Comparing Graph Kernels and Graph Neural Networks for Brain Disease Classification

Midterm Presentation

Erica Choi[1] Sally Smith[2] Ethan Young[3]

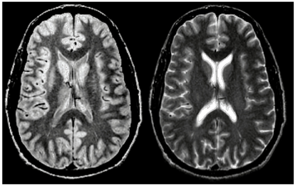
[1]Columbia University

[2]Georgia Institute of Technology

[3]University of California - Los Angeles

Emory REU/RET, 2022

Models Meet Data



[1]

Figure 1: Example fMRI brain scan.

- Can you tell if this brain is diseased or not?

[1] Center for Functional MRI - UC San Diego

Goals & Motivation

- Accurately classify patients as diseased or healthy
- Improve upon existing graph neural network performance by developing novel architectures
- Contribute to computational neuroscience literature by improving models that could eventually be used for mental illness diagnosis

Datasets

- We are working with 2 datasets, each classifying HIV and BP (bipolar disorder).
- Each dataset consists of DTI scans, FMRI scans, and classification labels (diseased (-1)/ non-diseased (1)).
- Both datasets have been cleaned for us and consist of less than 100 patients.

Datasets

- The DTI and FMRI brain scans of each patient i are represented as weighted adjacency matrices $\mathbf{W}_i \in \mathbb{R}^{M \times M}$.
    - FMRI scans are considered to be more robust than DTI scans, so our experiments prioritize working with them.
- Nodes in the brain network represent regions of interest (ROI), and edge links between nodes indicate the strength of the connection between ROI's.

Data Preprocessing

- For our data, we implemented a rounding scheme to remove edge weights and sparsify the adjacency matrices.
- We have: $A_{ij} = \begin{cases} 1 & \text{if } \mathrm{A}_{ij} \geq \alpha \\ 0 & \text{otherwise} \end{cases}$, where A_{ij} is the ij-th entry of the adjacency matrix A and $\alpha \in (0, 1)$ is our rounding threshold.

Data Preprocessing

- We further manipulate the data to obtain a list of (graph) objects that can be used with the Python packages *GraKel* and *PyG*.
- *GraKel*'s functions and classes implement efficient computations of graph kernels to be used for tasks such as classification.
- *PyG* (*PyTorch Geometric*) builds on *PyTorch* and streamlines the implementation of graph neural network (GNN) pipelines.

Classification Task

- The standard graph classification task considers the problem of classifying graphs into two or more categories.
- The goal is to learn a model that maps graphs in the set of graphs G to a set of labels Y.

Graph Kernels

- Graph kernels[2] are popular in graph-based learning and have applications in many fields because their computation boils down to an inner product.
- Our goal is to compute graph kernels and plug them into a kernelized learning algorithm to benchmark their performance on our datasets.

[2] Yanardag and Vishwanathan, "Deep Graph Kernels", 2015

Graph Kernels - Graphlet Sampling

- Intuitively, this counts the frequency of size-k subgraphs and compares that between two graphs.
- This kernel is defined as $\mathcal{K}_{GK}(\mathcal{G}, \mathcal{G}') = \langle \mathrm{f}^{\mathcal{G}}, \mathrm{f}^{\mathcal{G}'} \rangle$.

Graph Kernels - Weisfeiler-Lehman

- Intuitively, this kernel compares the number of shared subtrees between two graphs.
- The W-L kernel is defined as $\mathcal{K}_{WL}(\mathcal{G}, \mathcal{G}') = \langle 1^{\mathcal{G}}, 1^{\mathcal{G}'} \rangle$.

Graph Neural Networks (GNN's)

- GNN's combine node features and graph structures to perform specific prediction tasks
- A generic framework of GNN:
  - computing the representation of each node
  - applying a pooling strategy to obtain the graph representation
  - Multilayer perceptron (MLP) can be applied to make predictions

GAT and GCN

- Graph Attention Network (GAT) is a type of Convolutional Neural Network that operates on graphs
- Graph Convolutional Network (GCN) is a special case of GAT's with attention fully determined by graph structure alone, without node features

BrainGB

- BrainGB: A Benchmark for Brain Network Analysis with Graph Neural Networks[3]
- Measures accuracy, F1-score, and AUC of different parameters
  - Node feature construction
  - Message passing mechanisms
  - Pooling Strategies

[3] Cui et al., "BrainGB: A Benchmark for Brain Network Analysis with Graph Neural Networks", 2022

BrainGB - Node Feature Construction

- Natural node features are usually not available in brain network analysis
- Connection profile

BrainGB - Message Passing Mechanisms

- Message vector

$$m_i^l = \sum_{j \in \mathcal{N}_i} m_{ij} = \sum_{j \in \mathcal{N}_i} M_l(h_i^l, h_j^l, w_{ij})$$

$$h_i^{l+1} = U_l(h_i^l, m_i^l)$$

- Node concat

$$m_{ij} = MLP(h_i \mid\mid h_j)$$

BrainGB - Pooling Strategies

- Pooling strategy

$$g_n = R(\{h_i \mid v_i \in \mathcal{G}_n\})$$

- Concat pooling

$$g_n = ||_{k=1}^{M} h_i = h_1 \mid\mid h_2 \mid\mid ... \mid\mid h_k$$

Support Vector Machines (SVM)

- SVM is a supervised learning model that maps training data to points in Euclidean space, then separates them with a hyperplane.
- Because of the small number of observations, we averaged classification accuracy over 20 different train-test splits to get a handle on how well SVM is performing.

