International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 7 4530 - 4534

Accuracy Optimization of Centrality Score Based Community Detection

Suhas S Thorat
Dept of Computer Engineering
JSCOE, Hadapsar, Pune, India
thoratsuhas@yahoo.com

Sharmila M Shinde
Dept of Computer Engineering
JSCOE, Hadapsar, Pune, India
sharmi_anant@yahoo.co.uk

Darshana R Patil
Dept of Computer Engineering
JSCOE, Hadapsar, Pune, India
drpatil2009@gmail.com

***Abstract*:-** Various concepts can be represented as a graph or the network. The network representation helps to characterize the varied relations between a set of objects by taking each object as a vertex and the interaction between them as an edge. Different systems can be modelled and analyzed in terms of graph theory. Community structure is a property that seems to be common to many networks. The division of the some objects into groups within which the connections or relations are dense, and the connections with other objects are sparser. Various research and data points proves that many real world networks has these communities or groups or the modules that are sub graphs with more edges connecting the vertices of the same group and comparatively fewer links joining the outside vertices. The groups or the communities exhibit the topological relations between the elements of the underlying system and the functional entities. The proposed approach is to exploit the global as well as local information about the network topologies. The authors propose a hybrid strategy to use the edge centrality property of the edges to find out the communities and use local moving heuristic to increase the modularity index of those communities. Such communities can be relevantly efficient and accurate to some applications.

Keywords:- *commmunity; clustering; edge centrality; network; graph*

*****

I. INTRODUCTION

Community concept is defined in many ways. One of the widespread informal definitions of the community concept is as follows - Community as a sub group of nodes which are densely interconnected to each other as compared to the rest of the network. In other terms, a community is a cohesive subset which is clearly separated from the rest of the network. Formal interpretations try to formalize and combine both these aspects of dense and sparse connectivity. Also note that this formal interpretation is not always explicit. In the algorithms and procedural approaches the end result of the processing gives a definition to the notion of community. It is not always straightforward to classify the definitions in some categories. These definitions are widely categorized in sub four groups: density-, pattern-, node similarity- and link centrality-based approaches [4].

There are various applications of communities. One of the applications is for the improvement of performance of services provided by World Wide Web. Web clients who have similar interests and are geographically closer to each other can be grouped as a community. These geographically closer groups of clients can be served by a dedicated mirror server located closer to them. Identifying communities of customers with similar interests in particular products or topics can assist online retailers (like, e. g., www.amazon.com) to set up efficient recommendation systems, that better help customers via the list of items of the sellers and enhance the business opportunities [4].

Community detection is important in many other applications. Identifying groups and their boundaries helps to categorize vertices, according to their structural position in the groups. The vertices which are at a central position in their groups, i.e. sharing a large number of edges with the other group partners, may have an important function of control and stability within that particular group. The vertices on the boundaries between groups can be of an important role of mediation and lead the relationships and exchanges between different communities. These types of classifications seem to be meaningful in social and metabolic networks. Another important aspect related to clustering is the hierarchical organization displayed by most networked systems in the real world. Real networks are usually composed by communities, which in turn include smaller communities, etc. The main purpose of community detection in networks is to find out the groups and, possibly, their hierarchical organization, by only using the information encoded in the graph topology. The community detection problem has a long tradition and it has appeared in various forms in several disciplines [2].

With the arrival of various social networking websites, and because of the need of Social Network Analysis (SNA), the demand and relevance of community detection in networks has grown in the recent years. In fact, social phenomena such as Facebook, LinkedIn and Twitter amongst others join together millions of users under a unique network. These social networks and their features are a goldmine for Social Scientists. Several research works are focused on the analysis of social networking websites; while some research describe the strategies of analysis themselves [2].

II. LITERATURE REVIEW

The community detection in networks has been studied since a long time. It is closely related to graph partitioning or clustering in graph theory and computer science, and hierarchical clustering in sociology. Finding communities within an arbitrary network can be a computationally difficult task. The number of communities, if any, within the network is typically unknown and the communities are often of unequal size and/or density. Despite these difficulties, however, several methods for community detection have been developed and each has its own advantages/disadvantages. The number of inter-community edges needn't be strictly minimized either, because more such edges can be present between large communities than between small ones.

A. Overview of community detection methods

At first sight the problem of community detection looks intuitive, but it is actually not well defined. The main elements of the problem themselves, i.e. the concepts of community and partition, are difficult to define in one definition, and require

some degree of arbitrariness and/or common sense. Indeed, some ambiguities are hidden and there are often many equally legitimate ways of resolving them.

