

Implementation of Local Higher-Order Graph Clustering

Term Paper

Anjali M(2020H1030116H)

Patel Krupal Rajeshkumar(2020H1030117H)

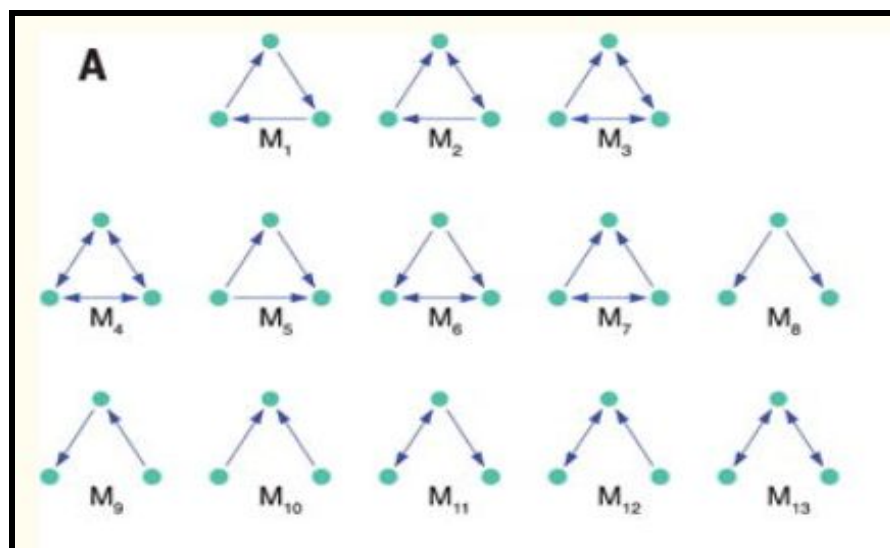
Clusters or community structures are considered to be a significant property of real world networks. graph clustering refers to clustering of data in the form of graphs. Despite the ambiguity in the definitions of the community and clusters of graph structures, numerous techniques are developed for both efficient and effective graph clustering. Quite recently, there has been a growing interest in finding communities by locally expanding a good seed set in the neighborhood of interest. Most of the current local graph clustering methods do not focus on the higher-order structures. The motif-based Approximate personalized pagerank algorithms gives experimental validation on community detection tasks.

In this project, we implemented “Local Higher-Order Graph Clustering Algorithm”. This paper aims to develop a local algorithm for finding clusters of nodes based on higher-order network structures (also called network motifs).

Unlike global clustering, this algorithm targets community detection by only exploring the local neighborhood of seed nodes

Existing local clustering methods find clusters with many internal edges and few external edges. However, edges are not the only interesting structures in networks! Employing motifs may give more insight into the targeted communities in the graph network.

Higher-order connectivity patterns, or network motifs, mediate complex networks.



NOTE : Figure shows some of the possible motifs, but in our implementation, we tried to implement motifs- M1 to M7.

This algorithm searches for a cluster (a set of nodes) ‘S’ with minimal motif conductance, which is a cluster quality score designed to incorporate the higher-order structure and handle directional edges. More precisely, given a graph G and a motif M, the algorithm aims to find a set of nodes ‘S’ with a good motif conductance (for a given motif M) such that S contains a given seed node.

$$\phi_M(S) = \frac{\text{cut}_M(S)}{\min(\text{vol}_M(S), \text{vol}_M(\bar{S}))}.$$

Motif conductance has the benefit that it allows us to focus the clustering on a particular network substructures that are important for networks of a given domain.

For example, triangles are important higher-order structures of social networks and thus focusing the clustering on such substructures can give important information.

Cluster S is said to have good (low) motif conductance for some motif M if the nodes in S participate in many instances of M and there are few instances of M that cross the set boundary defined by S.

Our main approach is to generalize Approximate Personalized PageRank (APPR).

The APPR method is a graph diffusion method that “spreads” from a seed set to identify the cluster. It has an extremely fast running time, which is roughly proportional to the size of the output cluster.

