Abstract

In this project, we analyze different approaches for modeling brain networks, ranging from traditional shallow graph kernel models to modern deep graph neural networks. Our goal is to find models that can be used as a “layer of skin” over traditional graph kernels to aid in the analysis of mental disorders and diseases such as bipolar disorder, HIV, PTSD, and depression. We adapt different graph mining techniques for brain networks, statistically and visually analyze the results, and quantitatively evaluate them in the standard graph classification setting. We found that deep models (GNNs) outperformed shallow models (kernel methods), and the most successful model was able to classify HIV patients with 81% accuracy.

Problem Formulation

- The standard graph classification task considers the problem of classifying graphs into two or more categories; in this project, we perform binary classification on networks.
- Our datasets consist of brain networks, represented as weighted, undirected adjacency matrices constructed from fMRI scans. For more details on network construction, we refer you to Section 3 of a paper by Cui et al. [2].
- Depending on the classification model, we further preprocess the datasets with various methods, such as threshold rounding.

Graph Kernels & SVM

Our first classification method is a “shallow” model: computing graph kernels and plugging them into support vector machines (SVM). We employ three graph kernels: Weisfeiler-Leiman (WL), Weisfeiler-Leiman optimal assignment (WLOA), and propagation (Prop) kernels. Figure 1 shows a high-level visualization of SVM. For more details on graph kernels, we refer you to Section 2 of a recent paper proposing a "deep" graph kernel framework [6].

Results

The highest mean classification accuracy is highlighted (for ties, we highlight the accuracy with the lowest standard deviation).

Graph Neural Networks

Our second classification method is a "deep" model: graph neural networks (GNNs). Specifically, we implement message passing GNNs (MPGNN) using the BrainGB Python package, which is built on the Pytorch and Pytorch Geometric libraries. Section 4 of the paper presenting the MPGNN framework by Cui et al. [2] gives further details on the MPGNN design choices. Figure 2 visualizes the MPGNN architecture.

Lastly, we seek to leverage any higher-order information given by graph kernels as well as local information given by GNNs. To this end, we intend to combine both approaches and benchmark the performance of graph neural networks that integrate graph kernels on our datasets. Of particular interest to us are:
- the graph convolution layer (GCL) proposed by Cusco et al. [1] and
- the kernel graph neural network (KerGNN) proposed by Feng et al. [4].

Future Work

There are many ways to incorporate graph kernels with graph neural networks and make them more interpretable. For example, we seek to integrate the GKL layer from [1] into BrainGB’s MPGNN framework. We also hope to test the performance of KerGNN architecture, illustrated by Figure 3, on our datasets to analyze the effectiveness of incorporating WL graph kernels into GNN’s message passing process.

Acknowledgements

- Thank you to Dr. Carl Yang, our mentor, for his guidance.
- Thank you to Hejie Cui and Dai Wei for their technical assistance.
- This work is supported in part by the US National Science Foundation awards DMS-2010109 and DMS-1754360.

References