It is important to stress that the identification of structural clusters is possible only if graphs are sparse, i.e. if the number of edges m is of the order of the number of nodes n of the graph. If m >> n, the distribution of edges among the nodes is too homogeneous for communities to make sense. In this case the problem turns into something rather different, close to data clustering, which requires concepts and methods of a different nature. The main difference is that, while communities in graphs are related, explicitly or implicitly, to the concept of edge density (inside versus outside the community), in data clustering communities are sets of points which are close to each other, with respect to a measure of distance or similarity, defined for each pair of points [1].

Below are broad level categories of the different methods for community detection:

1) Partitioning:

In these methods, the network is partitioned into groups. The numbers of partitions are predefined and usually are of approximately the same size. The partitions are formed in a way that the number of edges between groups is minimized. These methods find communities regardless of whether they are implicit in the structure or not. The number of communities will be a fixed number. This method is not always an ideal method for finding community structure in general networks.

2) Hierarchical clustering:

Hierarchical clustering is another method for finding community structures in networks. A similarity measure is used in these methods. It quantifies some type of similarity between the pair of objects. The measure is usually topological. The cosine similarity, the Jaccard index, and the Hamming distance between rows of the adjacency matrix are some of the commonly used measures. In these methods, the nodes are grouped into communities which have the similar measure. There are several common schemes for grouping nodes into communities. The widely used schemes are single-linkage clustering and complete linkage clustering. In single-linkage clustering, two groups are considered separate communities if and only if all pairs of nodes in different groups have similarity lower than a given threshold. In complete linkage clustering, all nodes within every group have similarity greater than a threshold [4].

3) Modularity optimization:

Modularity optimization is one of the most widely used methods for community detection. Modularity is a function that measures the quality of a particular division of a network into communities. The modularity optimization method detects communities by searching over possible divisions of a network for one or more, in a way that the community will have particularly high modularity. As processing search over all possible divisions is usually not practical, the most of the algorithms are based on approximate optimization methods such as greedy algorithms, simulated annealing, or spectral optimization, with different approaches offering different balances between speed and accuracy [3][5][6].

4) Statistical inference:

Methods based on statistical inference try to apply a generative model to the network data to find out the community structure. The bigger advantage of this approach compared to the other methods is that it is more principled in nature. These methods have the capacity to inherently address issues of statistical significance.

5) Clique based methods:

Cliques are the sub graphs in which every node is connected to every other node in that particular group. The nature of these types of connections is the most tightly coupled and no other type of connections can exist more than this. Hence there are many approaches to community detection in networks based on the detection of cliques in a graph.

B. Elements of Community Detection

As we have seen, various networks like social, biological, technological etc. are found to divide naturally into communities or groups. The first key step to understand the complex relations in the networks is detecting and characterizing the community structure. The concept of community detection is very much related to data clustering, graph partitioning, and hierarchical clustering. Therefore, traditional approaches in these areas can be applied for community detection. Two key categories of methods that have been widely investigated in community detection are: 1) spectral clustering-based strategies and 2) network modularity optimization techniques. Spectral clustering-based approaches rely on the maximization of the process of cutting the graph representing the given network. Since this problem falls into NP-hard category, different approximation techniques such as the normalized cuts algorithm and ratio cuts algorithm have been proposed. The main problem with spectral clustering-based methods is that the number and the size of communities in the network are defined in advance. On the other hand, the methods based on Network modularity rely on the modularity function Q to determine the maximal number of clusters in the network. A good partitioning of a network is expected to have high modularity Q with Q = (number of edges within communities) - (expected number of such edges), where the expected number of edges is evaluated for a random graph. For a graph G = (V, E) representing a directed weighted network with N nodes and an association matrix A, the modularity function is given as [4]:

$$Q=1/W \sum_{i,j=1}^{N}[A_{ij}-(S_i^{out} S_j^{in})/W]\delta_{C_i,C_j} \qquad (1)$$

Where

A_{ij} - The weight of edge $e_{i\rightarrow j}$

$S_i^{in} \sum_j A_{j,i}$, $S_i^{out} \sum_j A_{i,j}$ - The inflow, outflow of the node

i,W= $\sum_{i,j} A_{i,j},C_iC_j$ - Community that node (i, j) belongs to

$\delta_{C_iC_j}$ - Equal to 1 when i and j are in the same community and is equal to 0 otherwise.

