Grenze International Journal of Computer Theory and Engineering , Oct 2015

Community Detection in Social Networks using Structural Strength and Node Attributes

Devi kannuru[1] and P. Santhi Thilagam[2]
[1-2]Department of CSE, National Institute of Technology karnataka , Surathkal, 575025, India.
devi.kannuru@gmail.com, santhisocrates@gmail.com

***Abstract*—Social Network Analysis (SNA) is focused on discovering and analyzing the pattern of interactions between the nodes of a social network to understand the implications of these interactions. Community Detection divides a social network into meaningful groups with nodes having frequent interactions among them. In order to obtain semantically relevant communities, we need to consider not only the structural information but also semantic information associated with nodes. Identifying such communities have wide applications in viral marketing, recommender systems etc. In this paper, we propose a method which leverages the semantic content along with structural interactions to detect the non overlapping communities which are not only strongly connected but also have similar interests. Experimental study demonstrates our proposed method detects semantically relevant communities as compared to structure based approaches.**

***Index Terms*— Social network analysis, Viral marketing, Recommender systems.**

I. INTRODUCTION

Social network analysis [20] is the mapping and analysis of relationships and flows between people, groups, organizations, computers, URLs, and other connected information or knowledge entities. The nodes in the network are the individuals or groups while the links show relationships between the nodes. SNA is used for visual and a mathematical analysis of human relationships, information propagation modelling, user attribute and behaviour analysis, community detection, social sharing and filtering, recommender systems development [3], link prediction [2] and entity resolution. This field is applied in wide areas varying from sociology to biological networks, collaboration networks, market analysis, information diffusion, health care etc. Social networks have the important property that individuals tend to form closely knit groups called communities within the network. Communities in a social network are groups of nodes obtained after dividing the network in such a way that nodes in the same community have more number of interactions when compared to nodes of different communities. This division in the network can be overlapping or non-overlapping. Overlapping communities are the one in which a node can be member of more than one community. For example, in a social network, a user can be part of a family, workplace, friends etc. whereas in non-overlapping communities a node is member of only one community. Communities in a social network can be defined in several ways based on the density of the edges, similarity between vertices, actions done by the users [1]. Density based algorithms define community as a set of nodes within which the number of edges is significantly higher than the expected number of edges in a random graph. Applying these methods to a social graph yields communities where the ratio of number of edges within communities is greater than the

Grenze ID: 01.GIJCTE.1.1.564

ratio of number of edges between communities. Vertex based algorithms defines community as set of nodes which are structurally similar. The general properties considered are the type of interactions between the nodes, number of common neighbour's present between nodes etc. Action based algorithms defines communities as set of nodes which perform similar kind of actions in a network. For a social network like Facebook, all the nodes which like similar pages can be identified as part of a single community.

All these methods use only the structural interactions available in the network in order to identify the communities. These methods use the graph properties such as degree, betweenness, modularity, density, conductance etc. to find the communities in the network. Conventional algorithms are useful when the nodes of the network are homogeneous. For example, identifying the processes which closely interact with each other for input/output as communities help in parallelizing an algorithm by assigning all the nodes of the community to one processor thereby reducing the communication cost. In this case, using any of the density based algorithms is sufficient to identify communities since all the nodes of network are of same type. But for networks containing heterogeneous nodes, we need to identify communities in which nodes are not only dense but also similar in nature for which the conventional methods cannot be used. For example, while identifying communities in a social network for product recommendations, we have nodes as users and the information available about nodes is user interests, preferences and the user's demographic information such as the age, major, education etc. along with the structural interactions. To overcome this problem, in this paper we propose a community detection algorithm based on structural strength and node similarity (CDSAN) to identify communities containing nodes which are not only densely connected but also homogeneous in nature.

The rest of the paper is organized as follows. Section 2 presents the work already done in this area. Section 3 gives a formal definition of the problem. Section 4 describes the proposed approach. Section 5 gives the details of the datasets used for experiments and the results obtained along with the computational complexity. Section 6 concludes our work and suggests the future scope.

II. Related Work

Several methods have been proposed to identify communities in social networks which can be mainly categorized into two classes. The first class is the structure based algorithms which depend only upon the structural interactions among the nodes to identify communities. The second class of algorithms are the semantic information based algorithms which use the semantic information such as node properties and link properties along with structural interactions to identify communities in the network.

