Summary

Summer 2023

see the jupyter notebooks for more insight on the results(interactive plots)

Contents

1 Context

In previous work, that was shown that low rank RNN (lrRNN) can be used to predict the output of four cognitive tasks. The trajectories of lrRNN can also be fitted to those of a full rank.

In this project, we try to test this on a similar yet different setting, we want to use number of visits of Wikipedia pages to reconstruct the links between pages. Finding the Wikipedia graph would permit to study in what way Wikipedia acts as human brain and how memory and real world events trigger activation of specific part of the graph.

2 Goal

The initial goal was to use Wikipedia data to train the lrRNN model then analyse the connectivity matrix describing the neural network.

3 Summary/General remarks

Even with large rank, the the full rank RNN (frRNN)outperforms the lrRNN in all the experiments we had. However, results from the frRNN are not necessary considered good. Further details will be given.

4 Train the model

Given as input the number of visits on the first n days, predict the number of visits on the next k days.

4.1 first steps

After understanding the paper and the corresponding code, we had familiarize ourselves with the wiki data. We have information related to the number of visits of each Wikipedia page on during 4 months. We only kept the daily visits. We completed missing days with average number of visits.

To avoid complex computations and long wait, we use sub-graphs from Wikipedia.

4.2 Training and results

The aim was to to train the model so that it gives as output the number of visits for each page.

At first we fed the model with noise or null input, the target being the visits per page, but the performance was bad.

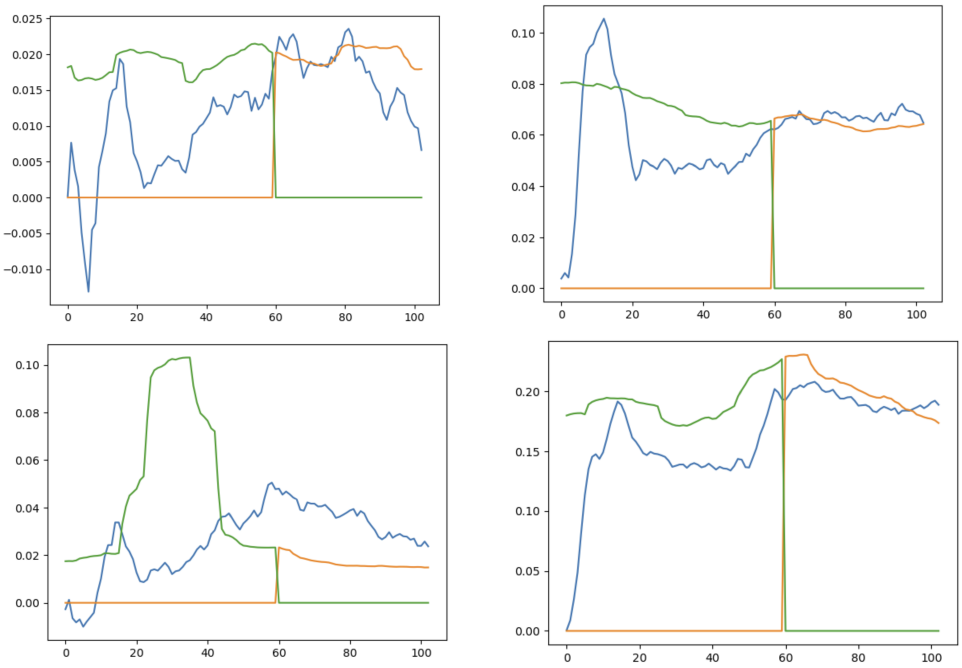

Figure 1: full rank, predictions from day 60

We had to manipulate data and change the task of the model. For that we decided to give as input the number of visits on the first n days, predict the number of visits on the next k days.

for the training step, several

- Manipulations of the data were used:

  - Normalization : both global normalization and normalization signal by signal. We even tried with not normalized data which did not affect the performance.

  - Fourier transformation of the signals was also tried but was not a good idea.

  - Transformation of the data: to lessen the irregularity of the signals we have, we tried multiple ways of smoothing like Gaussian smoothing, taking the log of the signals, moving average... The moving average yields good results and even very good ones for large smoothing window. For some samples of predictions using full rank and smoothed data (moving average), see Figure 1. Predictions from day 60. green: input, orange: target, blue: prediction

  For the low rank RNN with smoothed data (moving average), see Figure 2. Predictions are to be considered from day 80.

