RL and Time-series programming challenge

Villin Victor
github.com/villinvic/UNIBE_RL_Timeseries

April 27, 2022

1 Chosen approach

In order to learn how to predict future Blood Glucose (BG) measures, I considered the use of a recurrent neural network (RNN), and more specifically LSTM layers. As the data is already generated, and that our objective is to predict a single floating value, I assumed that using RNNs instead of RL or transformers would be much more adapted and straightforward.

2 Training

2.1 The model and its optimization

As such, I trained a model (that can be found in config/models/predictor1.csv) composed of a LSTM layer and dense layers, by minimizing the following loss through gradient descent :

$$L = \mathbb{E}[(y_{true} - y_{pred})^2], \tag{2.1}$$

where L is the mean squared error of the difference between the predicted value by the model y_{pred} and the actual future BG value y_{true}.

2.2 The data

I used both 2018 and 2020 datasets to build a more reliable model. Therefore, I omitted the "missing current blood glucose" and "heart rate" features to train the model. The 12 training sub dataset were also split into batches of n trajectories of length m. Those batches were then used to train the model over p epochs:

- First, I train a base model by using all 12 patients data,
- Then, I finetune a model for each patients by training the model only using data of the corresponding patient.

In order to deal with missing data, I filled with linear regression or 0 depending on the nature of the variables. For instance, missing "carbon input" values were treated as zeroes, whereas missing "current blood glucose" values were estimated through linear regression.

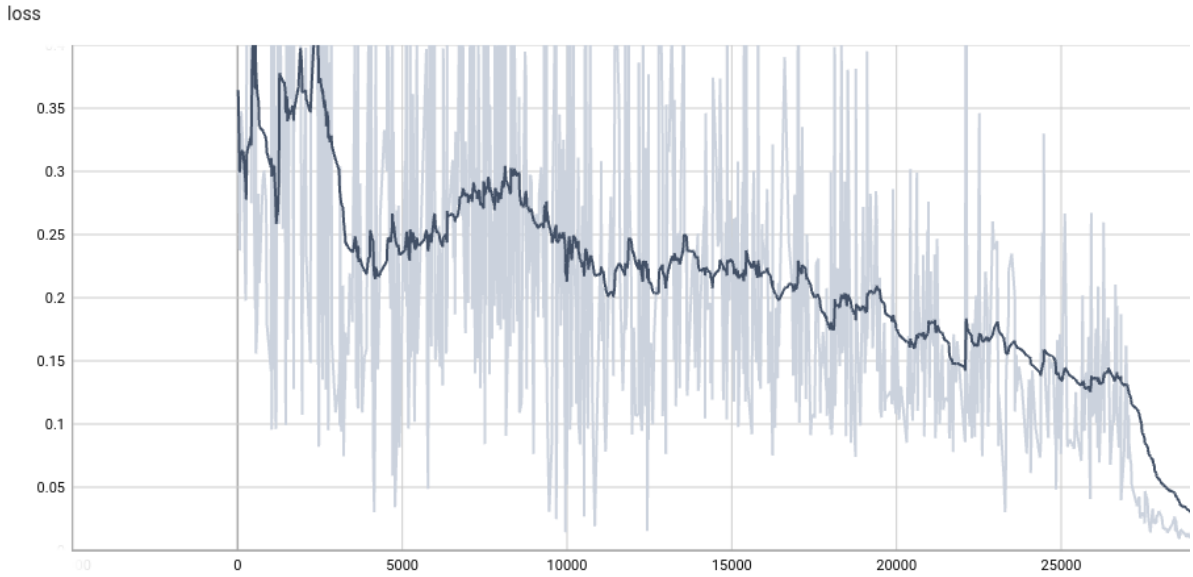

Figure 1: Training loss over the iterations, before and after finetuning for a given patient.

3 Results

I tested the obtained models against the validation datasets, with a prediction goal of one hour (12 timesteps in the future).

As it can be seen over Fig. 1, the model successfully managed to reduce the prediction error. Interestingly, the error consequently drops when we start finetuning the model for a single patient. This strongly suggests that BG fluctuation highly depends on the individual, and that the model needs some customization for each new patient.

This can be further be noted when putting side by side the comparisons made by the base model and a finetuned model (Fig. 2 and Fig. 3). We see that the finetuned model is less likely to miss-predict a spike in BG and more capable when it comes to predicting spikes in BG.

All in All, we can state that the finetuned models are somewhat capable when it comes to making one hour predictions in terms of BG, often succesfully predicting dangerous spikes.