Our generalization, the motif-based APPR method, or MAPPR, employs a pre-processing step that transforms the original graph into a weighted undirected graph where the weights depend on the motif of interest.

The MAPPR method handles directed graphs on which graph clustering has been a longstanding challenge.

Algorithm(MAPPR)

Input: a Graph network, a seed node, and a motif.

Output: a cluster containing the seed node with minimal motif conductance.

Algorithm 1: Motif-PageRank-Nibble method for finding localized clusters with small motif conductance.

Input: Unweighted graph $G = (V, E)$, motif M , seed node u , teleportation parameter α , tolerance ε

Output: Motif-based cluster (set $S \subset V$)

- 1 $W_{ij} \leftarrow \#(\text{instances of } M \text{ containing nodes } i \text{ and } j)$
 - 2 $\tilde{p} \leftarrow \text{Approximate-Weighted-PPR}(W, u, \alpha, \varepsilon)$ (Algorithm 2)
 - 3 $D_W \leftarrow \text{diag}(We)$
 - 4 $\sigma_i \leftarrow i\text{th smallest entry of } D_W^{-1}\tilde{p}$
 - 5 **return** $S \leftarrow \arg \min_{\ell} \phi_M(S_{\ell})$, where $S_{\ell} = \{\sigma_1, \dots, \sigma_{\ell}\}$
-

Algorithm 2: Approximate-Weighted-PPR

Input: Undirected edge-weighted graph $G_w = (V_w, E_w, W)$, seed node u , teleportation parameter α , tolerance ε

Output: an ε -approximate weighted PPR vector \tilde{p}

- 1 $\tilde{p}(v) \leftarrow 0$ for all vertices v
 - 2 $r(u) \leftarrow 1$ and $r(v) \leftarrow 0$ for all vertices v except u
 - 3 $d_w(v) \leftarrow \sum_{e \in E_w: v \in e} W(v)$
 - 4 **while** $r(v)/d_w(v) \geq \varepsilon$ for some node $v \in V_w$ **do**
 - 5 /* push operation */
 - 6 $\rho \leftarrow r(v) - \frac{\varepsilon}{2}d_w(v)$; $\tilde{p}(v) \leftarrow \tilde{p}(v) + (1 - \alpha)\rho$; $r(v) \leftarrow \frac{\varepsilon}{2}d_w(v)$
 - 7 **for each** $x : (v, x) \in E_w$ **do** $r(x) \leftarrow r(x) + \frac{W(v, x)}{d_w(v)} \cdot \alpha\rho$
 - 8 **return** \tilde{p}
-

Matrix-based formulations of the weighted motif adjacency matrix W_M (Equation S19) for all triangular three-node simple motifs. $(P \circ Q)$ denotes the Hadamard (entry-wise) products of matrices P and Q . If A is the adjacency matrix of a directed, unweighted graph G , then compute, $B = (A \circ A^T)$ and $U = (A - B)$. Note that in all cases, W_M is symmetric.

Motif	Matrix computations	$W_M =$
M_1	$C = (U \cdot U) \circ U^T$	$C + C^T$
M_2	$C = (B \cdot U) \circ U^T + (U \cdot B) \circ U^T + (U \cdot U) \circ B$	$C + C^T$
M_3	$C = (B \cdot B) \circ U + (B \cdot U) \circ B + (U \cdot B) \circ B$	$C + C^T$
M_4	$C = (B \cdot B) \circ B$	C
M_5	$C = (U \cdot U) \circ U + (U \cdot U^T) \circ U + (U^T \cdot U) \circ U$	$C + C^T$
M_6	$C = (U \cdot B) \circ U + (B \cdot U^T) \circ U^T + (U^T \cdot U) \circ B$	C
M_7	$C = (U^T \cdot B) \circ U^T + (B \cdot U) \circ U + (U \cdot U^T) \circ B$	C

KEY STEPS:

1. Create a weighted graph as follows

$$W_{ij} = \text{\#motif instances containing nodes } i \text{ and } j.$$

The motif conductance (approximately) equals the weighted edge conductance in this weighted graph [Benson et al., 16].