ISSN: 2321-8169
4530 - 4534

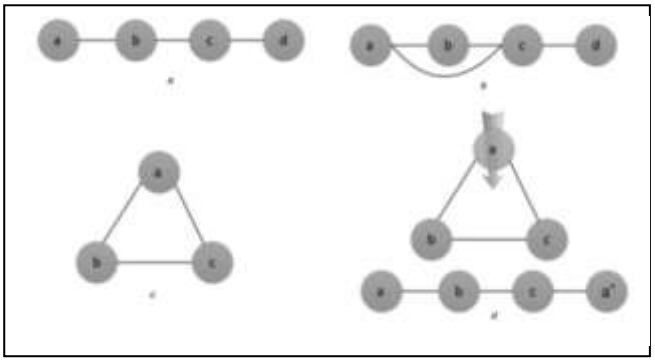


Figure 1. Edge Anti triangle centrality

III. Edge Antitriangle Centrality

The edge anti triangle centrality is defined as the ratio of the number of P4 to which a given edge belongs divided by the number of the potential P4 that might include it. The definition is proposed based on the fact that the inner links in a community belong to the more potential P4 but fewer P4, and the outer links belong to the fewer potential P4 but more P4. The denser the edges are, the more circles they belong to. The intra-community edges are denser than the inter-community edges. If the edge has larger edge anti triangle centrality score, that means it is more likely an outer link, and the lower score an edge has, it is more likely an inner link. Thus the edge anti triangle centrality property can be used for differentiating the outer links from the inner links for community detection [1]

IV. Local Moving Heuristic

A frequently used approach to modularity optimization is the local moving heuristic. The idea of the local moving heuristic is to repeatedly move individual nodes from one community to another in such a way that each node movement results in a modularity increase. The local moving heuristic iterates over the nodes in a network in a random order. For each node, it is determined whether it is possible to increase modularity by moving the node from its current community to a different (possibly empty) community. If increasing modularity is indeed possible, the node is moved to the community that results in the largest modularity gain. The local moving heuristic keeps moving nodes until a situation is reached in which there are no further possibilities to increase modularity through individual node movements. The local moving heuristic has been quite popular in the literature, probably in part because it can be implemented in an efficient way (Blondel et al., 2008). The local moving heuristic plays a central role in the proposed approach [3].

V. Proposed Approach

In the proposed work it is planned to apply the local moving heuristic to the edge anti triangle centrality based community detection algorithm. In the first step, centrality scores would be calculated for the edges. Depending on the score, the edges are grouped in the communities. In the second step, the local moving heuristic is applied to increase the modularity.

A. Algorithm

First step: Community detection using anti-triangle centrality

Input: G = (V, E)
Output: the result communities

1: Calculate the anti-triangle centrality score for each available edge
2: While the highest score 0 do
3: Remove the edge with the highest score
4: Recalculate the scores for remaining edges
5: End
6: Implement the isolated vertex handling strategy
7: Output the vertices inside the non-trivial components as those of the result communities.

Second step: Local Moving Heuristic

1: While Q increases at least of € (arbitrarily small) do
2: P = Community(G)
3: Q ← NetworkModularity(P)

B. System Architecture

The Fig. 2 depicts the system architecture. The input data is in terms of flat files having network data. These files would be read and network structure objects are created in the first component - File Reader. The Community Finder component would have the algorithm implementation for community detection using edge anti-triangle centrality. The detected communities would be stored in a temporary memory. In the third step, the local moving heuristic would be applied to increase the modularity index of the identified communities.

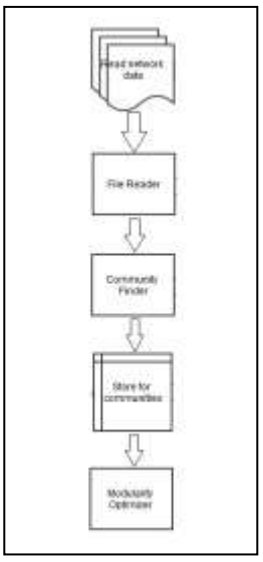


Figure 2. System Architecture

C. *Mathematical model for proposed system*

Let S, be a system such that

$S = \{I, e, I_n, O_u, T, f_{me}, DD, NDD, f_{edge}, MEM_{shared}, CPU_{CoreCnt}, Ø\}$

where

S is the proposed system
I is initial state at T <init> i.e. passing network data to the system
e end state of generated communities
I_n input of the system i.e. network data
O_u output of the system i.e. found communities
T set of serialized steps to be performed in pipelined machine cycle. In a given system serialized steps are read network data, find communities, optimize modularity etc.
f_{me} main algorithm resulting into outcome O_u, mainly focus on success defined for the solution.
DD Deterministic Data, it helps identifying the load-store function or assignment function.
NDD Non Deterministic Data of the system to be solved. These being computing function or CPU time or ALU time function contribute in time complexity.
f_{edge} set of the edge weights.
MEM_{shared} memory required to process all these operations, memory will allocated to every running process.
$CPU_{CoreCnt}$ more the number of counts double the speed and performance.
Ø null value if any.

