Predict Psychiatric Disorders via Graph Classification on Brain Networks using K-Top Contrast Subgraphs

Ann Huang
Montreal, Canada
zixiang.huang@mail.mcgill.ca

Jenny Cao
Montreal, Canada
yuanhui.cao@mail.mcgill.ca

Alex Liu
Montreal, Canada
alex.liu@mail.mcgill.ca

ABSTRACT

Empirical findings from the neuroscience literature suggest that psychiatric disorders are often associated with altered connectivity patterns between different regions of the brain. At the intersection of neuroscience and network science, it is a growing line of research to exploit the network structure of the brain, and apply graph mining algorithms to extract brain network patterns that differ between healthy individuals and psychiatric patients. Here we aim to refine a previously proposed, highly interpretable graph classification algorithm called "Contrast Graph", in an effort to improve its performance and robustness in discriminating psychiatric patients from healthy individuals. Our goal here is to establish a robust, effective, and interpretable approach to predict psychiatric disorder, and extract interpretable findings from our experimentation on open-source brain-imaging datasets, and map our findings back to the neuroscience literature.

Author Keywords

graph classification, psychiatric disorder, brain network, functional connectivity, contrast subgraph

INTRODUCTION

Neurological or severe psychiatric disorders can often be reflected by the impairments in the structural or functional connectivity of the brain network [10]. The development of brain imaging techniques such as the functional Magnetic Resonance Imaging (fMRI) provides us with powerful tools to construct connectivity patterns between brain regions. Such techniques paves the way to the study of "connectome" in the human brain, which is a comprehensive mapping of the interactions and connections between different brain regions. It inspires us to regard the brain as an interconnected network where each individual functional modality or brain area is the node, and the connectivity between areas serves as the edge. Hence, if the connectivity between brain areas is perturbed in patients with neurological or psychiatric disorders, the graph structure of the patient's brain is highly likely to differ from that of a healthy individual. The problem of diagnosing psychiatric diseases can therefore be framed as, given the graph structure of the brain network obtained from fMRI scan, how can we faithfully discriminate a psychiatric patients from healthy individuals? From a network analysis's perspective, a binary classification of brain networks into "typically developed" class and "functionally impaired" class via graph classification techniques can be used to solve this problem.

In this project, we will focus on improving a highly interpretable graph classification algorithm called "Contrast Subgraph"[5], in an effort to extend its originally proposed computational framework and improve its classification accuracy, and map our experimentation findings back to neuroscience literature. A contrast subgraph is a set of vertices whose induced subgraph is densely connected in one classification group but sparsely connected in another group, which can be used to find the differences between two graphs.

In order to find contrast subgraphs, we first calculate the difference graphs that are obtained by transforming the graphs into matrices, calculate the differences between the matrices, and use the results to establish difference graphs. By extracting the subgraph(s) with highest contrasts (i.e highest connectivity difference) from the difference graph, we will successfully find the contrast graphs. We can then do binary classification on the extracted contrast subgraph of each brain networks. The classification results from this approach are highly explainable in such a way that we can access which brain regions have the largest change in connectivity induced by the psychiatric disorder, and at which point the progression in connectivity change signifies the development of psychiatric disorder.

RELATED WORK

Previous research has proposed to use simple machine learning techniques, such as Support Vector Machine (SVM)[6], Linear Discriminant Classifier (LDC)[4], and logistic regression [9] to separate brain imaging data from psychiatric patients from typically developed individuals. However, these methods completely disregard the network structure of the brain, causing the loss of important information which may affect the classification accuracy.

Later on, the development of deep learning methods has drawn people's attention to its potential application to neuroimaging data analysis. There are successful cases where Deep Neural Networks (DNN) are applied to classify brain imaging data [8]. However, this model tends to have an abundance of parameters and is therefore prone to overfitting. In practice, the number of parameters in a DNN often far exceeds the number of neuroimaging samples we have, making the model hard to

CHI'20, April 25–30, 2020, Honolulu, HI, USA

ISBN 978-1-4503-6708-0/20/04…$15.00

DOI: https://doi.org/10.1145/3313831.XXXXXXX

generalize to unseen data, in such a way that it becomes no longer meaningful to seek diagnosis advice on new patients from such model. In addition, using deep learning methods on brain imaging data also ignores the underlying network structure of the brain.