A. Structure based algorithms

Palla et al. [17] proposed a method which finds out maximal subgraphs of size k called k-cliques present in the original graph and identified these groups as communities. This method first finds out all the cliques in the network and further finds out communities by performing analysis of clique-clique overlap matrix. This method is too restrictive as it ignores the communities in which nodes are densely connected but do not form a clique. Moreover in real world networks, the size of clique cannot be estimated. Newman et al. [4] proposed a method of finding communities in a network by identifying and removing the edges which act as bridges in the network. An edge is identified as a bridge if it is part of shortest path between many nodes. The algorithm has the time complexity of $O(n^3)$ due to the calculation of shortest paths between nodes which was further reduced in an algorithm proposed by Clauset, Newmann and Moore [5] to $O(n\log^2 n)$ using max-heaps. Pascal et al. [9] defined a method to detect communities based on random walks. The main goal in this algorithm is to identify sets of nodes which see the network in the same way and are thus the part of same community. This algorithm is based on the intuition that random walks on a graph tend to get trapped into densely connected parts leading to communities. Raghavan et al. [10] proposed a method called label propagation where each node is initially assumed to have its own label value. In successive phases, each node changes is label value based on the label values of its neighbours. Eventually all the nodes having same label values form a community. Evans and Lamboitte [11] propose a method based on the partitioning of links to detect overlapping communities. It divides the links into communities where a node can be part of more than one community. It uses the concept of line graph to identify communities. Blondel et al. [18] proposed a two phase algorithm to detect communities which uses the concept of modularity optimization. This algorithm runs in linear time for identifying communities. Bing Kong et al. [6] proposed a dynamic algorithm for community detection in social networks. This method uses the concept of Modularity Maximization and dynamic division method similar to k-means to find out communities . The time complexity of the algorithm is O(n^3). Thang et al. [8] proposed an approximation algorithm which works on directed networks to find out communities based on an approximation factor and modularity maximization. Nam P. Nguyen et al. [19] proposed a fast algorithm (DOCA) for detecting overlapping communities in social networks. The algorithm works toward the classification of nodes into local communities based on the number of interactions and then tries to combine highly overlapped communities if they share significant substructures. An adaptive algorithm is proposed in [7] to identify communities in an evolving network. The algorithm uses communities obtained from previous snapshot of the network and the changes in the network to generate communities of the new network.

B. Semantic algorithms

These algorithms use the semantic information available in the network such as user preferences, user action history, user demographic information or relation attributes such as type of link etc. to find communities. Moser et al. [12] pro-poses a method(CoPAM) for finding out set of all maximal cohesive patterns in a graph with nodes having feature vectors associated with them which gives the node information. A subspace cohesion function is defined to find dense cohesive subgraphs which are of size greater than a given threshold value. Zhou et al. [13] proposed a method called SA clustering which constructs an augmented graph by adding new vertices between the structural vertices which share a common attribute value. A neighbourhood random walk model is proposed to estimate the vertex closeness on the augmented graph. Cruz et al. [14] proposed a method based on entropy minimization. The algorithm initially takes graph modularity as the reference value and performs modularity optimization and entropy optimization steps followed by community aggregation to get the final communities. Jaho et al. [15] proposed a framework for interest similarity based community detection in social networks. In this algorithm, the edge weight set is generated using the user interest distributions. Saeed et al. [16] proposed a method which constructs an attribute augmented graph from the original graph by adding edges between the nodes sharing similar attributes and applies Markov clustering to get the final clusters. The major problem is here to keep an upper limit on the number of edges that can be added to the original graph to get the augmented graph.

III. PROBLEM STATEMENT

Given an attributed graph denoted as G = (V,E,A) where V = $v_1,v_{2....}v_n$ is the set of vertices, E=(v_i, v_j) : v_i,v_j ϵV is the set of edges, A is an attribute matrix with dimensions v×r where r is the number of attributes associated with the vertices of the graph, the problem is to identify communities $C_1,C_{2,}C_{3....}C_k$ where $\bigcup_{i=1}^{k} C_i = V$ and $C_i \cap C_j$ =Ø such that the nodes in a community are densely connected as well as homogeneous with respect to attributes.

