing of the data was done to eliminate redundancy and outliers between consecutive measurements. Table II summarizes information about the considered patients.

To evaluate the performance of our model, we use the root mean square error (RMSE) which is a common performance metric used to assess BGL prediction [7]. It evaluates the difference between the actual value and the predicted one by means of the Equation (9):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{e}_i - e_i)^2}$$

(9)

where $e_i$ and $\hat{e}_i$ are respectively the actual and the predicted values while $n$ is the sample’s size. The value of RMSE goes from 0 to $+\infty$ and the prediction is better when the RMSE is low.

V. EXPERIMENTAL DESIGN

This paper proposes an LSTM NN to predict the BGL of diabetic patients. This NN uses one LSTM layer and two dense layers. To identify the best configuration to adopt, we performed by means of a Grid Search (GS) described in Table III and inspired from [24], the following steps:

Step 1: Train and test the model for each patient by varying the number of LSTM units (LU) according to the GS values of Table III. The chosen value is the one that gives the best RMSE over the patients’ datasets.

TABLE II. DATA SET DESCRIPTION. THE BGL IS IN MG/DL

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>Number of measurements</th>
<th>Min BGL</th>
<th>Max BGL</th>
<th>Mean BGL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PT01</td>
<td>766</td>
<td>40</td>
<td>339</td>
<td>114.78</td>
</tr>
<tr>
<td>PT02</td>
<td>278</td>
<td>57</td>
<td>283</td>
<td>120.96</td>
</tr>
<tr>
<td>PT03</td>
<td>283</td>
<td>103</td>
<td>322</td>
<td>185.89</td>
</tr>
<tr>
<td>PT04</td>
<td>923</td>
<td>40</td>
<td>400</td>
<td>188.44</td>
</tr>
<tr>
<td>PT05</td>
<td>562</td>
<td>50</td>
<td>270</td>
<td>179.71</td>
</tr>
<tr>
<td>PT06</td>
<td>771</td>
<td>62</td>
<td>400</td>
<td>187.45</td>
</tr>
<tr>
<td>PT07</td>
<td>897</td>
<td>42</td>
<td>400</td>
<td>210.26</td>
</tr>
<tr>
<td>PT08</td>
<td>546</td>
<td>43</td>
<td>310</td>
<td>152.88</td>
</tr>
<tr>
<td>PT09</td>
<td>831</td>
<td>40</td>
<td>400</td>
<td>157.50</td>
</tr>
<tr>
<td>PT10</td>
<td>246</td>
<td>72</td>
<td>189</td>
<td>116.51</td>
</tr>
</tbody>
</table>
Step 2: Using the LU obtained in Step 1, we train and evaluate the model for each patient by varying the number of dense units (DU) respecting the GS values of Table III. The chosen value is the one that gives the best RMSE over the patients’ datasets.

Step 3: Using the LU and DU obtained in Step 1 and Step 2 respectively, we train and evaluate the model for each patient by varying the length of the input sequences (SL) according to the GS values of Table III. The chosen value is the one that gives the best RMSE over the patients’ datasets.

Step 4: We compare in terms of the RMSE criterion our best LSTM model with the LSTM of [8] and an AR model.

Step 5: We evaluate the statistical significance of performance differences between the three models by means of the Wilcoxon statistical test [25].

VI. RESULTS AND DISCUSSION

This section presents and discusses the obtained results: (1) We present the empirical results of the Steps 1 to 3 of the experimental process related to the parameters’ setting. (2) The results of the performance comparison of our model to the two other models are presented. (3) We report the results of the statistical test. And (4), We discuss all the empirical results.

Our LSTM model is developed using Python 3.6 with the framework Keras 2.2.4 and Tensorflow 1.12.0 as backend and under the operating system Windows 10.

