Sequence Classification in Brain Functional networks

Khai Nguyen

khainguyen@temple.edu

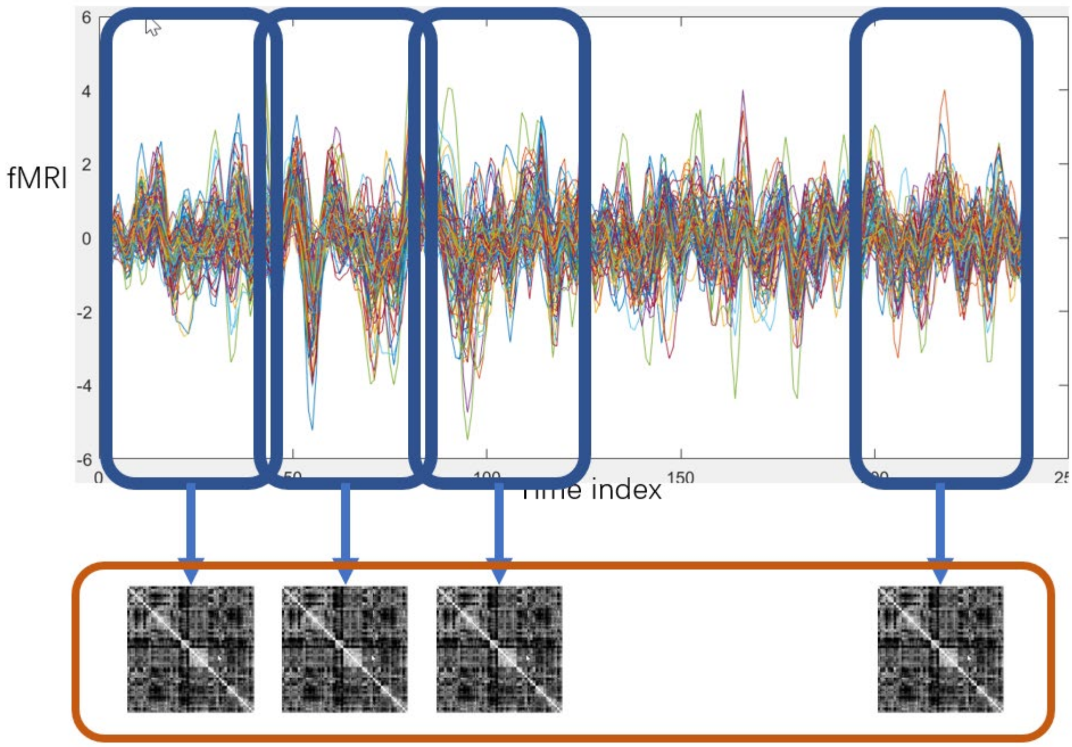
CIS 5525 – Neural Computation

Project description

The goal of this project is to apply deep learning techniques to solve problems in the diagnosis of psychiatric diseases, based on the multi-dimensional time series through fMRI.

Data set.

1. Training data: 246 samples; testing data 20 samples.
2. Each sample is a csv file, with 240 rows (each row is a time point) and 94 columns (each column is a brain region). Namely the csv file is a multi-dimensional time series.
3. The 94 brain regions asssre the same for each sample. The time points, though numbered from 1 to 240 for every sample, do not have any correspondence across different samples.



Each person is represented as a sequence of correlation matrices (brain functional networks)

Task: Use RNN on the raw signal to perform classification of sequence of dynamic correlation matrices.

1. Algorithm design

- We can see this problem is relatively similar to the popular sentiment classification problem where an RNN is used with a linear classifier at the end of the RNN network.
- When working with RNN, the vanishing gradient problem is one that we need to be considered. LSTM were proposed to avoid such problem but due to its longer training time, Gated recurrent units (GRUs) were used as our model.
- At the end of the GRUs there will be a linear classifier with sigmoid activation that squeezes the output into probability of a sequence belonging to class 0/1 (2 nodes).
- **Input** will be of dimension (batch_size, *sequence_length*, input_size_for_each_GRU_cell). **Output** will be of dimension (batch_size, 2)
- The loss function for optimization task is the CrossEntropyLoss implemented in pytorch.

2. Performance curve

246 data samples are splitted into training/validation set of 235/11 datapoint. The experiment was done with 2 GRUs stacked together, with hidden dimension of 256 and drop out probability of 0.2.

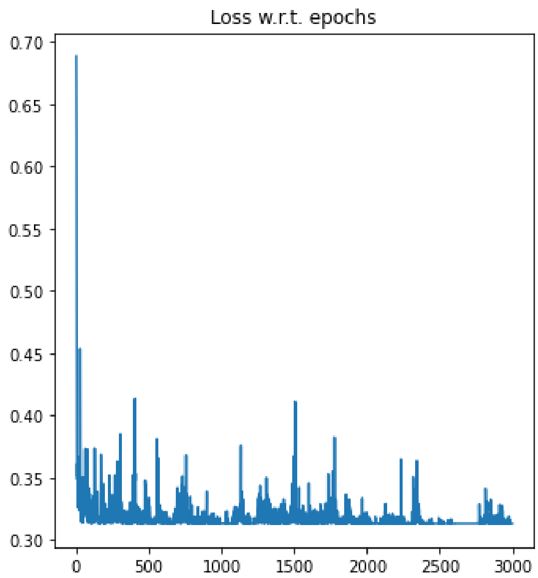


Figure 1. Loss w.r.t epoch on training data

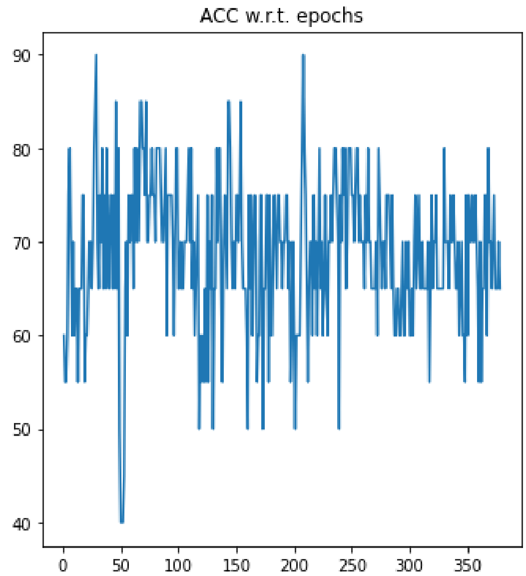


Figure 2. Accuracy on validation data w.r.t epochs

3. Predicted labels for the 20 testing data points (1/0).

File	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Label	0	1	1	1	1	0	1	1	1	1	1	0	1	1	0	1	0	0	1	0	1

4. Discussion

- The model could have performed better with addition of the Attention mechanism.
- In addition, a larger dataset or using appropriate data augmentation methods could have produced better results.

5. References

https://doi.org/10.1016/j.ebiom.2019.08.023
http://www.philipcorr.net/uploads/downloads/367.pdf