International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

A Trusted Collaborative Technique for Detecting Common Communities under Social Networking Environments

G. Sreelatha[1], Pathuri Siva Kumar[2]

[1]Computer Science and Engineering, Rise Krishna Sai Gandhi Group of Institutions, Ongole, India

Abstract: ***Social networks represent relationships between people. Communities are groups of people with some common interests or features. In this context, overlapping means a community member could be member of some other communities at the same time. In this paper, we introduce a new framework to enhance the performance of overlapping community detection techniques. In this method, the target network is divided into several subnets and after detecting their communities, this information is used as an initialization for the final community detection technique.***

Keywords: social network, overlapping community detection, algorithm

1. Introduction

In recent years, with growing communication technologies such as online social networking websites, E-mail, Short Message Service (SMS) and cell phones, social networks are raised and grow faster. These networks are very huge with thousands or millions of members and change very fast. Analysis of these networks could reveal useful information with various applications in many domains such as communication, economics, politics, security etc.

Social network analysis (SNA) is a hot research trend in sociology, biology and computer science aiming to study social networks. Community detection is a subset of SNA. Simply, community is a group of people with some common interests or features. Communities may have overlaps with each other. For example, one person could be a member of its family, a sport club and a friendship group. In sociology, detecting and analysis of communities give the researchers useful information such as the major groups, habits and viewpoints of people in society [1].

A big challenge in the community detection area is how to define the community concept in an exact mathematical way. Unfortunately, there is no universally accepted definition, but if we consider the network as a graph, we expect that a community should be a connected subgraph with more edges inside compared to the outside edges to the rest of the graph [2], because in reality, community members have more relations between themselves rather than other people outside the community.

Different information such as network structure, user profile, user location, etc might be used in community detection. In this paper, we propose an efficient algorithm to enhance detection of overlapping communities only based on the network structure.

The rest of this paper is organized as follows: Section 2 describes some of the state of the art methods in the literature. In section 3, we introduce our proposed method. The implementation is explained in Section 4 and finally, we conclude the paper in Section 5.

2. Related Work

In recent years, many algorithms with different techniques have been proposed to detect overlapping communities.

The first attempt was done by Palla [3] with the clique percolation method (CPM). CPM considers communities as overlapping sets of fully connected subgraphs. At first, it finds all cliques of size *k* in the network. Then, a new graph is constructed based on relations of those cliques. If two cliques have *k*-1 joint members, those nodes are connected together. Connected components in cliques graph are considered as communities. CFinder1 is implemented based on CPM method and its time complexity is polynomial in many cases [3]. Its performance is not promising in large networks.

Another approach is selecting some nodes as seeds of candidate communities and trying to maximize their benefit function with expanding or removing some nodes. This method is sensitive to initial seeds and benefit function. Iterative scan (IS) [4] and LFM [5] algorithms are proposed based on this method. LFM selects a random node and expands its neighbours to construct a community. Node expansion continues until benefit function remains unchanged. Then, it randomly selects another node which has not member of any community as a new seed. Benefit function considers community size, inside and outside edges to compute its value.

Using fuzzy membership degrees to define relations of each node with candidate communities is another method [6]. In these algorithms, for each node, a membership vector, which is called belonging factor is defined. Size of this vector is *k*, equal to the communities count. These algorithms try to optimize a membership function with respect to more similar nodes (based on a similarity measure) should be in the same community.

The main weakness of these methods is determining the number of communities.

Game-th eoretic view is another method [7, 8 and 9]. In this family, each node is considered as a selfish agent who tries to leave some communities or join others based on its own utility. Each agent could join to more than one community in order to get more benefit. The game is continued until a Nash equilibrium occurred. A drawback of this family is their high computational complexity.

In label propagation approach [10], each node tries to share its label with others. At the end, nodes with the same label are located in the same community. COPRA [11] and SLPA [12] are proposed based on this technique. SLPA is a speaker listener label propagation algorithm. Each node has a memory to store listened labels from its neighbors based on a listening rule. In addition, it tells a label from the memory to its neighbors based on a speaking rule. After several iterations, the probability of observing a label in each node's memory is equal to its membership degree to that community.

Xie et al. [13] have done a comparative study on overlapping community detection algorithms. In this study, definitions, algorithms, benchmarks and other subjects about overlapping community detection are explained in details. Interested reader is highly encouraged to read this article.