A. Parameters selection

The main objective of this research is to propose an LSTM model that achieves good performance (RQ1). To do that, a phase of parameters selection was carried out. This phase consists of the three first steps of the experimental design of Section V. The results of the first step are shown in Fig. 2 which represents the RMSE for each patient according to the search space of LU defined in Table III. We observe that the RMSE is in general better for LU=50. Therefore, at that step the LU was chosen to be 50 units.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Signification</th>
<th>Search space</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM Units</td>
<td>Number of neurons in the gates</td>
<td>{5, 10, 20, 30, 40, 50, 60, 70}</td>
</tr>
<tr>
<td>Dense Unit</td>
<td>Number of the neurons in the dense layer</td>
<td>{10, 20, 30, 40, 50}</td>
</tr>
<tr>
<td>Sequence length</td>
<td>Dimension of the input vector</td>
<td>{5, 10, 15, 20}</td>
</tr>
</tbody>
</table>

Table III.

In step 2, we did the same experimentations with LU=50 and we vary DU according to the GS values of Table III. Fig. 3 shows the results obtained: we observe that for DU=40 and DU=50, the RMSE is worse than the others. Furthermore, the values 10, 20 and 30 of DU gave comparable results. However, by considering the times that a configuration (DU is equal to 30, 20 or 10) is better than the others, the minimum RMSE was reached 4 times, 3 times and once respectively. Therefore, the chosen DU value is 30.

The Step 3 uses LU=50, DU=30, and varies the SL according to the GS values of Table III. Fig. 4 shows that SL=10 achieves in general better RMSE values.

To sum up, the best configuration of our LSTM uses LU=50, DU=30 and SL=10. This LSTM variant will be used to compare it with existing BGL predictors.

B. Models Comparison

To answer RQ2, we applied the LSTM model of [8] referred to as Sun_LSTM and AR on the same dataset. We carried out the comparison with Sun_LSTM since it is, according to the best of our knowledge, the only study that proposed an LSTM model for BGL prediction. Regarding the comparison with an AR model, it is motivated by the fact that AR models are suitable for time series prediction and according to [7] they yield along with NNs highest accuracy rates.

The Sun_LSTM contains one LSTM layer with 4 LU followed by one bi-directional LSTM layer with 4 LU and 3 fully connected dense layers with respectively 8, 64 and 8 DU and finally the output Dense Layer. The SL is set to 4. Regarding the AR model, we used the default model developed in Weka with a lag of 10 similarly to the SL proposed in our model which is equal to 10.

Fig. 5 presents the RMSE got for the three models: we observe that our LSTM model outperforms Sun_LSTM for all patients and outperforms AR in 9 cases over 10 patients.
Furthermore, we have compared the training time for the two LSTM models presented in Fig. 6. Our model needs overall less time to be trained comparatively to Sun_LSTM.

Finally, we recorded the number of epochs required for each patient to train the model. Results are presented in Table IV.

The number of epochs is low and ranges between 46 and 260, the mean is equal to 132.8.

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>Number of epochs</th>
<th>Patient ID</th>
<th>Number of epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PT01</td>
<td>91</td>
<td>PT06</td>
<td>260</td>
</tr>
<tr>
<td>PT02</td>
<td>94</td>
<td>PT07</td>
<td>46</td>
</tr>
<tr>
<td>PT03</td>
<td>259</td>
<td>PT08</td>
<td>49</td>
</tr>
<tr>
<td>PT04</td>
<td>71</td>
<td>PT09</td>
<td>168</td>
</tr>
<tr>
<td>PT05</td>
<td>88</td>
<td>PT10</td>
<td>81</td>
</tr>
</tbody>
</table>

C. Statistical test

We statistically assessed the results obtained based on the Wilcoxon statistical test. It is a non-parametric test that validates if the differences between the compared models are statistically significant by the means of a statistical hypothesis [25]. In our case, two null hypotheses (NH) were considered to test if our LSTM model is better than Sun_LSTM and AR. These hypotheses are the following:

• NH1 : Our LSTM model does not outperform Sun_LSTM.
• NH2 : Our LSTM model does not outperform AR model.

The statistical test was two-tailed considering the significance level $\alpha$ equals to 0.05. Note that the test is considered significant if the result which is the $p$-value is less than $\alpha$.