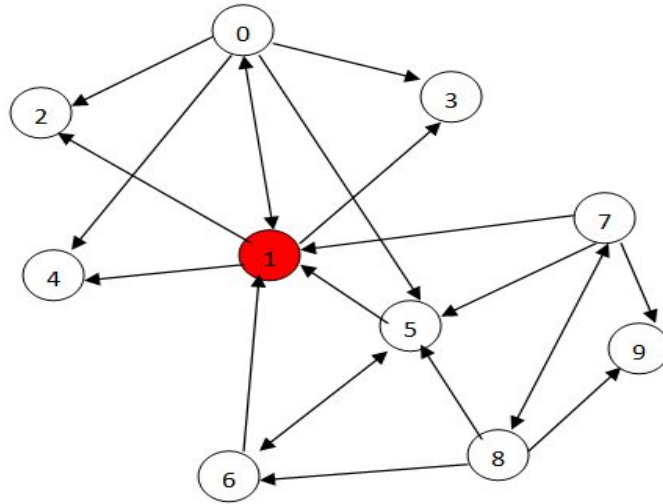
2. Compute an approximate PPR vector for this weighted graph.

- The PPR vector p is the stationary distribution of a random walk which at each step it “teleports” back to the seed with some probability.
- $p(u)$ measures an “integrated closeness” of node ‘ u ’ to the seed.
- On a weighted graph, we choose each edge with probability proportional to its weight.
- We adapted the approximate PPR algorithm [Anderson et al., 06] for weighted graphs.

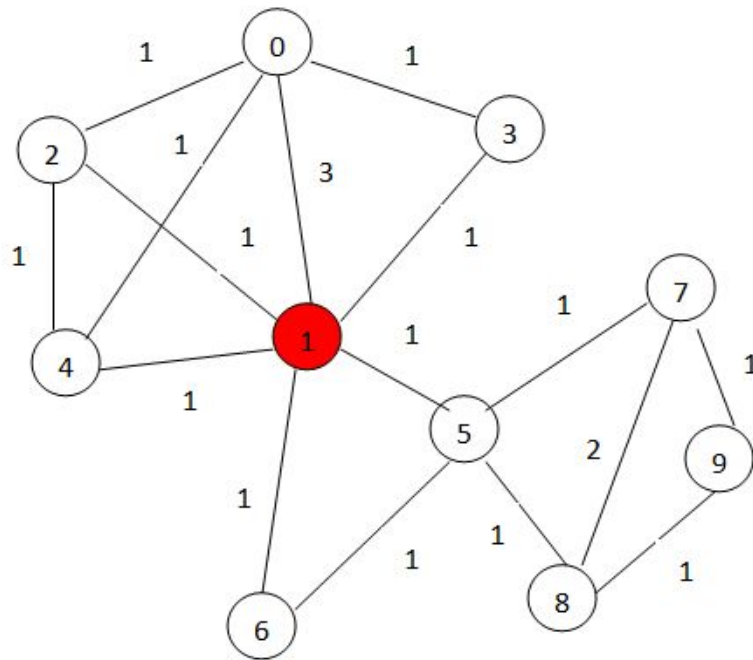
3. Use the sweep procedure on APPR vector ‘ p ’ to output the set with minimal weighted edge conductance [Anderson et al., 06].

- Sort nodes by $p(u)/d_w(u) : \{u_1, u_2, \dots, u_{[p]}\}$;
- Compute the conductance of each $S_r = \{u_1, u_2, \dots, u_r\}$
- Output the SN with minimal weighted edge conductance.