IV. PROPOSED WORK

In this section, we propose our method which takes the structural interactions among the nodes of the network and the node features of the network to detect communities in the network. The node features of the nodes of the network are assumed to be available in the form of an attribute similarity matrix where each row of a matrix is a vector of 'd' elements where d represents the number of attributes available for the node. Our approach mainly focuses on optimizing an objective function. The objective function is constructed using exponentially weighted moving average which is a statistic for monitoring the process that averages the data by giving different weights to different entities. The two entities considered here are the structural strength and the semantic strength. The structural strength of a division indicates the strength of a particular division and measures the density of edges within a particular division. The semantic strength of a partition indicates how similar are the nodes within a particular partition. Semantic strength is calculated by finding out the similarity that the nodes of the communities have with the other members of their community with respect to a particular division. The objective function is given by:

$$F = \alpha Q_{str} + (1-\alpha)Q_{sim} \tag{1}$$

where Q_{str} represents the structural strength, Q_{sim} represents the semantic strength and is a parameter which reflects the weight given for structural strength and takes a value between 0 and 1.
The structural strength of a division is given by modularity, is the property of the network which indicates the strength of a particular division. Modularity of a division is the difference between the fraction of edges that are present between the communities and the fraction of edges that are expected to be present between the communities. The modularity of a partition is given by :

$$Q = \frac{1}{2m} \sum_{i,j} [A_{i,j} - \frac{k_i k_j}{2m}] \qquad (2)$$

where $A_{i;j}$ is the number of edges present between nodes i and j, k_i and k_j are the degrees of the vertices i and j respectively. Modularity takes a value between 0 and 1 and higher values of modularity indicates a good partition.

The attribute similarity measures the semantic strength of a division. It quantifies the similarity of the nodes to other nodes of their community. The attribute similarity of a partition is given by :

$$Q_{sim} = \sum_{k \in C} \sum_{i,j \in k} sim(i,j) \qquad (3)$$

where sim(i,j) indicates the similarity between nodes i and j and is given by the cosine similarity. Cosine similarity is a measure of similarity between two vectors that measures the cosine of the angle between them and is given by

$$Cosine(X(x_1, x_2....x_d), Y(y_1, y_2....y_d)) = \frac{(x1 * y1 + x2 * y2.... + xd * yd)}{\sqrt{x_1^2 + x_2^2 + \ldots + x_d^2} * \sqrt{y_1^2 + y_2^2 + \ldots + y_d^2}} \qquad (4)$$

A. Algorithm

We initially start with an assumption that every node is in a separate community. We calculate the gain in modularity that can be obtained by moving a node from its community to its neighbour's community. If the two nodes are the only nodes in their respective communities, the gain in modularity [5] can be given by

$$\delta(Q_{str}) = \frac{wt}{m} - \frac{k_i * k_j}{2m^2} \qquad (5)$$

where wt represents the weight of the edge between the nodes, m is the total number of edges present in the network. k_i,k_j represents the degrees of k_i and k_j respectively.The gain in similarity value is the cosine value between the two nodes and using equation 4, we find out the gain in objective function

$$\delta(F) = \alpha\delta(Q_{str}) + (1 - \alpha)\delta(Q_{sim}) \qquad (6)$$

We move the node to the neighbouring community which gives the maximum increase in objective function. This move is done only if the gain in objective function value is greater than zero. The gain in modularity by moving an isolated node 'n' from its community to a community C is given by [5]

$$\delta(Q_{str}) = [\frac{\sum_{in} + 2k_{i,in}}{2m} - (\frac{\sum_{tot} + k_i}{2m})^2] - [(\frac{\sum_{in}}{2m}) - (\frac{\sum_{tot}}{2m})^2 - (\frac{k_i}{2m})^2] \qquad (7)$$

Where $\sum_{in}$ is the sum of weights of links inside C, $k_{i,in}$ is the sum of weights of links from node i to nodes in C, m is the sum of edge weights of the network and $\sum_{tot}$ is is the sum of weights of links incident to nodes in C.

The gain in similarity value in this case is the summation of similarity of the individual node with all the nodes in the community and is given by

$$\delta(Q_{sim}) = \sum_{i}^{k} Cosine(N, I) \qquad (8)$$

where k is the number of nodes in community C, N and I represents the attribute vectors of node n and i respectively.

The calculation of semantic gain is done only if modularity gain is positive because negative values of modularity gain indicates that the node will not be well connected with rest of the members of the community and thus, finding gain in objective function value does not yield good results. If this objective gain is greater than the previous value, the cycle is continued. At the end of the first cycle, small communities are formed. This cycle is to be repeated until there is no further increase in objective function value caused by movement of individual nodes. This ends the first phase of the algorithm.