In another line of research, the researchers have also proposed to use graph embedding methods followed by a Convolutional Neural Network (CNN) model for binary classification on brain imaging data [7]. Despite an effort in exploiting the network structure of the brain and in reducing the number of model parameters, the classification results from this model lacks interpretability and explainability, which is of uttermost importance in diagnosing any type of disorders or diseases.

Lanciano et al. [5] designed a highly interpretable algorithms that extracts contrast subgraphs and classify brain networks based on the contrast subgraph. The detail of this approach will be stated in the Problem Definition section. This approach simplifies the vast number of features related to the brain networks into only two scalar features for linear classification. It is shown to outperform several state-of-the-art graph classification algorithms.

MOTIVATION

Despite outperforming other baseline methods, the classification accuracy of Contrast Graph drops significantly as the heterogeneity of the dataset gets larger, from 83% to nearly 60% on a heterogeneous dataset as shown by the original paper [5]. This suggests that the original approach which extracts only the subgraph with highest connectivity difference between two groups (i.e the top contrast subgraph) for linear classification in a 2D space may not suffice for more complex classification tasks. Therefore, in an attempt to improve the model's performance in face of data heterogeneity and task complexity, we propose to extend the original framework by extracting k subgraphs with highest connectivity contrasts between two groups (i.e the top k contrast graphs), and find a hyperplane to separate two groups of data in the high-dimensional feature space (we will limit k to preserve the interpretability of the classification outcomes). This way, we can strength the robustness of this approach, improve its overall performance, while preserving its explainability, thus providing a highly transparent and effective way of predicting psychiatric disorders from brain imaging data.

OUR PROPOSAL AND CONTRIBUTION

In this project, we propose three objectives:

1. On the algorithmic side, we want to extend the original framework to extract top-k contrast subgraphs, and combine information from multiple contrast subgraphs for binary classification on brain networks, aiming to improve the classification accuracy.

2. On the application side, we want to first apply the top-k contrast subgraph method to the ABIDE dataset used in the original paper to classify Autism Spectrum Disorder patients and healthy individuals. Our aim here is to interpret the contrast subgraph extraction results to find the top-k brain regions that are most affected by Autism Spectrum Disorder. We then seek to apply top-k contrast graph method to other psychiatric datasets such as schizophrenia and ADHD, to validate the classification accuracy as well as to obtain interpretable results regarding the brain regions that are most affected by the psychiatric disorder.

3. For a bigger picture, we want to map our contrast subgraph extraction findings back to neuroscientific and psychiatric literature, to validate our model's ability to capture meaningful information about the underlying functional impairments of the brain under various psychiatric disorders. Successfully doing so will fully exploit the interpretability of this method and will inform researchers about the change in brain network structures induced by psychiatric diseases that are otherwise hard to detect.

PROBLEM DEFINITION

Finding the single top contrast graph

Here we focus on binary classification on brain networks $G = (V,E)$ based on the contrast subgraph, which requires us to find the set of vertices whose induced subgraph has dense connections in one class and sparse connections in another, and classify brain networks based on the extracted contrast subgraphs.

Consider two sets of graphs $A = \{G_1^A, ..., G_{r_A}^A\}$ and $B = \{G_1^B, ..., G_{r_B}^B\}$, which in this context represents the typically developed group and the functionally impaired group. To summarize the high-level connectivity information about group A and group B, we introduce two *summary graphs* $G^A = (V, w^A)$ and $G^B = (V, w^B)$. The summary graphs are defined over the same vertices as the original group of graphs, while w^A, w^B are weight functions that assigns a real value to each pair of nodes representing the average connectivity density between the two nodes in group A and group B, respectively. In particular, for undirected unweighted graphs, w^A is defined to be the fraction of graphs $G_i^A \in A$ in which u and v is connected by an edge:

$$w^A(u,v) = \frac{1}{r_A} |G_i^A \in A \quad \text{s.t.} \quad (u,v) \in E_i^A|$$