For our tests, the $p$-value of NH1 and NH2 were 0.00512 and 0.0285 which means that the performance differences between our models and respectively the two other models were statistically significant.

D. Discussion

In this study, we propose an LSTM NN to predict BGL. Our first concern was to determine the best configuration by tuning the three following parameters: LU, DU and SL. As we observe from the results of Fig. 2, Fig. 3 and Fig. 4, these parameters have an impact on the model’s performance. Thus, when building an LSTM model, one has to determine
the optimal values of these parameters. The final model achieved good accuracy comparatively to what was found in literature by [7]. In fact, for the RMSE, the minimum, maximum and mean values were equal to 4.67, 29.12 and 12.38 respectively. On the other hand, our model outperforms a previous LSTM [8] and AR models. Furthermore, the performance of our model over the two others was assessed by the Wilcoxon statistical test which showed that our model significantly outperformed Sun_LSTM and AR.

These results show that the use of LSTM NNs for BGL prediction is promising. In fact, LSTM NNs and Deep NNs in general are gaining more interest due to their performances in different fields [26]. Actually, the deep NNs took advantage from their training algorithms that are computationally efficient [27], and from the large hidden neurons’ number which results on huge number of free parameters [26]. On another hand, the performance of our model over Sun_LSTM can be explained by the fact that our LSTM is deeper than Sun_LSTM: in fact, the former uses 50 LU while the latter uses 4 LU.

However, we believe that the accuracy can be improved by considering: 1) other parameters to tune such as learning rate, the loss function and the optimizer, and 2) more information as input such as medication and exercise.

From another point of view, the training time and the epoch numbers were low, which means that the model can be used on a mobile platform or wearable device [7] which require rapidity [8]. In fact, knowing the importance of the mobile personal health records in health management [28], [29], the deployment of our model on a mobile platform is another direction to explore. Toward that aim, we used TensorFlow as it provides an end-to-end support which goes till the deployment of the models on mobile apps, cloud server and others [30].

**VII. THREADS TO VALIDITY**

Four of threads to validity are identified which are:

Internal validity: This thread concerns the evaluation methodology which can be inappropriate. To overcome this limitation, we used 10 data sets, each of them was divided in training and test sets representing respectively 66% and 34% of the whole data set. The model was trained on the former and evaluated on the latter.

External validity: This thread is related to the perimeter of validity. This was settled by considering 10 diabetic patients randomly taken from a public dataset. Note that some studies used only one dataset as reported in [7]. Furthermore, the datasets are with different sizes, it goes from 246 to 923 instances.

Construct validity: It concerns the measurement validity. To limit this thread, we used the RMSE which is a common performance criterion used to assess BGL prediction [7]. Moreover, it was used by many studies such as [8], [15], [18], [19], [21], [22]. Note that it was used lonely by [15], [21].

Statistical conclusion validity: It affects the conclusion related to the comparison performed. To avoid this thread, we used the Wilcoxon statistical test to assess the significance of the performance differences.

**VIII. CONCLUSION AND FUTURE WORK**

This paper considered a deep NN for BGL prediction which is a sequential model with one LSTM layer and two fully connected layers. Multiple runs were done by varying the three parameters: LU, DU and SL to determine the best configuration. The model achieves good accuracy and significantly outperforms, based on RMSE, an existing LSTM model and an AR model.

These promising results encourage to carry out further investigations using the LSTM NNs. Ongoing work aims at building an LSTM model for multi-step prediction by exploring different strategies from literature. Considering more input data such as medication and exercise may improve the accuracy. Furthermore, a problem that can be encountered is the missing data when for example the patient takes off the CGM device, investigating how to handle automatically the missing data is an interesting direction to consider.

**REFERENCES**


[8] Q. Sun, M. V. Jankovic, L. Bally and S. G. Mougiakakou, “Predicting Blood Glucose with an LSTM and Bi-LSTM Based Deep Neural Network,” In Symposium on Neural Networks and Applications (NEUREL), Belgrade, 2018, pp. 1-5, https://doi.org/10.1109/NEUREL.2018.8586990