Consider, the following directed graph (G) :



After finding the corresponding Weighted Graph for the above graph, based on the chosen Motif(here, M7), we get the following :



For the given graph and Motif-M7, our implementation gives the following result.

```

Problems @ Javadoc Declaration Console
<terminated> ClusteringAlgComparision [Java Application] C:\Program Files\AdoptOpenJDK\jdk8u202-b0
Enter the Motif M(1-7) :
7
The local motif Cluster (for the seed node(u :1), Motif (M7)) :
1 0 4 2 3 6 5
Execution Time for Local Motif7 : 56.0 milli seconds

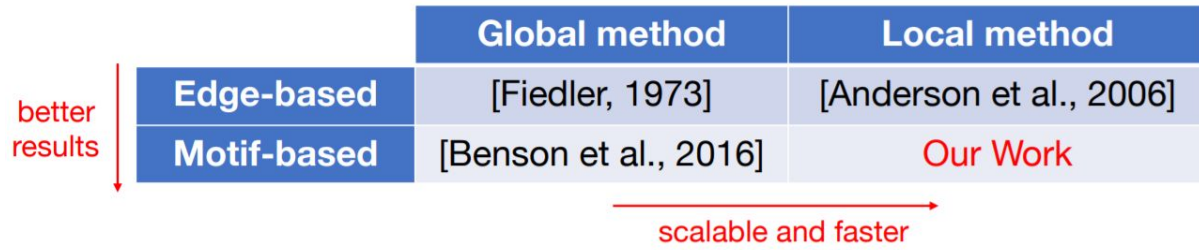
```

We can see that local cluster nodes around the seed node '1' are taken based on the structure of Motif (M7).

COMPARISON WITH OTHER ALGORITHMS:

The Paper promises the following. In our implementation, we verified these claims.

	Global method	Local method
Edge-based	[Fiedler, 1973]	[Anderson et al., 2006]
Motif-based	[Benson et al., 2016]	Our Work

better results 

COMPARING WITH A GLOBAL MOTIF CLUSTERING ALGORITHM.

We simultaneously implemented “Higher-order organization of complex networks” which is a global clustering approach.

The main difference between our local and global algorithm implementation is that, rather than using the global minimum in the sweep procedure in the last step of the Nibble method, we apply the common heuristic of finding the first local minimum. The first local minimum is the smallest set where the PageRank vector suggests a border between the seed and the rest of the graph. It also gives a better model for the small size scale of most ground truth communities that we encounter in our experiments.

This is why the global clustering approach may take more time compared to the local algorithm.

NOTE: Finding the local minimum is referred from “Defining and evaluating network communities based on ground-truth. Knowledge and Information Systems” paper.

Basically, in the local motif clustering approach, after finding the *PageRank* Vector using PageRankNibble random walk method with error $< \epsilon$ in time $O(1/\epsilon)$ independent of the network size.

We detect the local minima of *PageRank* Vector using the following heuristic.

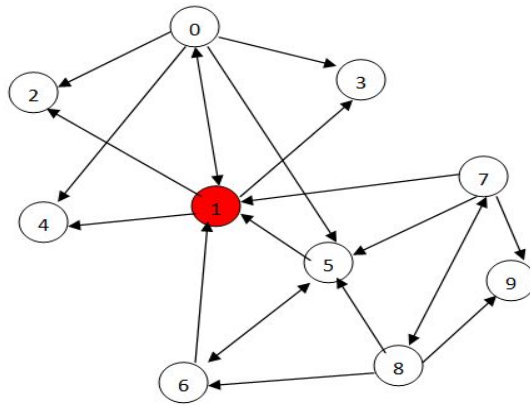
For increasing $k = 1, 2, \dots$, we measure $f(S_k)$.

At some point, $f(S_k)$ will stop decreasing and this k^* becomes our “candidate point” for a local minimum. If $f(S_k)$ keeps increasing after k^* and eventually becomes higher than $\alpha f(S_{k^*})$, we take k^* as a valid local minimum. However, if $f(S_k)$ goes down again before it reaches $\alpha f(S_{k^*})$, we discard the candidate k^* .

When the tolerance with which the random walk reaches back to the seed node is, $\alpha = 1.2$, it gives good results across all the datasets. This promises better results compared to global approach.