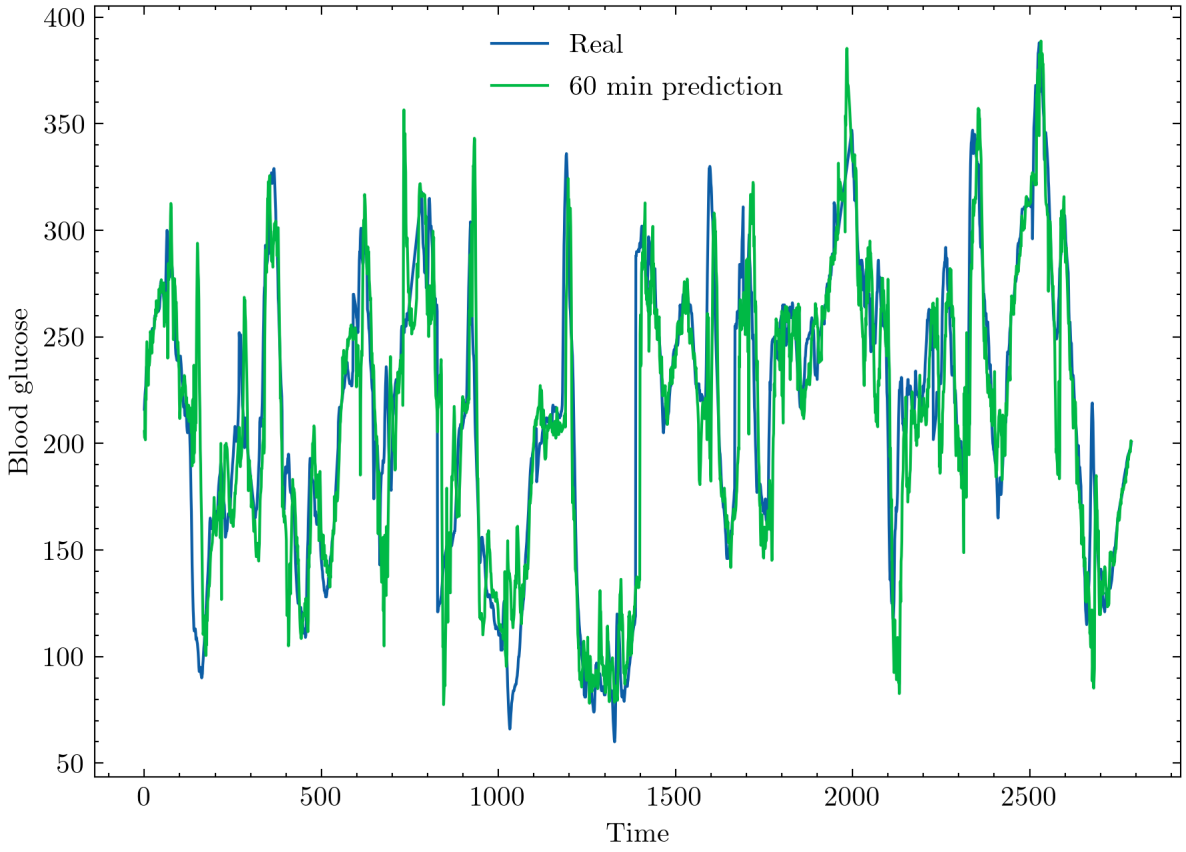

Figure 2: Plot of the predicted values without finetuning against the real values on patient 559.

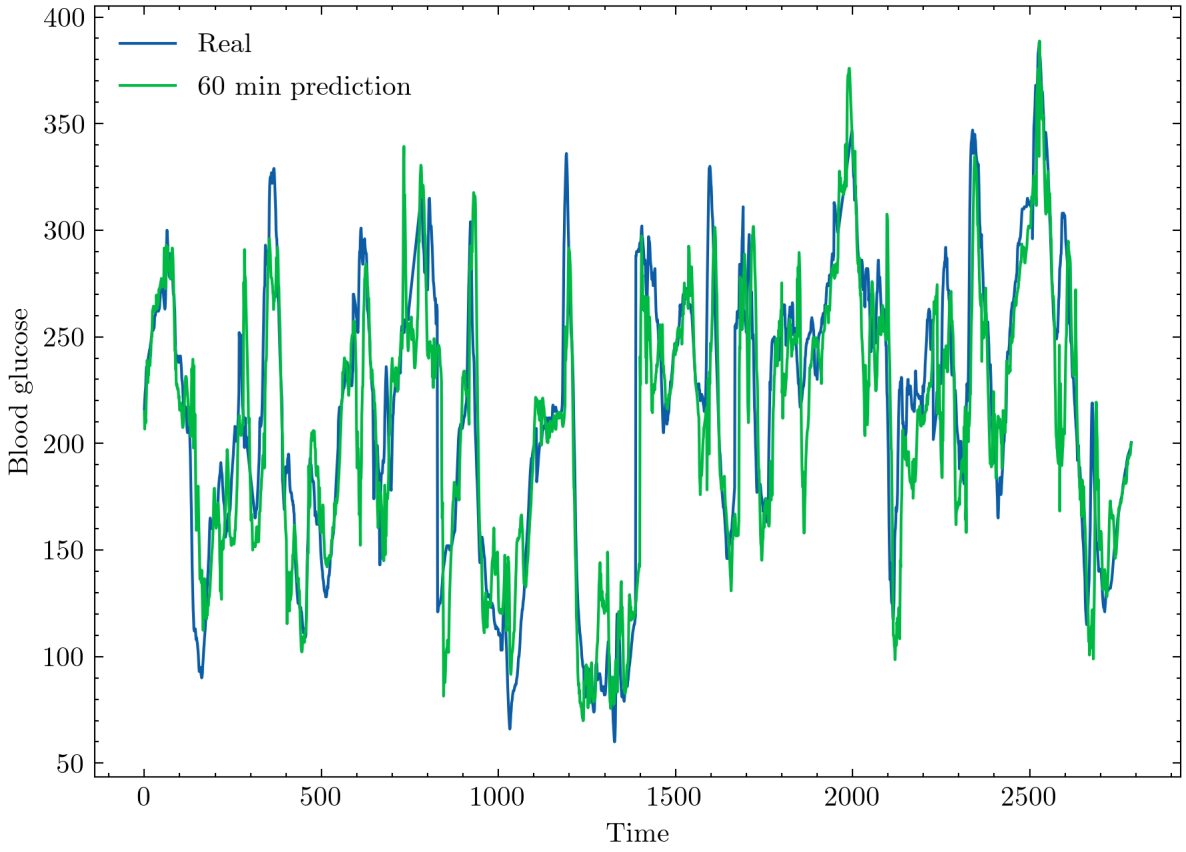

Figure 3: Plot of the predicted values with finetuning against the real values on patient 559.