The second phase of the algorithm starts with taking the graph with small communities as an input and reconstructing the graph G to G' by modelling the communities in G to nodes in G' and the no. of edges between communities to weighted edges between the nodes. The attribute similarity matrix of graph G' is constructed, where the attribute vector of a node v_i is derived by taking the mode of the attribute values of the nodes present in the community C_i of the graph G.

The two phases are repeated till there is no increase in objective function value. The algorithm is given in Algorithm 1 The communities obtained are non overlapping as each node is part of a single community. The communities obtained at different stages of the algorithm is given in Figure 1.

V. EXPERIMENTS AND COMPLEXITY

We have implemented our algorithm using the R language with Igraph package on a system with 4GB RAM and CPU freqency 3.40 GHz and evaluated its performance using three real world datasets in terms of two quality metrics Modularity and Entropy.

A. Datasets

a) Polbooks dataset: This is a network of books about US politics sold by the online bookseller Amazon.com. Edges represent frequent co-purchasing of books by the same buyers. This network consists of 105 nodes and 441 edges. Each node has an attribute type depicting the type of the book which can be liberal, neutral or conservative.

b) Facebook100 dataset: The next two networks are taken from the Face-book100 dataset. These networks contains the interactions between the students of two colleges Caltech and Reed. Caltech network consists of 769 nodes and 16656 edges and the Reed network consists of 962 nodes and 18812 edges. Each node is associated with 7 attributes student, faculty, status, gender, major, second major/minor, dorm, house, year and high school. The dataset description is given in Table I.

TABLE I. DATASET DESCRIPTION

Dataset	Nodes	Edges	No. of attributes
Polbooks	105	441	1
Caltech Network	769	16656	7
Reed Network	962	18812	7

B. Evaluation

We have evaluated our algorithm based on two parameters such as Modularity and Entropy. Modularity evaluates the strength of the community. Modularity value lies between 0 and 1. A greater value of Modularity denotes a good partition. Entropy measures the uncertainty in a random variable. Entropy is used to quantify the dissimilarity of the attributes of the nodes in a partition. An entropy value of 0 indicates that the nodes within the partition are all alike in terms of attribute values. Entropy takes a value from 0 to ∞.

Algorithm 1: CDSAN

```
repeat
 Let membership[1..n] be the membership vector
 Outercombfunc$\leftarrow -\infty$
 repeat
        for i=1 to v do
       nei$\leftarrow$neighbors[i]
     maxgain$\leftarrow -\infty$
     for j$\in$ nei do
        if comm.(i) $\neq$ comm.(j) then
        calculate gain as modgain
        if modgain > 0 then
        calculate  similarity gain
        combgain$\leftarrow \alpha$modgain+(1-$\alpha$)simgain
        if(combgain > maxgain) then
        ver$\leftarrow$j
        if maxgain >0 then
        membership[i] ← membership[ver]
  until no change in membership
construct Graph G'(V,E,X) by considering communities as nodes and edges connecting the communities as edges between nodes .
construct new attribute similarity matrix
until  no increase in combgain
return membership[1..n]
```

Given a partition V1,V2,….Vn, the entropy of an attribute a with respect to a partition is given by