RESULTS :

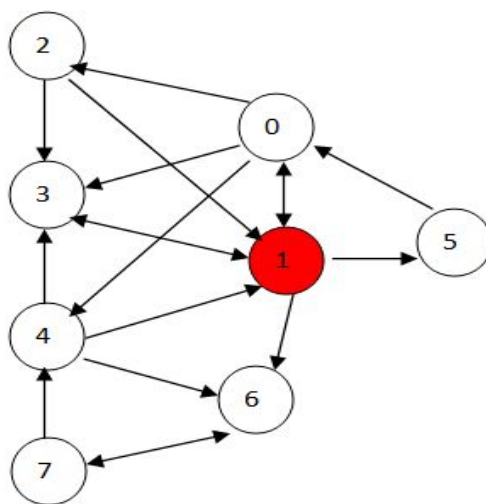
1. Consider the following graph :



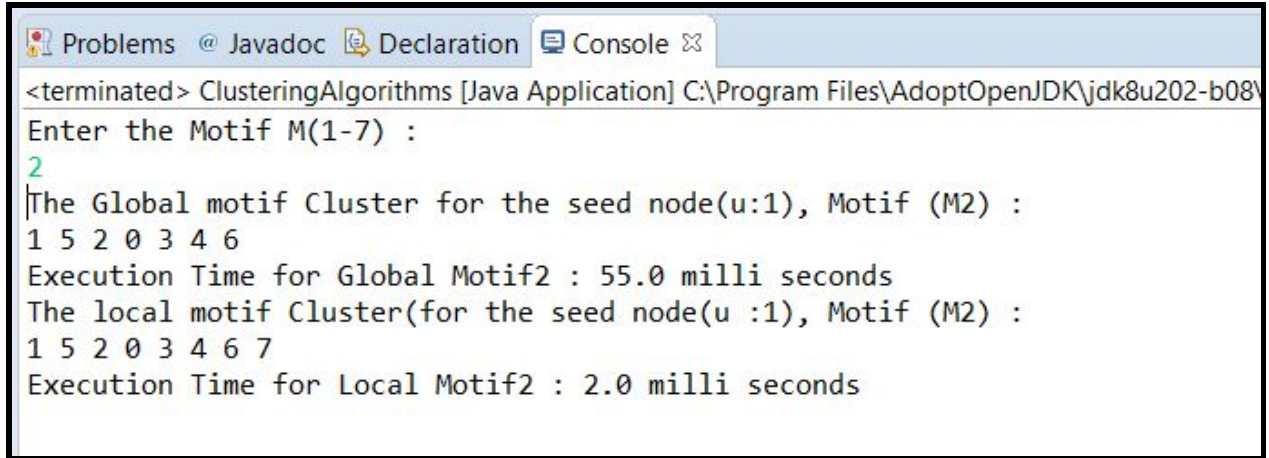
For the above graph G, and Motif (M7) and SeedNode(1) :

```
Problems @ Javadoc Declaration Console
<terminated> ClusteringAlgorithms [Java Application] C:\Program Files\AdoptOpenJDK\jdk8u202-b08
Enter the Motif M(1-7) :
7
The Global motif Cluster for the seed node(u:1), Motif (M7) :
1 0 4 2 3 6 5 8 7
Execution Time for Global Motif7 : 59.0 milli seconds
The local motif Cluster(for the seed node(u :1), Motif (M7) :
1 0 4 2 3 6 5
Execution Time for Local Motif7 : 5.0 milli seconds
```

2. Consider the graph:



For the above graph G, and Motif (M2) and SeedNode(1) :



```
<terminated> ClusteringAlgorithms [Java Application] C:\Program Files\AdoptOpenJDK\jdk8u202-b08\
Enter the Motif M(1-7) :
2
The Global motif Cluster for the seed node(u:1), Motif (M2) :
1 5 2 0 3 4 6
Execution Time for Global Motif2 : 55.0 milli seconds
The local motif Cluster(for the seed node(u :1), Motif (M2) :
1 5 2 0 3 4 6 7
Execution Time for Local Motif2 : 2.0 milli seconds
```

In both the scenarios, we can clearly observe that the Local-Motif clustering algorithm is more efficient than the global motif clustering approach.

