A graph neural network for brain functional connectivity analysis

Elena Pitsik, Semen Kurkin, Alexander Hramov,
Rositsa Paunova, Denitsa Simeonova, Sevdalina Kandilarova and Drozdstoy Stoyanov

Abstract

Machine learning methods have been widely applied in neurophysiological data studies for classification and clusterization purposes. However, analysis of connectivity data requires consideration it's graph-like structure to utilize it's structural and functional information. To address this issue, we propose a machine learning model based on graph convolutional network for classification of connectivity matrices obtained from fMRI data. We show that out relatively simple model has a good generalization ability and is able to achieve high performance on moderate amount of data.

fMRI dataset

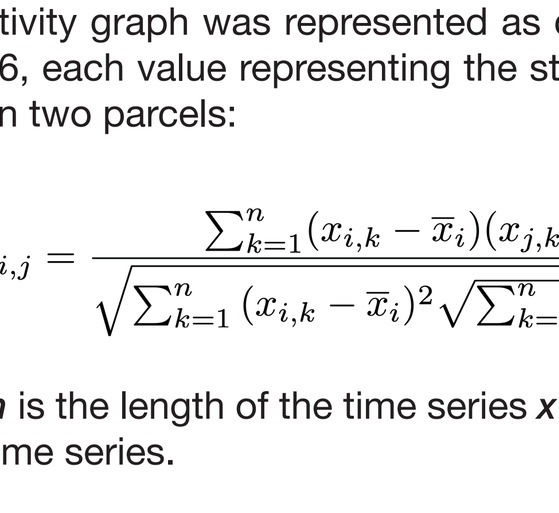
The performance of the proposed architecture was evaluated with rs-fMRI dataset containing 91 correlation matrices: 50 healthy controls and 41 major depressive disorder patients. The scanning procedure was performed on a GE Discovery 750w MRI system. We estimated the connectivity for 166 regions of interest by calculating an average BOLD time series across the voxels in each parcellation ***i*** and Pearson correlation coefficients for all pairs of mean parcellation activities. As a result, each connectivity graph was represented as connectivity matrix 166x166, each value representing the strength of the edge between two parcels:

$$r_{i,j} = \frac{\sum_{k=1}^{n}(x_{i,k} - \overline{x}_i)(x_{j,k} - \overline{x}_j)}{\sqrt{\sum_{k=1}^{n}(x_{i,k} - \overline{x}_i)^2}\sqrt{\sum_{k=1}^{n}(x_{j,k} - \overline{x}_j)^2}}$$

where ***n*** is the length of the time series ***x***, and $\overline{x}$ is the mean of the time series.

A representation of each graph included a node feature matrix, edge data in coordinate format, a 1D vector of edge weights and label. Since each node of the correlation matrix represents a certain brain region, we emphasized their uniqueness by assigning a single feature to each of them, thus obtaining a 166x166 node feature matrix.

The dataset was normalized and randomly shuffled before splitting into training, validation and testing subsets in the ratio 60/30/10 %.



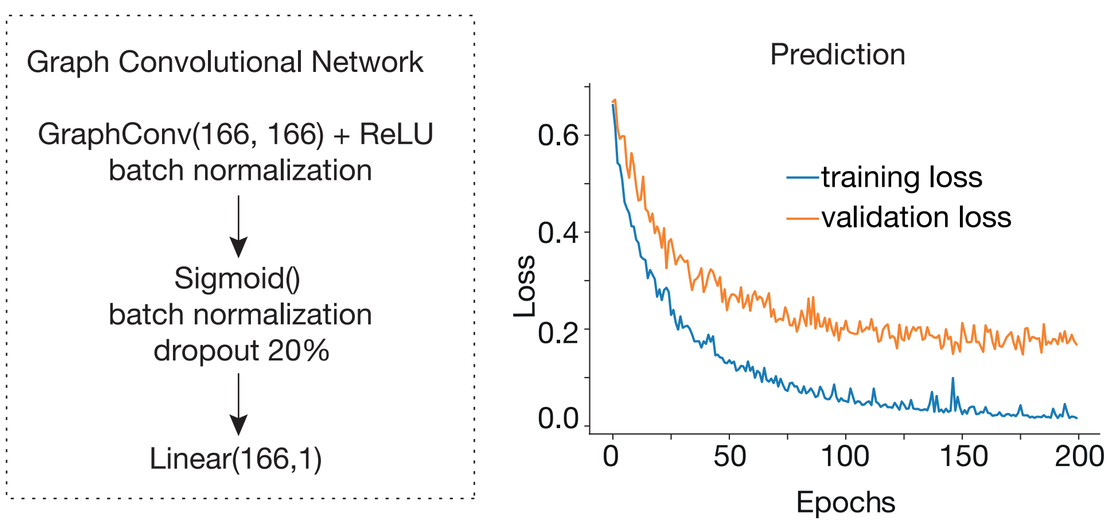
GNN model

The model included one k-dimensional graph convolutional layer (GraphConv) proposed in[1] with 166 input and 166 hidden neurons. The GraphConv layer computes the feature vector $\boldsymbol{f}_k^{(t)}$ as follows:

$$f_k^{(t)}(s) = \sigma\left(f_k^{(t-1)}(s)W_1^{(t)} + W_2^{(t)}\sum_{u \in N_L(s) \cup N_G(s)} f(k)^{(t-1)}(u)\right)$$

where σ is an activation function (we used ReLU activation in proposed model), $\boldsymbol{W_1^{(t)}}$ and $\boldsymbol{W_2^{(t)}}$ are parameter matrices (sequences of weights), $\boldsymbol{N_L(s)}$ and $\boldsymbol{N_G(s)}$ are local and global neighborhoods of the node set ***s***.

The element-wise sigmoid function was applied to the feature vector before proceeding with fully-connected linear layer with one output neuron, which provides the prediction. Before each step, we applied the batch normalization, and before the output layer we used 20% dropout.



The model was trained for 200 epochs using Adam optimizer with learning rate 0.0001. We used binary cross entropy to calculate training, validation and testing loss. The data was fed to the model in batches with the batch size of 32. Although both training and validation loss continue to improve, there is a gap between the curves. Such pattern suggests the problem of unrepresentative validation set, i.e. the too small amount of validation data, which is rather expected considering a relatively small amount of data used to train the model. At the same time, the model was able to achieve training accuracy of 100\% and validation accuracy of 98.89%. We also tested the model on a small amount of data that was never used in the training process and achieved 100% accuracy of classification. Therefore, the model demonstrated a good generalization ability.

Conclusion

In present research, we applied a graph convolutional network to classify rs-fMRI data from healthy controls and major depressive disorder patients. Despite the relatively small amount of initial dataset, the proposed model was able to achieve 100% of testing accuracy without the signs of strong overfitting.

[1]C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, “Weisfeiler and leman go neural: Higher-order graph neural networks,” in Proceedings of the AAAI conference on artificial intelligence, vol. 33, no. 01, 2019, pp. 4602–4609

The reported study was funded by RFBR and National Natural Science Foundation of China according to the research project 19-52-55001.