$$entropy(a, V_k) = -\sum_{s=1}^{|dom(a)|} p_s log p_s$$

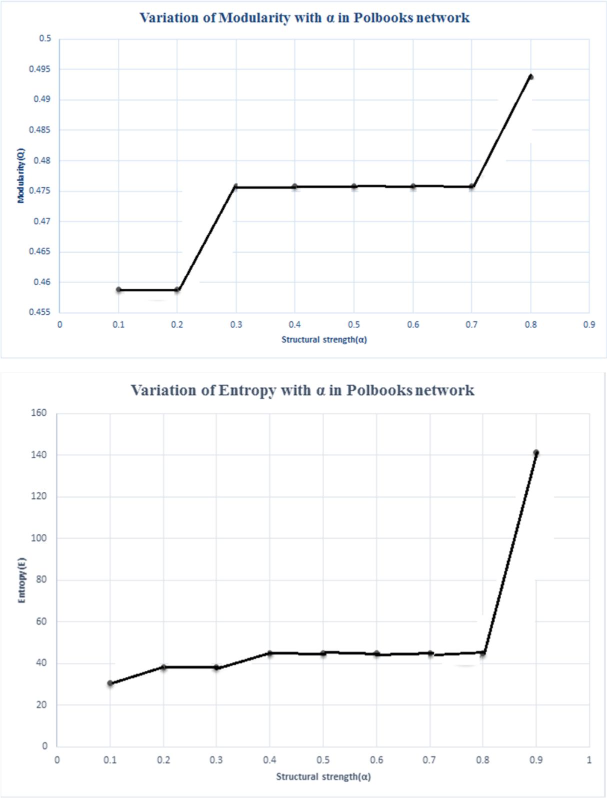

Figure1.Variation in modularity and entropy with α for Polbooks network

where p_s is the fraction of vertices in cluster V_k that take the s^{th} value in dom(a).
Two types of experiments are conducted on the datasets described above.The first experiment is conducted to compare CDSAN with Louvain method in terms of Modularity and Entropy. Table 2 shows the results obtained. The experimental results show the reduction in the entropy of the network when CDSAN is used to find communities in all the three cases with comparable modularity value. The aim of the second experiment is to study the variation in modularity and entropy values with variation in value. The results of the second experiment are shown in Figure 2, 3 and 4 respectively.

VI. Complexity

The complexity of the algorithm in the worst case is quadratic in case of complete graphs. But since the Social Networks are sparse in nature, the complexity will be O(E). Most of the execution time is taken by the

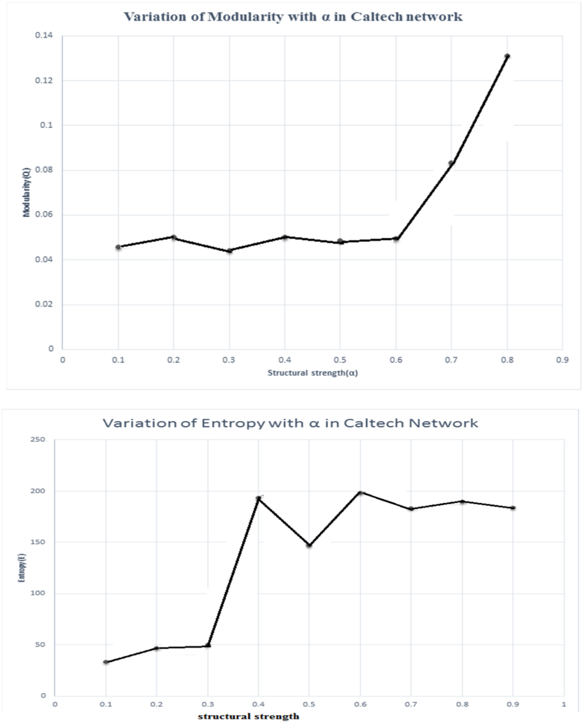

Figure2.Variation in modularity and entropy with α for Caltech network

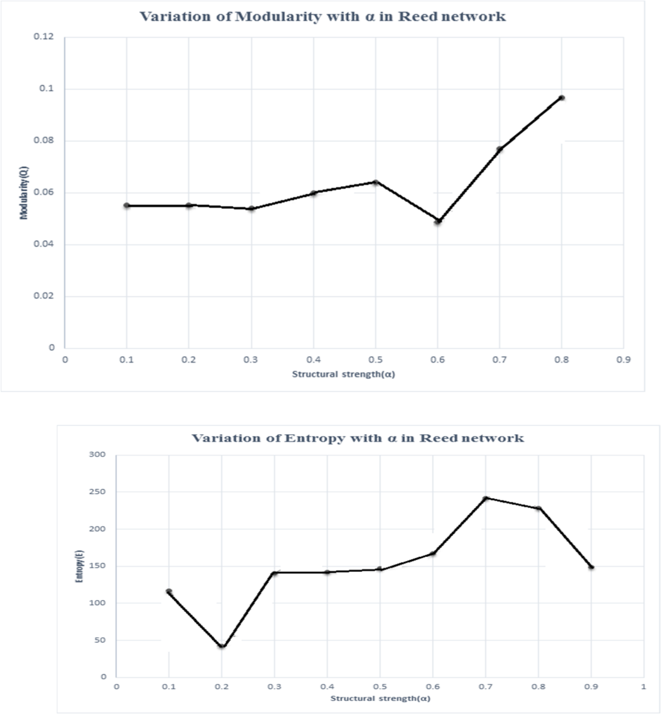

Figure3.Variation in modularity and entropy with α for Reed network

rst phase of the algorithm and reduces drastically when it comes to the later stages because of the reduction in the number of nodes in the reduced graphs. Moreover, the modularity gain and the similarity gain an be calculated in O(n) time which also contributes for the reduction of complexity.

