Student's Declaration

I hereby declare that the work presented in the report entitled **"...(title)..."** submitted by me for the partial fulfillment of the requirements for the degree of *Bachelor of Technology* in *Computer Science & Engineering* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of my work carried out under guidance of **....(advisors' name).....**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

..... **Place & Date: .....**
...(student's name)...

Certificate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

..... **Place & Date: .....**
..(advisors' name)...

Abstract

In this project I study the problem of identifying critical nodes in a network which minimize different community metrics like NMI(Normalized Mutual Information), Modularity, Conductance etc. and compare their results. Currently I focus on Modularity metric of communities in a social network. The ability to see such results allows us to better understand the vulnerability of the network since the nodes selected across different metrics might be different and hence seeing those results in conjunction will tell us a lot more about the network, than seeing only one metric which is what is done in some literature.

Keywords: (Graph Mining, Algorithms, Community Analysis)

Acknowledgments

XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX
XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX
XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX
XXXXXX XXXXXX XXXXXX XXXXXX XXXXXX

Work Distribution

Contents

Chapter 1

Introduction

1.1 Problem Definition

Given a graph $G(V, E)$, each node i is associated with a cost C_i, and there is a total budget K. For a vertex set $S \subseteq V$, let $G[S]$ be the graph induced by S, $\textsc{Cost}(S) = \sum_{i \in S} C_i$, and $f(S) = \textsc{Modularity}(G) - \textsc{Modularity}(G[\bar{S}])$. Here, $\bar{S} = V \setminus S$. Note that, $f(S)$ is a real-valued *set function*. We need to identify a set of nodes $S \subseteq G$ which,

$$\begin{aligned} \textbf{maximize} \quad & f(S) \\ \textbf{subject to:} \quad & \textsc{Cost}(S) \leq K \end{aligned}$$

Modularity is defined as follows:

$$\frac{1}{2m} \sum_{i,j} (A_{i,j} - \frac{k_i * k_j}{2m}) \delta(c_i, c_j)$$

where $\boldsymbol{m}$ = Number of edges

$\boldsymbol{k_i, k_j}$ = Degree of node i, j

$\boldsymbol{c_i, c_j}$ = Community label of node i and node j

$\boldsymbol{A_{i,j}}$ = Adjacency Matrix

$$\boldsymbol{\delta(c_i, c_j)} = \begin{cases} 1 & c_i = c_j \\ 0 & otherwise \end{cases}$$

The unit cost formulation of the above problem has only one change. The value of $C_i = 1$ i.e each node in the graph G has unit cost. We are going to prove that this version of the above mentioned problem is NP-Complete. Since the unit cost formulation is just a trivial case of the general cost version, if we prove that the unit cost version is NP-Complete then clearly the general case would also be NP-Complete.

Chapter 2

NP-Completeness Proof

2.1 Reduction Problem

We will use *Maximum Vertex Coverage - Bipartite Graphs*(MVC-B) for reduction. This problem states the following,

Given a bipartite graph $G = (V, E), V = V_L \cup V_R, V_L \cap V_R = \emptyset$ and positive integers b, c, check if there exists a subset of vertices $S \subseteq V with$ |S|= b such that at least c edges are incident to nodes in S.

2.2 Proof

Given an instance of MVC-B with bipartite graph $G = (V, E)$, $V = V_L \cup V_R, V_L \cap V_R = \emptyset$ and positive integers b, c.
Now we construct our problem instance $G' = (V', E')$ by connecting V_L, V_R to cliques K_L, K_R, respectively. We create an edge between all $\{(u, v)|u \in K_L, v \in V_L\}$ and all $\{(u, v)|u \in K_R, v \in V_R\}$. We choose the size of K_R, K_L such that, when b vertices in V are removed with at least c edges incident to them in E, $A(Algorithm)$ will detect two communities $K_L \cup V_L \setminus S_L, K_R \cup V_R \setminus S_R$ where $S_L \in V_L$ and $S_R \in V_R$ and when we do not do any such operation it will detect only one community. By this we mean that $\mid K_L \mid$ is not necessarily equal to $\mid V_L \mid$ and similarly $\mid K_R \mid$ is not necessarily equal to $\mid V_R \mid$. Suppose when E is sparse then we would want $\mid K_L \mid$ and $\mid K_R \mid$ to be small enough so that $A(G')$ will detect only one community and $A(G'[V' \setminus S]$ will detect 2 above mentioned communities when we remove $\mid S \mid = b$ vertices. Similarly when E is dense we would want K_L and K_R to be large enough for the same reason.

Let $k = b$ and,

$$a = \min_{S_L \in V_L, S_R \in V_R, |S_L|+|S_R|=b} Modularity(A(G')) - Modularity(Y')$$

where

$$Y' = \{K_L \cup V_L \setminus S_L, K_R \cup V_R \setminus S_R\}$$

Now assume we have a solution S, $\mid S \mid = K$ to our problem. As

$$Modularity(A(G') - Modularity(A(G'[V' \setminus S])) \geq a$$

, $A(G'[V' \setminus S])$ must contain two communities. By construction, number of edges removed from E is at least c. If $S \subseteq V$ then we directly obtain the solution to MVC-B. If $\exists v \in S, v \notin V$, we can always find a vertex $u \in V, u \notin S$ and update S to S' = $S \cup \{u\} \setminus \{v\}$ while keeping the number of edges incident to S greater than c. Therefore, we have a solution $S', \mid S' \mid$=K= b for MVC-B.

Now assume we have a solution $S, \mid S \mid = b$ to MVC-B. Then at least c edges in E are incident to vertices in S. If we remove all vertices in S, by construction, $A(G'[V' \setminus S)$ will output 2 communities, $K_L \cup V_L \setminus S'_L$ and $K_R \cup V_R \setminus S'_R$. Then we have,

$$Modularity(A(G')) - Modularity(A(G'[V' \setminus S])) \geq a$$

as a is the maximum Modularity difference value for communities in the form $\{K_L \cup V_L \setminus S_L, K_R \cup V_R \setminus S_R\}$. Therefore, we have a solution $S, \mid S \mid = b = K$ for our problem.

2.3 Conclusion

Since our problem statement has a solution if and only if *Maximum Vertex coverage - Bipartite Graph* has a solution, hence our problem statement is NP-Complete.

Chapter 3

Heuristics

3.1 Get baseline results

- Use UCI Karate Club network with 2 communities and 34 nodes.
- ${}^nP_r$ combinations $\rightarrow$ Try all combinations by removing one by one. This will give the exhaustive list of vertices k.

3.2 An iterative greedy algorithm to maximize $f(S)$

- $S \leftarrow \emptyset, \bar{S} \leftarrow V$.
- **for** $v \in \bar{S}, S_v \leftarrow S \cup \{v\}$.
- Find the vertex v^* for which $\textsc{Cost}(S_{v^*}) \leq K$ and $f(S_{v^*})$ is maximized.
- **if** $f(S_{v^*}) \geq f(S)$ **then** $S \leftarrow S \cup \{v^*\}, \bar{S} \leftarrow \bar{S} \setminus \{v^*\}$.
- **else** return S.

Bibliography

1. Structural Vulnerability Assessment of Community-Based Routing in Opportunistic Networks - My T. Thai
2. The Budgeted Maximum Coverage Problem - Samir Khullar
3. On the Permanence of Vertices in Network Communities - Tanmoy Chakraborty
4. Detecting Critical Nodes in Interdependent Power Networks for Vulnerability Assessment - My T. Thai
5. Community Detection in Scale-free Networks: Approximation Algorithms for Maximizing Modularity - My T. Thai
6. Scribe - Animesh Mukherjee