VI. IMPLEMENTATION AND DISCUSSION

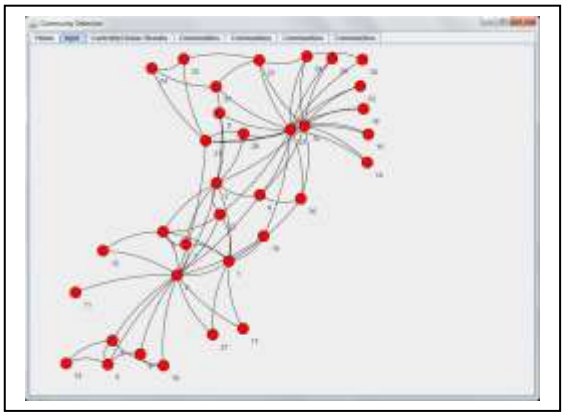


Figure 3. Input Network Data

The edge centrality based algorithm is free of any parameters. It does not need any prior number of the expected communities as well as any additional measures to decide the community structure. This approach is appropriate for community detection as the edge betweenness and the edge clustering coefficient. The algorithm is tested with various synthetic networks. The results of repetitive iterations have been analyzed. The local moving heuristic refines the communities and increases the modularity index of the communities. The results are more efficient, accurate and consistent with comparison to the plain centrality based algorithm. The Table I and the Fig 3 trend chart shows the modularity index value for the Karate Club network data having 34 nodes and 78 edges.

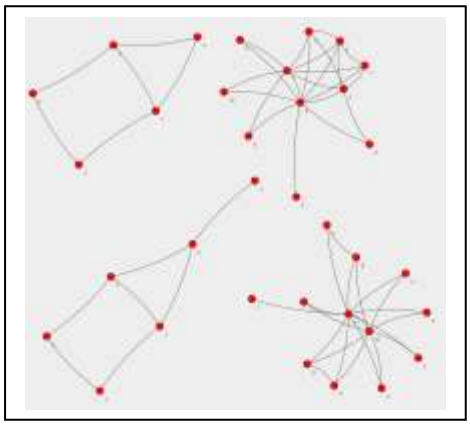
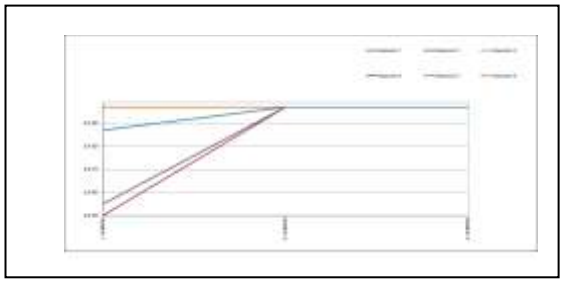


Figure 4. Resultant communties

TABLE I. MODULARITY FUNCTION RESULTS

	Iteration I	***Iteration II***	***Iteration III***
Execution I	0.4188	0.4198	0.4198
Execution II	0.4151	0.4198	0.4198
Execution III	0.4198	0.4198	
Execution IV	0.4156	0.4198	0.4198
Execution V	0.4188	0.4198	0.4198
Execution VI	0.4198	0.4198	



Modularity increase trend

VII. CONCLUSION

Several efficient approaches have been proposed to analyze networks and find communities. The main drawback of the existing techniques is that either they consider global information about the topology of the network or the local information. The proposed work is an attempt of a novel strategy to use the hybrid approach that has advantages to improve the results. The proposed strategy uses both local and global information; which will help to find (possibly) more convenient identified community groups relevant to some applications.

REFERENCES

[1] Songwei Jia, Lin Gao, Yong Gao, and Haiyang Wang, "Anti-triangle centrality based community detection in complex networks", IET Systems Biology, 2013.

[2] Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, Alessandro Provetti, "Generalized Louvain method for community detection in large networks", Proceedings of the 11th International Conference On Intelligent Systems Design And Applications, 2011.

[3] Ludo Waltman and Nees Jan van Eck, "A smart local moving algorithm for large-scale modularity-based community detection", Physics Reports, arXiv:1308.6604, 2013.

[4] Santo Fortunato, "Community detection in graphs", Physics Reports, vol. 486, 2010.

[5] Newman, M.E.J., Girvan, M., "Finding and evaluating community structure in networks", Phys. Rev. E, 2004.

[6] Clauset, A., Newman, M.E.J., Moore, C., "Finding community structure in very large networks", Phys. Rev. E, 2004.