TABLE II. COMPARISON OF MODULARITY AND ENTROPY

Dataset	Nodes	Edges	Algorithm	Avg Entropy	Modularity
Polbooks	105	441	Louvain	68.569	0.5197166
Polbooks	105	441	CDSAN	38.4388	0.4589009
Facebook	769	15931	Louvain	57.14	0.0446892
Facebook	769	15931	CDSAN	36.836	0.4336775

VII. CONCLUSION

Most of the algorithms proposed in the literature only use the structural interactions in order to identify the communities. The proposed method uses the demographic information and the interactions among the nodes to find out the good quality communities. As a part of the future work, it is necessary to take into account that all the attributes associated with a node are not equally important in determining communities. Therefore, we need to propose modifications to incorporate the prominence of attributes in the algorithm.

REFERENCES

[1] Coscia, M., Giannotti, F., Pedreschi, D. (2011). A classification for community discovery methods in complex networks. Statistical Analysis and Data Mining, 4(5), 512-546.

[2] Gong, Neil Zhenqiang, et al. "Jointly predicting links and inferring attributes using a social-attribute network (san)." arXiv preprint arXiv:1112.3265 (2011).

[3] Wu, Hao, Vikram Sorathia, and Viktor K. Prasanna. "When diversity meets special-ity: Friend recommendation in online social networks." HUMAN 1.1 (2013): pp-52.

[4] Newman, Mark EJ, and Michelle Girvan. "Finding and evaluating community struc-ture in networks." Physical review E 69.2 (2004): 026113.

[5] Clauset, Aaron, Mark EJ Newman, and Cristopher Moore. "Finding community structure in very large networks." Physical review E 70.6 (2004): 066111.

[6] Kong, Bing, et al. "A dynamic algorithm for community detection in social net-works." Intelligent Control and Automation (WCICA), 2012 10th World Congress on. IEEE, 2012.

[7] Nguyen, Nam P., et al. "Adaptive algorithms for detecting community structure in dynamic social networks." INFOCOM, 2011 Proceedings IEEE. IEEE, 2011.

[8] Dinh, Thang N., and My T. Thai. "Community Detection in Scale-Free Networks: Approximation Algorithms for Maximizing Modularity." Selected Areas in Commu-nications, IEEE Journal on 31.6 (2013): 997-1006.

[9] Pons, Pascal, and Matthieu Latapy. "Computing communities in large networks using random walks." Computer and Information Sciences-ISCIS 2005. Springer Berlin Heidelberg, 2005. 284-293.

[10] Raghavan, Usha Nandini, Rka Albert, and Soundar Kumara. "Near linear time algorithm to detect community structures in large-scale networks." Physical Review E 76.3 (2007): 036106.

[11] Evans, T. S., and R. Lambiotte. "Line graphs, link partitions, and overlapping communities." Physical Review E 80.1 (2009): 016105.

[12] Moser, Flavia, et al. "Mining Cohesive Patterns from Graphs with Feature Vec-tors." SDM. Vol. 9. 2009.

[13] Zhou, Yang, Hong Cheng, and Je rey Xu Yu. "Graph clustering based on struc-tural/attribute similarities." Proceedings of the VLDB Endowment 2.1 (2009): 718-729.

[14] Cruz, Juan David, Ccile Bothorel, and Franois Poulet. "Entropy based community detection in augmented social networks." Computational Aspects of Social Networks (CASoN), 2011 International Conference on. IEEE, 2011.

[15] Jaho, Eva, Merkouris Karaliopoulos, and Ioannis Stavrakakis. "Iscode: a framework for interest similarity-based community detection in social networks." Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on. IEEE, 2011.

[16] Salem, Saeed, et al. "Discovering Communities in Social Networks Using Topology and Attributes." Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on. Vol. 1. IEEE, 2011.

[17] Palla, Gergely, et al. "Uncovering the overlapping community structure of complex networks in nature and society." Nature 435.7043 (2005): 814-818.

[18] Blondel, Vincent D., et al. "Fast unfolding of communities in large networks." Journal of Statistical Mechanics: Theory and Experiment,2008.10 (2008): P10008.