3. The Proposed Method

Most of mentioned algorithms start with a simple initialization of membership values or node labels without considering the network structure. For instance, in SLPA each node is initialized with its label. So at the beginning, the number of communities is equal to nodes count and after several iterations many of those communities are merged. If the start condition initialized with better values, performance of the selected algorithm seems to be enhanced.

The main idea of our approach is considering both local and global structure of the network to extract information from local structures to use them as an initial condition for global analysis. Because communities are modular and connected subnets, this idea seems to be practical and experimental results seem promising. The main advantage of our method is achieving better run time by speed up the base method using extracted information from parallel analysis of sample subnets as the initial information.

The main cycle of the proposed method includes the following steps:

1) At first, some connected subnets are sampled from the network. This task may be done in several ways such as iteratively selecting a random node and adding it and its neighbors with breadth-first visiting while subnet size reaches to a predefined maximum size.
2) Then, for each subnet, using one of the proposed algorithms in the literature, overlapping communities are detected. This algorithm and its parameters could be the same for all subnets or be different based on each subnet structure or other strategies. This step could be done parallelly.
3) Finally, the extracted communities from previous step are used as an initial information for final community detection. This information could be used in different ways based on the final detection algorithm, for example initial labels, probabilities, or membership degrees for each node. Also, this information could be used with different weights. If the assigned weight was very low, final detection will not consider the extracted information and if it was very high, detection will be more sensetive to the initial information. This feature allows us to have a flexible behavior based on our network.

The algorithmic view of our framework could be found in Algorithm 1.

Algorithm 1: The proposed framework

[*Net*]=loadnetwork();

Step 1: Sampling from network

[*SubNets*]=GetSamples (*Net*, *c*, *s*);

Step 2: local analysis

LocalInfo=empty;

For each *subnet* **in** *SubNets*

[*Communities*]=Detect Communities(*subnet*, *params1*);

LocalInfo+= Communities;

Step 3: final analysis

[*Result*]= Detect Communities(*Net*, *params2*, *LocalInfo*, *w*);

---

4. Implementation

To study performance of our method, we compare it with the same detection method with a regular initialization. As a base method, we used SLPA-based community detection approach [12]. This method uses label propagation technique and has very good performance comparing other algorithms in the literature [13].

4.1. The base method implementation

For the base method, we used SLPA-based approach [12]. SLPA is an iterative algorithm and has three main steps: *initialization*, *evolution* and *post-processing*. Each node has a memory which keeps listened labels. At the beginning, all nodes memories initialized with their own labels. In evolution step, iteratively each node propagates it's the most probable label in the memory and updates it based on received labels from its neighbors. After some iteration, finally each node communities are extracted by selecting more probable labels from its memory. Then, nested communities are removed and maximal communities remain.

In our implementation, we initialized each node's memory with its label. Then, we shuffled nodes order. For each node, new label with the most suggestion rate from its neighbors

was added to the memory. This procedure was done for *T1* iterations. Then, for each node, we revise its memory and labels with lesser than *r1* occurrence probability were removed. Then, we generate communities regarding connected nodes with the same label and nested communities were removed to achieve maximal communities.

4.2. The proposed method implementation

In our implementation, we first get *c* sample subnets from the network with maximum size of *s* by selecting random nodes and using breadth-first selection. For community detection step, for each subnet, we use SLPA-based approach. We initialize each node's memory with its label. Then, we shuffle nodes order. For each node, new label with the most suggestion rate from its neighbors was added to its memory. This procedure was done for *T2* iterations. Then, for each node, we revise its memory and labels with lesser than *r2* occurrence were removed. Remaining labels were used as an initialization for final detection (Figure 1). After detecting communities in all subnets, for the target network, we use the base method with our initialization. We initialize each node's memory with labels detected from previous step with the weight of *w*. If there was not any information for a node, its memory is initialized with its label.

