Graph Clustering Optimization

Samarth Bhatia

1 Background

Graph clustering partitions the nodes of a graph into $\mathbf{k}$ groups based on their specific attributes and interaction patterns. The objective while partitioning is to minimize the weights of edges connecting nodes of different clusters, i.e., the dissimilar nodes should be assigned to different clusters.

There is a plethora of applications of the graph clustering ranging from detection of terrorist groups in online social networks to single-cell RNA sequencing in bioinformatics and lung cancer detection in medicine.

1.1 Notations

Let $\mathbf{G} = \{\mathbf{V}, \mathbf{E}, \mathbf{W}, \mathbf{X}\}$ be a graph with node set $\mathbf{V} = \{\mathbf{v_1}, \mathbf{v_2}, ..., \mathbf{v_p}\}$ ($|\mathbf{V}| = \mathbf{p}$), edge set $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$ ($|\mathbf{E}| = \mathbf{e}$), weight (adjacency) matrix $\mathbf{W}$ and node feature matrix $\mathbf{X} \in \mathbb{R}^{p \times n}$.

1.2 Spectral Graph Clustering

Spectral Clustering is the most direct approach to graph clustering, where we minimizes the volume of inter-cluster edges. This is the same as the MinCut objective. An additional balancing term is included in the objective function to penalize trial solutions with disproportionate partition sizes.

Graph clustering quality functions

1.3 Modularity Maximization

Modularity matrix $\mathbf{B}$ of a weighted undirected graph was defined by Girvan and Newman [3]:

$$\mathbf{B} = \mathbf{W} - \frac{\mathbf{d}\mathbf{d^T}}{\mathbf{2e}} \tag{1}$$

where, $\mathbf{d}$ is the degree vector

$$\sum_{\mathbf{i=1}}^{\mathbf{p}} \mathbf{d_i} = \mathbf{2e}$$

1.4 Graph Neural Networks

GNNs are message passing networks with arbitrary structure. They have been applied to a range of applications, like social networks and computational chemistry. Unsupervised training in GNNs has been a topic of interest in the recent years, starting with the semi-supervised GCN [1] which dealt with node classification. More recent methods like DeepGraphInfomax [6] and InfoGraph [4] extend this to learning representations for graphs too, by maximizing the mutual information between suitable representations.

2 Related Works

2.1 Graph Clustering with GNNs

Deep Modularity Network (DMoN) [5] This is an unsupervised method for attributed graph clustering using GNNs. A soft and different cluster assignment matrix $\mathbf{C'}$ is learnt using a GCN trained using on normalized adjacency matrix $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$.

$$\mathbf{C'} = \mathbf{softmax}(\mathbf{GCN}(\tilde{\mathbf{A}}, \mathbf{X})) \tag{2}$$

The loss function proposed for optimizing the cluster assignments comprises of a spectral modularity maximization terms and a regularizer to avoid degenerate solutions. Collapse regularization is defined as the Frobenius norm of

the normalized cluster counts. It attains a maximum value, i.e. $\sqrt{k}$ when all the nodes are assigned to one cluster and reaches 0 for a perfectly balanced assignment.

$$\mathcal{L}_{\mathbf{DMoN}} = \frac{\mathbf{-1}}{\mathbf{2e}}\mathbf{Tr}(\mathbf{C'^T B C'}) + \frac{\sqrt{\mathbf{k}}}{\mathbf{p}}\left\|\sum_{\mathbf{i}}[\mathbf{C'^T}]_{\mathbf{i}}\right\| - \mathbf{1} \tag{3}$$

2.2 Graph Coarsening

Feature Graph Coarsening Algorithm [2] The FGC Algorithm is an iterative algorithm which uses block successive upper bound minimization (BSUM) to solve the following multi-block non-convex optimization problem:

$$\begin{aligned} \min_{\tilde{X},C} \quad & -\gamma \log\det(C^T\Theta C + J) + \mathrm{tr}(\tilde{X}^T C^T \Theta C \tilde{X}) + \frac{\alpha}{2}\left\|C\tilde{X} - X\right\|_F^2 + \frac{\lambda}{2}\left\|C^T\right\|_{1,2}^2 \\ \text{s.t.} \quad & \mathbf{C} \geq 0,\ \left\|[\mathbf{C}^T]_i\right\|_2^2 \leq 1,\ \forall i = 1,2,3,\ldots,p \end{aligned} \tag{4}$$