COMPARING WITH A LOCAL EDGE-BASED CLUSTERING ALGORITHM.

Along with the “Local Higher-order organization of complex networks” and “Higher-order organization of complex networks”, we simultaneously implemented “Local Graph Partitioning using PageRank Vectors” which is a local edge-based clustering approach.

This edge-based approach focuses only on finding the cluster with low edge-conductance.

$$\Phi(S) = (\text{No. of edges cut by the cluster}) / \text{Volume of the cluster.}$$

$$\text{Vol}(S) = \text{No. of edge end points in } S$$

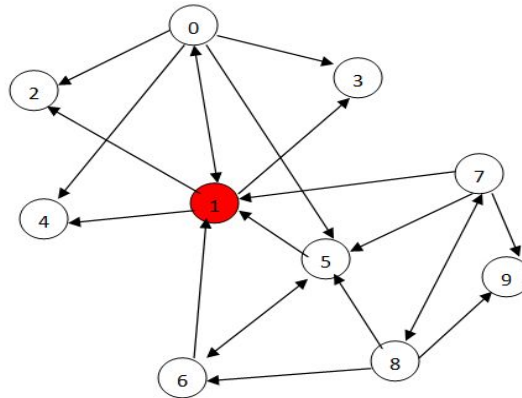
$$= \sum_{u \in S} \text{degree}(u)$$

Since it is a lower order approach, it definitely takes lesser time compared to the Motif-based higher order algorithms.

That is shown in the results below.

RESULTS

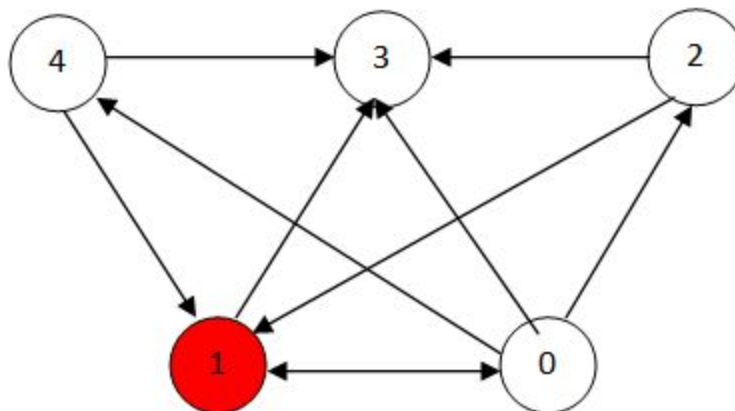
1.