The algorithm aims to find the subset of vertices $S^* \subseteq V$ that maximizes the contrast-subgraph objective, which is the absolute value of the difference in average-degree density between the two groups:

$$\delta(S) = \sum_{u,v \in S} (|w^A(u,v) - w^B(u,v)| - \alpha)$$

$$= |e^A(S) - e^B(S)| - \alpha \binom{|S|}{2}$$

where e^A is the sum of edge weights (or equivalently, the number of edges in unweighted graphs) in the subgraph of G^A induced by the set of vertices in S. The penalty term α controls the size of the subgraph to extract.

$$e^A(S) = \sum_{u,v \in S} w^A(u,V)$$

Finding the k-top contrast graphs

We want to find the k subgraphs with maximum aggregated difference in average degree density between two groups, while these k subgraphs cannot overlap with each other too much. This is quantified by the pairwise Jaccard coefficient between the sets of nodes of the k subgraphs. Therefore, we enforce an upper bound on the pairwise Jaccard coefficients between the k subgraphs.

The precise computational problem can be stated as:

$$S = (S_1, ..., S_k) = \arg\max_{S_1,...,S_k} \sum_{i=1}^{k} \delta(S_i)$$

$$\text{subject to} \quad \frac{|S_i \cap S_j|}{|S_i \cup S_j|} \leq \alpha \quad \forall S_i, S_j \subset S$$

$$\text{where} \quad \delta(S_i) = \sum_{u,v \in S} (|w^A(u,v) - w^B(u,v)| - \alpha)$$

is the penalized difference in average degree density between two classes for one single subgraph, as shown in the previous subsection of finding the single top contrast graph.

Previous work by Balulau et al.[1] proposed an efficient algorithm (MinAndRemove) and a simplified version of this algorithm (FastDSLO) for finding the k densest subgraphs within a graph. In the context of brain network classification, the problem of finding the top k contrast subgraphs between the two groups of graphs (typically developed VS functionally impaired) is mathematically equivalent to finding the k densest subgraphs of the difference graph $|w^A(u,v) - w^B(u,v)|$. Therefore, we adopt the algorithm proposed by Balulau et al. to solve our problem.

METHODOLOGY

Graph preprocessing

In order to be used as input to our densest subgraph extraction algorithm, the unweighted undirected brain networks need to be processed to construct the difference graph. The difference graph is defined as

$$D = |w^A(u,v) - w^B(u,v)|$$

where w^A, $w^B$is the summary graph of the group of typically developed brain networks and the graph of functionally impaired brain networks, respectively. We calculate the edge weight of the summary graph of one graph as the fraction of times this edge appears in brain networks within this group, as demonstrated by the following equation:

$$w^A(u,v) = \frac{1}{r_A} |G_i^A \in A \quad \text{s.t.} \quad (u,v) \in E_i^A|$$

where $i = 1, 2, ..., r_A$ and r_A is the total number of brain networks within the group A. We then use the summary graphs of the two groups to construct the difference graph D, and extract the k-densest subgraphs on the difference graph D for further analysis.

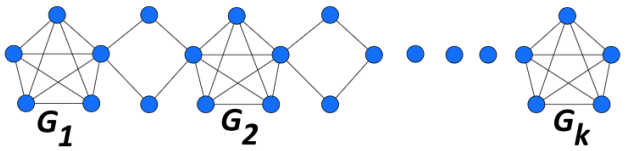

Figure 1. Worst case scenario using the Naive Charikar algorithm to find the densest subgraph.

Extract k-top contrast graphs

In order to extract top-k densest sub-graphs from the data, we first look at an algorithm called the FastDSLO algorithm.

The FastDSLO algorithm uses the linear programming (LP)-based algorithm for the densest sub-graph developed by Charikar[2] as the basis. The Charikar algorithm finds the current least connected node by the number of edges it has, and stores this node and the current average degree of the current graph. After all the nodes have been ranked, we have the order of connectivity and the current average degree of the subgraph for all the nodes. This means that the densest subgraph will be the subgraph that has the highest average degrees, and we can obtain this subgraph by removing all the nodes that have a lower connectivity rank. This enables us to find a 2-approximation of the densest subgraph from the current graph.

In FastDSLO, we set an α to represent the portion of least connected nodes to remove from the extracted subgraphs. Therefore, when α is 0, the entire subgraph calculated from the Charikar algorithm is the densest subgraph. When $0 < \alpha < 1$, we will remove α * m nodes from the densest subgraph calculated by the Charikar algorithm, with m being the total number of nodes in the Charikar densest subgraph. The reduced densest subgraph will be a more minimalized densest subgraph, which is more efficient in this case. In order to obtain the next densest subgraph, we will remove the currently found densest subgraph from the data, and then use the Charikar algorithm on the remaining graph again.