Besides, we also tried to make our signals look like the cognitive tasks signals used in the paper mentioned earlier, we used transformation into binary signals. With those rectangular signals, we only keep information on activation (if the number of visits suddenly increases) or deactivation (if the number of visits suddenly decreases) of a page. It makes sense that the model can predict future behaviors of the signals as Wikipedia pages are correlated. Usually a whole bunch of pages that are related somehow (related content for example) get activated together. Indeed, the experiment shows that the pages that had at least 1 activation on the n days of the input get accurate predictions.

We also tried to smooth the data (moving average) then transforming the signals into binary ones. We had interpolation but no Extrapolation (basically the output of the model is the input), see Figure 3. Predictions are to be considered from day 30. green: input, orange: target, blue: prediction

With the results of the experiments above, we deduce that we could make the mode learn from the binary signals and predict. The idea is to merge peaks that are too close and give more importance to the narrow ones by extending them in time. This way, we still have data that resembles to the input used for cognitive tasks. And most importantly, we obviously preserve the idea of activation of a page.

As shown in Figure 4, this experiment yields good predictions. and will be used to get the connectivity matrix.

Overall: we managed to get good accurate predictions using full rank RNN and simplification of the data. We did not get good results using lrRNN even when trying to fit the trajectories of lrRNN to those of the trained frRNN.

- rank

  - Each time both full and low rank RNN were used. Usually the full rank one yields better results.

- parameters

  - Vary the hyper parameters across different runs: learning rate, rank, number of epochs ...

- loss function used

  - MSE (as in the paper)

  - MAE

  - We also tried a variation of MSE were more weight is put when we have peaks. (implemented manually. I don't if this makes sense but think it could help push the model to learn the peaks, though was not that much useful)

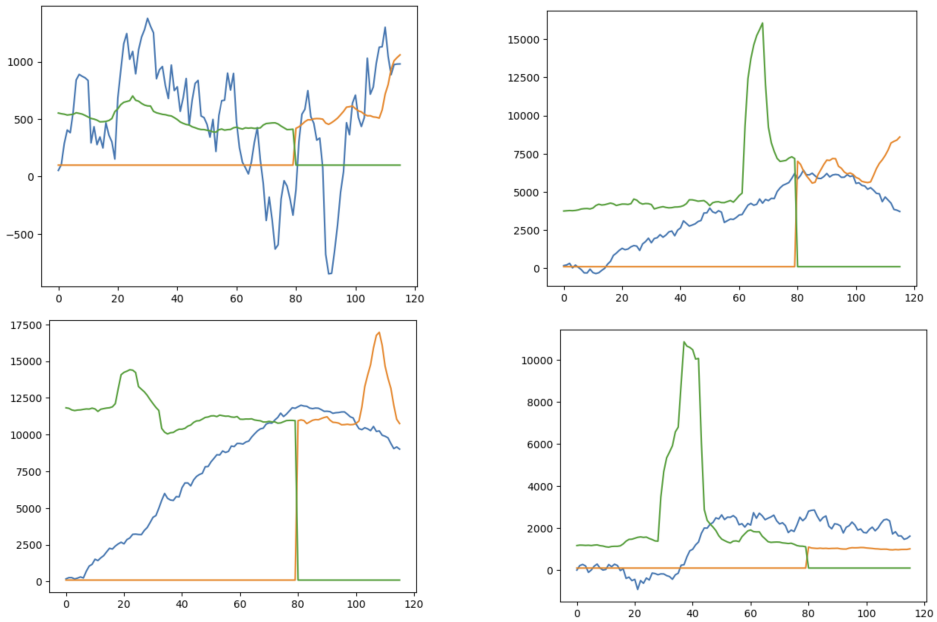

Figure 2: low rank, predictions to be considered from day 80

- when smoothing the data (moving average) then transforming the signals into binary ones. Interpolation but no Extrapolation (basically the output of the model is the input), see Figure 3. Predictions are to be considered from day 30. green: input, orange: target, blue: prediction

5 The connectivity matrix

Now, we will consider the connectivity matrix of the trained full rank RNN. The adjacency matrix describe a graph that is almost fully connected (if we don't consider direction). In order to make the matrix sparse, we perform nullify entries under a fixed threshold to build a an unweighted adjacency matrix. In the Jupyter notebooks, we have interactive plots that show both initial and obtained graphs with the clusters and titles of the pages. It is quite constructive to play with.

6 Conclusion and future work

We had good predictions when keeping only information about activation of each page. As pages are correlated, we think further work can use this idea. Also, in order to have more insight on the behavior of Wikipedia, it would be interesting to make advantage of imitation learning. As it would show in what way Wikipedia acts as a human brain.

