Revisiting Modularity Loss with Feature Information for a Better Graph Clustering?

Shashwat Kumar Panigrahi
National Institute of Technology, Rourkela
Rourkela, India
shashwat.boudh@gmail.com

Panthadeep Bhattacharjee
National Institute of Technology, Rourkela
Rourkela, India
panthadeep.edu@gmail.com

ABSTRACT

Graph Neural Networks (GNNs) are a widely used architecture for clustering tasks on attributed graphs. A common loss function used to train such networks is modularity (Q), a topology-based measure for cluster quality. However, the modularity loss used in the existing works only considers the graph structure, overlooking node attributes. This may reduce the quality of cluster assignment on attributed graphs. To address this limitation, we propose an extended measure, called Attributed-Modularity (Q_{attr}), that integrates both structural information and node features. Experimental results show that Q_{attr} consistently outperforms standard modularity and its existing variants in clustering performance.

1 INTRODUCTION

Graph clustering is an important problem in graph analysis. It helps reveal hidden patterns and discover relationships between nodes. As graphs are mainly defined by their topology and node attributes, effective integration of these two in learning node representation are vitally important for getting quality clusters. Graph neural networks (GNNs), aided by their message passing mechanism, can effectively integrate graph structure with its node attributes Thus, GNNs can learn node representations suitable for graph clustering.

To effectively guide the training of GNNs, Modularity Maximization has become a widely used objective function. Modularity quantifies how well connected the nodes within the same cluster are, compared to what we would expected by chance. Existing methods which used traditional modularity maximization only considers graph structure for estimating the cluster quality and doesn't take node attribute information into account. This may lead to incomplete estimation of cluster quality, which may misguide the training of GNN, resulting in suboptimal cluster estimation. Moreover, some of the existing methods also have the limitation that, they need the number of clusters to be given as input, which may not be always available.

To address this issue, we propose Attr-MODCluster, which is an end-to-end fully differentiable graph clustering network. The major contribution are a) **Attributed-Modularity Loss:** We propose a extented measure of modularity, where we have effectively incorporated graph structure with node attributes for modularity estimation. b) **Handling unknown number of cluster:** Our proposed methodology can automatically handle data with unknown number of clusters.

2 METHODOLOGY

We present Attr-MODCluster, an end-to-end fully differentiable deep clustering architecture. It consists of three components: a) GNN module b) attributed-modularity loss module.

GNN Module: In this paper we have use two different Graph neural networks (GNNs), based on the size of the dataset (Tab. 2). For smaller datasets (cora, citeseer, photo), we use Graph convolutional network (GCN) with full graph training and for larger datasets (Computers, Co-author CS and PubMed) we have used GraphSAGE with mini-batch training. We have tried to keep the network design simple by taking only two layers of GNN followed consisting of a leaky ReLU activation along with a dropout layer. The final output of the model is soft cluster assignment obtained through a softmax function. The use of softmax makes the cluster assignment differentiable.

Attributed-modularity loss module: Modularity(Q) in its traditional sense is defined as:

$$Q = -\frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{d_i d_j}{2m}\right)\delta(c_i, c_j) \tag{1}$$

where A_{ij} is the adjacency matrix, $\frac{d_i d_j}{2m}$ is the expected number of edges in a random graph and $\delta(c_i, c_j)$ is the Kronecker delta, which is 1 if node i and j belong to the same cluster and 0 otherwise. $c_i \in \{0, 1\}$ is the cluster to which node i is assigned. Due to the discrete nature of $\delta(c_i, c_j)$, maximizing Q is NP-Hard. However, for practical use cases, certain spectral relaxation methods offers efficient solutions:

$$Q = -\frac{1}{2m}Tr(C^T BC) \tag{2}$$

where $C \in$ R is the soft cluster assignment matrix and $B = \sum_{ij}\left(A_{ij} - \frac{d_i d_j}{2m}\right)$ is the modularity matrix.

Attributed-modularity loss formulation The limitation of traditional modularity (Eqn. 1), as explained in section 1 can be addressed by incorporating node attribute information into the modularity equation as follows:

Let x_i and x_j be the attributes values of node i and j. We define the attribute similarity matrix as the cosine similarity between all node pairs:

$$W_{i,j} = max(0, \frac{x_i^T x_j}{|x_i||x_j|}) \tag{3}$$

The strength vector of the graph for each node is given as: $S_i = \sum_j W_{ij}$ and total similarity strength of the graph as: $2w = \sum_i S_i$. Using these above results and attribute similarity matrix in Eqn. 3, we propose the attribute modularity as:

$$Q_{attr} = -\frac{1}{2m}\sum_{ij}\left(W_{ij} - \frac{S_i^T S_i}{2w}\right)\delta(c_i, c_j) \tag{4}$$

which can be rewritten as:

$$Q_{attr} = -\frac{1}{2m}tr(C^T B_{attr} C) \tag{5}$$

Loss Fn	Datasets																			
	Cora				Citeseer				Photo				Computers				PubMed			
	Acc	NMI	ARI	F1	Acc	NMI	ARI	F1	Acc	NMI	ARI	F1	Acc	NMI	ARI	F1	Acc	NMI	ARI	F1
Q_{attr}	0.59	0.48	0.37	0.54	0.52	0.29	0.28	0.42	0.60	0.55	0.46	0.43	0.36	0.10	0.05	0.13	0.60	0.21	0.18	0.60
Q	0.59	0.48	0.37	0.54	0.52	0.29	0.28	0.42	0.60	0.55	0.46	0.43	0.36	0.10	0.05	0.13	0.60	0.21	0.18	0.60

Table 1: Comparison of clustering metrics across datasets with different loss functions.

Table 2: Dataset statistics

Dataset	#Nodes	#Edges	#Features	#Clusters
Cora	2,708	5,278	1,433	7
CiteSeer	3,327	4,614	3,703	6
Photo	7,650	119,081	745	8
Computers	13,752	491,722	767	10
PubMed	19,717	44,324	500	3

The final loss is formulated as a linear combination of the two losses in addition to two regularization terms used to mitigate for cluster collapse.

$$Q_{total} = \alpha * Q + (1 - \alpha) * Q_{attr} + \beta * Col_r - \gamma * Ent_r \quad (6)$$

Where $Col_r = \frac{\sqrt{k}}{n} |\sum_i C_i - 1|$ is the collapse regularization term and $Ent_r = \frac{1}{n} \sum_I C_i \log(C_i + \epsilon)$ is the entropy-based regularization term.

3 EXPERIMENTS

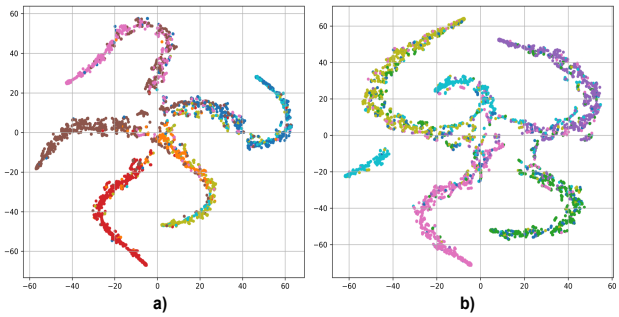

Figure 1: TSNE visualization of learned node representation for a) Cora and b) Citeseer datasets

4 CONCLUSION

5 GENAI USAGE DISCLOSURE

No GenAI tools were used to create the draft. The editing and subsequent revisions were conducted by the authors of this paper.

REFERENCES