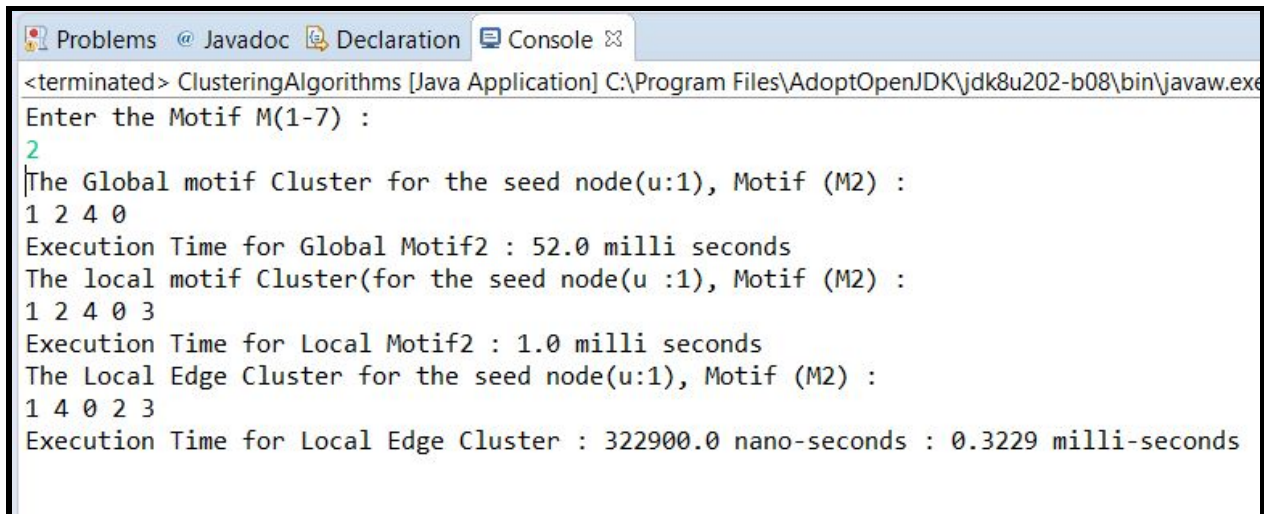
For the above graph G, and Motif (M7) and SeedNode(1)

```
Problems @ Javadoc Declaration Console
<terminated> ClusteringAlgorithms [Java Application] C:\Program Files\AdoptOpenJDK\jdk8u202-b08\bin\javaw.exe
Enter the Motif M(1-7) :
7
The Global motif Cluster for the seed node(u:1), Motif (M7) :
1 0 4 2 3 6 5 8 7
Execution Time for Global Motif7 : 56.0 milli seconds
The local motif Cluster(for the seed node(u :1), Motif (M7) :
1 0 4 2 3 6 5
Execution Time for Local Motif7 : 5.0 milli seconds
The Local Edge Cluster for the seed node(u:1), Motif (M7) :
1 4 2
Execution Time for Local Edge Cluster : 240200.0 nano-seconds : 0.2402 milli-seconds
```

2.Consider the following graph:



For the above graph G, and Motif (M2) and SeedNode(1)



```
<terminated> ClusteringAlgorithms [Java Application] C:\Program Files\AdoptOpenJDK\jdk8u202-b08\bin\javaw.exe
Enter the Motif M(1-7) :
2
The Global motif Cluster for the seed node(u:1), Motif (M2) :
1 2 4 0
Execution Time for Global Motif2 : 52.0 milli seconds
The local motif Cluster(for the seed node(u :1), Motif (M2) :
1 2 4 0 3
Execution Time for Local Motif2 : 1.0 milli seconds
The Local Edge Cluster for the seed node(u:1), Motif (M2) :
1 4 0 2 3
Execution Time for Local Edge Cluster : 322900.0 nano-seconds : 0.3229 milli-seconds
```

From the above results we can clearly observe the efficiency of each of the mentioned algorithms.

Also, we can see that Local Edge-based clustering algorithm is faster than Local motif-based clustering algorithm. But, we should keep in mind that motifs give more insight into the targeted communities in the graph network.

PROPERTIES :

Runtime guarantee

Since, the procedure stops upon finding a good cluster, no need to explore the rest of the graph.

Quality guarantee

Finds a near-optimal cluster regarding motif conductance.

Implementation Prerequisites -

1. Since, algorithm requires computing the inverse of a matrix, and conventional approach takes a huge computing time for the same, we imported Apache-Commons/Math Jar File in the Project.

2. The following are set by default in our code:

α:alpha (default:'0.98') //Teleportation Coefficient

ε:epsilon (default:'0.0001') //Tolerance

u:SeedNode (default: 1)

3. Please note that the seed node must be given relative to the Motif structure being employed to get better results. <Our implementation gives error when the seed node is not in chosen motif type>

NOTE : Please read the README.txt in the code folder.

REFERENCES

- 1.R. Andersen, F. Chung, and K. Lang. Local graph partitioning using pagerank vectors. In FOCS, 2006.
- 2.A. R. Benson, D. F. Gleich, and J. Leskovec. Higher-order organization of complex networks. Science, 2016.
- 3.J. Yang and J. Leskovec. Defining and evaluating network communities based on ground-truth. Knowledge and Information Systems, 2015.