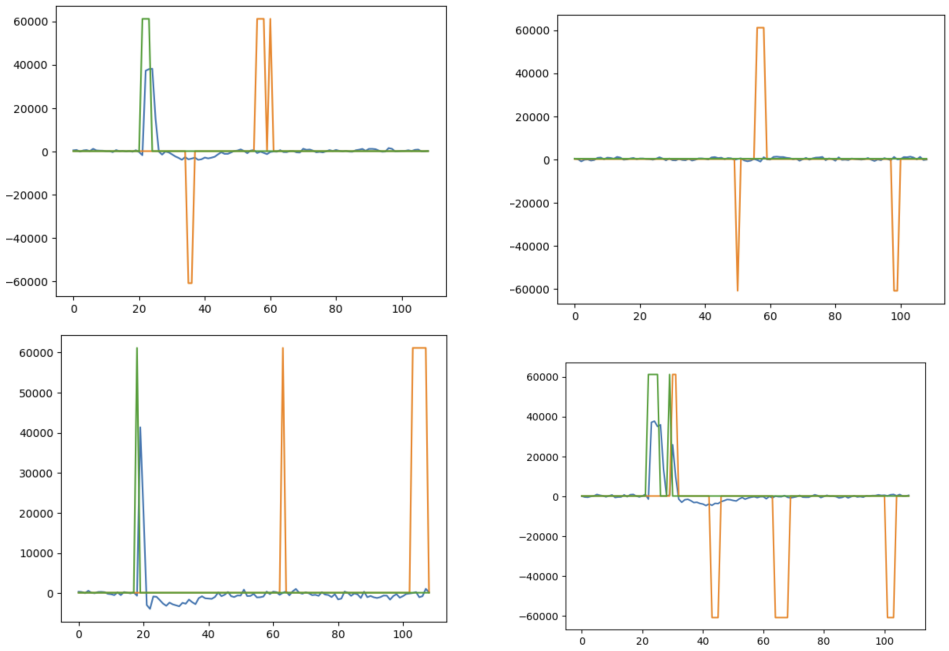

Figure 3: no extrapolation, predictions to be considered from day 30

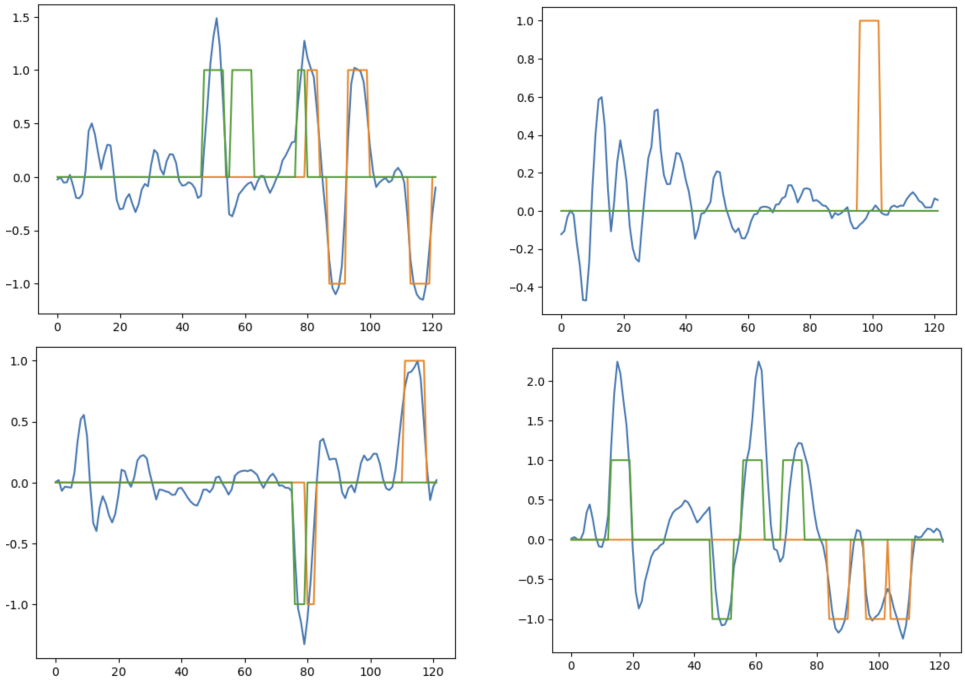

Figure 4: Predictions to be considered from day 80, green:input, orange:target, blue:prediction