Nevertheless, the Charikar algorithm doesn't guarantee a minimal solution. In the example in Figure 1[1], the graph has an average degree of 2, since there are 10k + 4(k - 1) edges and 5k + 2(k - 1) nodes, with k being the number of pentacles in the graph. If we run the Charikar algorithm on this graph, the result for the densest subgraph will be the entire graph. However, the minimal densest subgraph is the disjoint set of pentacles in the graph. Since we are trying to minimize the overlap of our top K densest subgraphs, this shows the potential for a poor output. Therefore, we need a more robust way to extract the densest subgraphs.

To that effect, we also make use of the MinAndRemove algorithm. This is essentially the same as FastDSLO with a minimizing process added. It achieves this by first running the Charikar algorithm[2] on the input graph to obtain a 2-approximation G', same as FastDSLO. However, in this case, it is only to speed up the following steps. As such, the algorithm then removes from G' all nodes with degrees less than its average density, while recording the densest resulting sub-

graph G''. Once that is done, in order to further minimize G'', the algorithm picks a node uniformly at random and, using the same linear programming method as Charikar[2], attempts to find a densest subgraph G''' of G'' which doesn't contain that node. If it manages to do so, a random node is picked from G''' and the process is repeated. Otherwise, there is then at least one densest subgraph which contains that node. Using the same LP technique, we find every such subgraph and return the one with the smallest cardinality. This effectively minimizes the 2-approximation[1].

It should however be noted that while MinAndRemove outputs more optimal solutions compared to FastDSLO, the latter still holds merit in that it is substantially faster, and therefore is more appropriate for very large datasets. Therefore, we will be examining the efficacy of both.

Binary classification

Our previous step finds the top-k densest subgraphs in the difference graph, which means that we have determined the top-k regions that show the highest connectivity change between the typically developed group and the functionally impaired group. Now we want to assess whether such local changes in connectivity can help us predict the functional state of the brain network. Therefore, we try to classify individual brain networks based on the connectivity information derived from the contrast subgraphs. Our aim is to evaluate the informativeness and the effectiveness of the contrast subgraphs extracted.

We will use Support Vector Machine (SVM) and Decision Tree for brain network classification. We choose these two classifiers because they have relatively few parameters and are more interpretable compared to other classification methods. In addition, for SVM, it finds an optimal boundary that best separates two groups of data and the weight parameter can be directly interpreted as the relative importance of individual contract graphs in terms of predicting the functional state of the brain. Similarly, for decision tree, the decision criteria at each decision point represents the determinants for the functional state of the brain.

We use design matrix from regression analysis as the input into the classification algorithms, where each row is one brain network sample, and each column corresponds to one of the feature vectors for classification. The number of features equals the number of contrast subgraphs extracted. For each contrast subgraph, we find their corresponding nodes in the individual brain networks. We then count the number of edges in the individual brain networks induced by this set of nodes as the value in the corresponding row and column, or in other words, the corresponding brain network and contrast subgraph. In general, we classify individual brain networks based on their connectivity density in each contrast subgraph.

For both approach, we randomly split all brain networks into 80/20 training/test sets. In the SVM approach, there is an additional step of hyperparameter tuning where we use 5-fold cross validation on the training set to select the best set of hyperparameters for the classifier. We then apply the trained classifiers to the test set and report the confusion matrix of the classification.

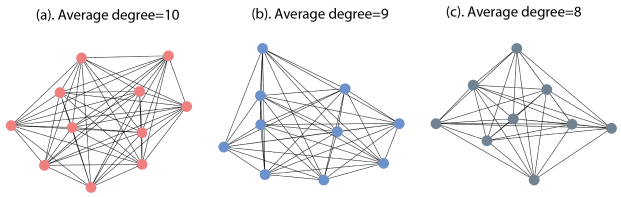

Figure 2. Densest subgraphs extracted from the difference graph. (a) the densest subgraph has an average degree of 10 (b) the second densest subgraph has an average degree of 9 (c) the third densest subgraph has an average degree of 8.