3 Method

3.1 Problem Formulation

The cluster assignment matrix $\mathbf{C'} \in \mathbb{R}^{p\times k}$ in DMoN [5] contains the probability distribution of each node belonging to the clusters. The row-wise argmax of $\mathbf{C'}$ is equivalent to the loading matrix $\mathbf{C}$ in the FGC Algorithm [2].

$$[\mathbf{C^T}]_{\mathbf{i}} = \mathbf{1}_{\mathbf{argmax}([\mathbf{C'^T}]_{\mathbf{i}})} \tag{5}$$

$$C_{i,j} = \begin{cases} 1 \text{ if } C'^T_{i,j} = \max_j C'^T_{i,j} \\ 0\ otherwise \end{cases} \tag{6}$$

We introduce the negative of modularity and collapse regularization terms to the minimization objective of the FGC algorithm.

$$\begin{aligned} \min_{\tilde{X},C} \quad & -\gamma \log\det(C^T\Theta C + J) + \mathrm{tr}(\tilde{X}^T C^T \Theta C \tilde{X}) + \frac{\alpha}{2}\left\|C\tilde{X} - X\right\|_F^2 + \frac{\lambda}{2}\left\|C^T\right\|_{1,2}^2 + \frac{-\beta}{2e}Tr(C^T B C) + \delta(\frac{\sqrt{k}}{p}\left\|\sum_i [C^T]_i\right\|_2 - 1) \\ \text{s.t.} \quad & \mathbf{C} \geq 0,\ \left\|[\mathbf{C}^T]_i\right\|_2^2 \leq 1,\ \forall i = 1,2,3,\ldots,p \end{aligned} \tag{7}$$

3.2 Metrics

We evaluate the performance of our method using both label alignment metrics which relate ground truth node labels to cluster assignments and graph structure metrics used for clustering.

Graph-based metrics used are modularity ($\mathcal{Q}$) and conductance ($\mathcal{C}$). Modularity is the difference the number of edges between nodes in cluster $\mathbf{C_i}$ and the expected number of such edges in a random graph with identical degree sequence. For a good clustering algorithm a higher value of modularity is desired.

$$\mathcal{Q} = \frac{1}{2e}Tr(C^T B C) \tag{8}$$

Conductance measures the fraction of total edge volume that points outside the cluster. $\mathcal{C}$ is the average conductance across all the clusters and a lower value is desirable.

We report the Normalized Mutual Information (NMI) between the cluster allotment and node labels and the F1 score between all pairs of nodes and their corresponding clusters pairs. Higher values of these metrics indicates better clustering.

3.3 Modularity and Collapse Regularization terms

Since FGC [2] uses a block successive upper bound minimization (BSUM) technique, which involves iterating while first updating C keeping X constant and then updating X keeping C constant, we have added our modularity and collapse regularization terms to the update of C (since they are not dependent on X). More specifically, the derivative of those terms w.r.t C is added:

For the modularity term $\frac{-\beta}{2e}Tr(C^T B C)$, we add

$$\frac{-\beta}{2e}(B^T + B)C$$

. For the collapse regularization term $\delta(\frac{\sqrt{k}}{p}\left\|\sum_i [C^T]_i\right\|_2 - 1)$, we add

$$\delta\frac{\sqrt{k}}{p}\frac{\sum_i [C^T]_i}{\|\sum_i [C^T]_i\|_2}$$

.

4 Results

We tested our method on the Cora dataset which consists of 2708 nodes distributed across 7 classes containing 1433 feature attributes.

Method	$\mathcal{Q}\uparrow$	**NMI** $\uparrow$	$\mathcal{C}\downarrow$	**F1 Score**	β	δ
FGC	23.4	18.6	56.5	33.6	0	0
FGC + Modularity	69.8	26.3	15.3	32.0	100	0
FGC + Modularity + C.R.	70.4	28.6	14.6	36.1	100	10
DMoN	76.5	48.8	12.2	48.8	-	-

Table 1: Initial Comparison of DMoN against FGC with Modularity and Collapse Regularization terms added.

From our experiments, it's clear that simply adding modularity terms to the objective significantly improved the performance of FGC Algorithm. Adding collapse regularization with modularity did not improve the performance by much except for an increase in the (pairwise) F1 score.

References

[1] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. arXiv, 2016.

[2] Manoj Kumar, Anurag Sharma, and Sandeep Kumar. A unified framework for optimization-based graph coarsening, 2022.

[3] Mark E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103 23:8577–82, 2006.

[4] Fan-Yun Sun, Jordan Hoffman, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *International Conference on Learning Representations*, 2019.

[5] Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. Graph clustering with graph neural networks, 2020.

[6] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep Graph Infomax. In *International Conference on Learning Representations*, 2019.