Results - BrainGB

BrainGB Benchmark			
Dataset	Accuracy	F1	AUC
HIV-GCN	$51.43_{\pm 17.73}$	$50.61_{\pm 12.87}$	$49.23_{\pm 17.97}$
BP-GCN	$61.74_{\pm 11.15}$	$65.72_{\pm 7.84}$	$61.06_{\pm 11.24}$
HIV-GAT	$57.14_{\pm 12.78}$	$59.18_{\pm 21.87}$	$51.43_{\pm 18.00}$
BP-GAT	$55.63_{\pm 9.52}$	$59.03_{\pm 9.54}$	$55.49_{\pm 9.51}$

Results - SVC (W-L)

SVC Benchmark (Weisfeiler-Lehman)		
Dataset	Threshold = 0.5	Optimal Threshold*
HIV-dti (0.85*)	$0.40_{\pm 0.18}$	$0.56_{\pm 0.18}$
BP-dti (0.5*)	$0.52_{\pm 0.14}$	$0.52_{\pm 0.14}$
HIV-fmri (0.2*)	$0.58_{\pm 0.20}$	$0.65_{\pm 0.17}$
BP-fmri (0.2*)	$0.53_{\pm 0.13}$	$0.57_{\pm 0.14}$

Results - SVC (GS)

SVC Benchmark (Graphlet Sampling, k=3)		
Dataset	Threshold $= 0.5$	W-L Optimal Threshold*
HIV-dti (0.85^*)	$0.54_{\pm 0.26}$	$0.47_{\pm 0.15}$
BP-dti (0.5^*)	$0.48_{\pm 0.17}$	$0.46_{\pm 0.15}$
HIV-fmri (0.2^*)	$0.30_{\pm 0.14}$	$0.30_{\pm 0.14}$
BP-fmri (0.2^*)	$0.50_{\pm 0.16}$	$0.50_{\pm 0.16}$

Challenges

- Limited data ($<$100 patients in each dataset)
  - Consequently, we need to feed the datasets through our models more, which increases computation time.
- Ethical considerations unique to the field of neuroscience

BrainGB - Limitations

- GNN's are usually shallow; deep GNN's are still an active area of research.
- For brain networks, what kinds of graph structures are effective beyond the pairwise connections are still unknown.

Graph Kernel GNN's

- Kernel SVC basically classifies at random; however, we can still leverage some notion of higher-order information given by kernels in GNN's
- Implement and establish benchmarks with different GNN architectures, such as the one proposed by Morris et al[4]
- Develop novel GNN architectures that incorporate graph kernels; this is motivated by work done by Feng et al[5]

[4] Morris et al., "Weisfeiler and Leman Go Neural: Higher-order Graph Neural Networks", 2019

[5] Feng et al., "KerGNNs: Interpretable Graph Neural Networks with Graph Kernels", 2022

Conclusion

- Our goal is to improve brain disease classification models.
- While we are limited by factors such as accessibility of datasets, we are working around the issues we are facing.
- Our next step is to work on combining graph kernels with GNNs.

Acknowledgements & Contact Info

- Thank you to Emory for bringing us to this research site, and special thanks to our mentor Dr. Carl Yang for his guidance and 24 hour email service.
- Thank you to Hejie for her assistance with using BrainGB.
- Thank YOU all for being here and listening to our talk!

- If you have any further questions, feel free to contact us!
  - Erica Choi - erica.c@columbia.edu
  - Sally Smith - sallysmith@gatech.edu
  - Ethan Young - young.j.ethan@gmail.com

References

- Cui, Hejie et al. “BrainGB: A Benchmark for Brain Network Analysis with Graph Neural Networks”. In: *arXiv preprint arXiv:2204.07054* (2022).
- Feng, Aosong et al. “KerGNNs: Interpretable Graph Neural Networks with Graph Kernels”. In: *AAAI*. 2022.
- Morris, Christopher et al. “Weisfeiler and Leman Go Neural: Higher-order Graph Neural Networks”. In: (2019).
- Yanardag, Pinar and SVN Vishwanathan. “Deep Graph Kernels”. In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015, pp. 1365–1374.