EXPERIMENT SETUP

Dataset

We use neuroimaging data from Autism Brain Imaging Data Exchange (ABIDE) project [3] for experimentation. It contains open-source neuroimaging data from 573 typically developed (TD) individuals and 539 patients who suffered from ASD. The original dataset was further divided into 4 nonexclusive subgroups: Children, Adolescents, EyesClosed, and Male. The number of TD subjects and number of ASD subjects were balanced in each group.

The raw dataset was preprocessed first by parcellating the brain into 116 Regions of Interest (ROIs). The time series data of each ROI reflecting its functional activity patterns was extract and the pairwise Pearson correlation coefficient was computed between any pair of ROIs, producing a 116x116 correlation matrix. This correlation matrix is binarized with a threshold equal to 80-th percentile of the distribution of all correlation coefficients, which is a common practice in literature [6]. After this step, every pair of nodes (u, v) with value 1 in the correlation matrix is considered as connected by an edge in the brain network, and every pair of nodes with value 0 in the correlation matrix is considered as disconnected. The brain network of each human subject is therefore converted to an undirected unweighted graph with 116 nodes. The thresholded correlation matrix is equivalent to the adjacency matrix of the graph.

Algorithm Application

The MinAndRemove algorithm and FastDSLO algorithm were implemented in Java originally.[1] To make use of the extensive open-source resources implemented in Python, we first convert the original algorithms to python, and apply it on the ABIDE dataset to assess its explainability and the classification accuracy.

PRELIMINARY RESULTS

We first applied the FastDSLO algorithm and extracted the top 3 densest subgraphs of the difference graph. The result is shown in figure 2. As a proof-of-concept, the FastDSLO algorithm extracts k-top subgraphs ranked by the average degree.

REFERENCES

[1] Oana Denisa Balalau, Francesco Bonchi, TH Hubert Chan, Francesco Gullo, and Mauro Sozio. 2015. Finding

subgraphs with maximum total density and limited overlap. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*. 379–388.

[2] Moses Charikar. 2000. Greedy Approximation Algorithms for Finding Dense Components in a Graph. *Approximation Algorithms for Combinatorial Optimization* (2000).

[3] Cameron Craddock, Yassine Benhajali, Carlton Chu, Francois Chouinard, Alan Evans, András Jakab, Budhachandra Singh Khundrakpam, John David Lewis, Qingyang Li, Michael Milham, and others. 2013. The neuro bureau preprocessing initiative: open sharing of preprocessed neuroimaging data and derivatives. *Frontiers in Neuroinformatics* 7 (2013).

[4] Tobias Kaufmann, Kristina C Skåtun, Dag Alnæs, Nhat Trung Doan, Eugene P Duff, Siren Tønnesen, Evangelos Roussos, Torill Ueland, Sofie R Aminoff, Trine V Lagerberg, and others. 2015. Disintegration of sensorimotor brain networks in schizophrenia. *Schizophrenia bulletin* 41, 6 (2015), 1326–1335.

[5] Tommaso Lanciano, Francesco Bonchi, and Aristides Gionis. 2020. Explainable Classification of Brain Networks via Contrast Subgraphs. (2020).

[6] Anton Lord, Dorothea Horn, Michael Breakspear, and Martin Walter. 2012. Changes in community structure of resting state functional connectivity in unipolar depression. (2012).

[7] Lu Meng and Jing Xiang. 2018. Brain network analysis and classification based on convolutional neural network. *Frontiers in computational neuroscience* 12 (2018), 95.

[8] Muhammad Faiz Misman, Azurah A. Samah, Farah Aqilah Ezudin, Hairuddin Abu Majid, Zuraini A. Shah, Haslina Hashim, and Muhamad Farhin Harun. 2019. Classification of Adults with Autism Spectrum Disorder using Deep Neural Network. *2019 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS)* (2019), 29–34.

[9] Mehdi Rahim, Bertrand Thirion, Claude Comtat, and Gaël Varoquaux. 2016. Transmodal learning of functional networks for Alzheimer's disease prediction. *IEEE journal of selected topics in signal processing* 10, 7 (2016), 1204–1213.

[10] Neil D Woodward and Carissa J Cascio. 2015. Resting-state functional connectivity in psychiatric disorders. *JAMA psychiatry* 72, 8 (2015), 743–744.