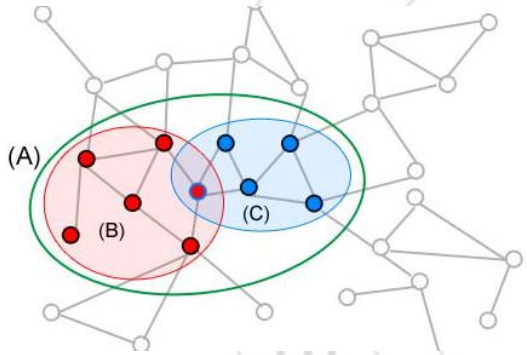

Figure 1: A sample subnet (A) and its communities (B and C).

4.3. The benchmark problems

For our experiments, we use the well-known synthetic LFR benchmark [14]. LFR lets us to generate different networks with different sizes, structures and degrees of overlapping. For evaluation results, we use extended normalized mutual information (NMI) measure proposed by Lancichinetti [15]. NMI varies between 0 and 1, with 1 corresponding to a perfect matching. In our tests, we use networks with the size of 5000 nodes. Average node degree is set to 10, where node degrees and community sizes are governed by the power laws, with exponents 2 and 1, the maximum degree is 50, the community size varies between 20 and 100, the mixing parameter μ varies from 0.1 to 0.3, which is the expected fraction of links of a node connecting it to other communities. The degree of overlapping is determined by parameters O_n (the number of overlapping nodes) and O_m (the number of communities to which each overlapping node belongs). We did our experiments on networks with different values of μ and O_m with O_n=10% (500 overlapped nodes). All figures results are mean of 100 runs with standard deviation of 0.01.

5. Conclusions

In this paper, we present a pre processing step for overlapping community detection which extracts useful initial information from subnets to enhance detection performance by achieving better results in lesser iterations. This framework could be implemented in various ways and has several useful settings to have flexible behavior based on the target network features.

References

[1] S. Wasserman. "Social Network Analysis: Methods and Applications," ***Cambridge University Press***, 1994.
[2] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri. "Extending the definition of modularity yo directed graphs with overlapping communities," ***J. Stat. Mech.***, p. 03024, 2009.
[3] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. "Uncovering the overlapping community structure of complex networks in nature and society," ***Nature***, vol. 435, pp. 814-818, 2005
[4] J. Baumes, M. Goldberg, M. Krishnamoorthy, M. Magdon Ismail, and N. Preston, "Finding communities by clustering a graph into overlapping subgraphs," in ***IADIS***, 2005.
[5] A. Lancichinetti, S. Fortunato, and J. Kertesz, "Detecting the overlapping and hierarchical community structure in complex networks," ***New J. Phys.***, vol. 11, p. 033015, 2009.
[6] S. Gregory, "Fuzzy overlapping communities in networks," ***Journal of statistical Mechanics: Theory and Experiment***, vol. 2011, no. 02, p. 02017, 2011.
[7] W. Chen, Z. Liu, X. Sun, and Y. Wang, "A game-theoretic framework to identify overlapping communities in social networks," ***Data Mining and Knowledge Discovery***, vol. 21, pp. 224-240, 2010.
[8] H. Alvari, S. Hashemi, A. Hamzeh, "To discover overlapping communities in social network: A game-theoretic approach," ***The European Journal on Artificial Intelligence (AICOM)***, 2012.
[9] A. Hajibagheri, H. Alvari, A. Hamzeh, S. Hashemi, "Social network community detection using the Shapley value," **Artificial Intelligence and Signal Processing *(AISP)***, pp. 222-228, 2012.
[10] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," ***Phys. Rev. E***, vol. 76, p. 036106, 2007.
[11] S. Gregory, "Finding overlapping communities in networks by label propagation," ***New J. Phys.***, vol. 12, p. 10301, 2010.
[12] J. Xie, B. K. Szymanski and X. Liu, "SLPA: Uncovering Overlapping Communities in Social Networks via A Speaker-listener Interaction Dynamic Process", ***IEEE ICDM workshop on DMCCI***, 2011, Vancouver, CA.
[13] J. Xie, S. Kelley and B. K. Szymanski, "Overlapping Community Detection in Networks: the State of the Art and Comparative Study," ***ACM Computing Surveys***, vol. 45, no. 4, 2013 (In press).
[14] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," ***Phys. Rev. E***, vol. 78, p. 046110, 2008.
[15] A. Lancichinetti, S. Fortunato, and J. Kertesz, "Detecting the overlapping and hierarchical community structure of complex networks," ***New Journal of Physics***, vol. 11, p. 033015, 2009.