Javaid Iqbal Bhat, Rumaan Bashir

A Comparative Analysis of Community Detection in Online Social Networks

Javaid Iqbal Bhat[1],Rumaan Bashir[2]

[1,2,] Department of Computer Science,
Islamic University of Science & Technology, Awantipora, J&K, INDIA.

Abstract

An imperative aspect of Online Social Networks is that they contain communities of entities typically thought of as a group of nodes with better interactions amongst its members than the members and the rest of network. Detection of such communities has emerged as a problem of immense interest because of the fact that it provides a valuable coarse-grained representation of complex networks, responsible for providing the assistance while analyzing the structure and function of the online social networks. Recognition of coherent and well-connected communities in a large real-world graph is considered a problem of NP hard nature and one needs to employ the heuristics or the approximation algorithms for its detection. Many community detection algorithms have been proposed based on various approaches from graph partitioning to spectral clustering. Most of the current community detection algorithms are limited to deal with non-overlapping communities, which largely do not work well on overlapping community recognition. On the similar lines, other clustering algorithms have either scalability or usability issues, as global algorithms that require entire graph to be accessible do not scale well or the local algorithms that do not require the full knowledge of the graph may cover only a portion of a graph. In this paper we will study the large scale community detection and compare them on both qualitative as well as quantitative standards for social network clustering.

Keywords:*Community Detection, GraphPartitioning, OnlineSocial Network, Overlapping Community Detection, Spectral Optimization.*

[2]***Author for Correspondence***E-mail: rumaan.b@gmail.com, Tel: +91-9906510488

INTRODUCTION

The term Community has no universally agreed upon definition as of now. However, a popular and a better one referred by most of the researchers is: “A Community in an online social network represents a group of people with very good intragroup communications than inter-group communication and are being represented by a graph of ‘n’ number of nodes with dense links connecting these nodes”. The community detection is never a simple partition of network but a meaningful breakage of social network. Whenever there is a network that involves the people for active communicational participation, there is always an inherent community structure. This community structure is hidden, unpredictable and unclassified and as a result of it, such community detection emerges as a cluster problem instead of classification one. Moreover, the detection of community structure is always as uneasy task because of its vast relationship within a social circle.
This community detection in online social networks has been divided broadly into two categories based on the node participation and implies that whether a unique group participation or multi-group

participation of any node in the social graph is permissible or not. These two type of the community detection methods are as: Non-Overlapped Community Detection and Overlapped Community Detection. In a non-overlapped method, a node can't be a member of more than one group while in an overlapped method a node can be a member of more than one group. Though the non-overlapped methods are very simple in understanding regardless of computing complexity but the communities in online social networks are overlapped in nature, so the real need is to study and find overlapped community structure in an online social network.

Community detection in online social networks has also been divided into other two categories based on the criteria that whether the sizes of communities remain same or change in time. These two are referred as Static Community Detection and Dynamic Community Detection. In the former the size of communities remain always constant while in later the communities shrink or expand in size. The static community detection methods are simple but in real the community structure is adaptive in nature in case of online social networks, so dynamic methods are real masters to be used to uncover the hidden community structure in online social networks.

Community detection in online social networks has not been restricted to only these above two simple cases but community detection methods can uncover nested community structure as well. This type of community detection is termed as Hierarchical Community Detection in online social networks. These hierarchical methods provide us a better way to go ladder stepwise uncovering in community structure detection because in online social networks there are low granular communities within high granular communities in nature. These hierarchical methods provide us more insight into the social relationships in an online social network, so that more real community structure can be uncovered.

The rest of the paper is organized as: In section 2, an overview of the existing Community Detection approaches in Online Social Networks is presented. Section 3 reflects upon the comparative analysis of various Clustering methods, followed by conclusion to be given in Section 4.

OVERVIEW OF EXISTING COMMUNITY DETECTION APPROACHES

The first step towards community detection has been made by Girvan and Newman [1] and then this field emerged as a major area of interest for the research community. Since the work of Girvan and Newman [1] every researcher has either put forward a new measure (metric) for community detection in social networks or has optimized the already existing measures. The various community detection approaches are given as

A. Traditional Methods

There are three types of traditional methods to detect communities in a social network.

1. Graph Partitioning

Graph partitioning method represents to divide the nodes in G groups of predefined size, such that the number of edges lying between the groups is minimal. The number of edges running between clusters is called as cut size. Most variants of the graph partitioning problem are NP-hard. If the solutions are not necessarily optimal, then also there are several algorithms that can do a good job. Many algorithms perform a bisection of the graph. Generally, partitioning into more than two groups is achieved by iterative bi-section. The Kernighan-Lin algorithm is one of the earliest proposed methods and is still frequently used. The problem of partitioning electronic circuits onto boards motivated the authors as the nodes contained in different boards need to be linked to each other with the least number of connections. The Kernighan-Lin algorithm was extended to get partitions in any number of parts; however, the run-time and storage costs increase rapidly with the number of clusters.

There are several efficient routines to compute maximum flows in graphs, like the algorithm of Goldberg and Trajan [2]. In the graph of the World Wide Web, Flake et al. have used maximum flows to identify communities. The web graph is directed but Flake et al. treated the edges as undirected for the purpose of the calculations. The internal degree of each node must not be smaller than its external degree in a community. So, Web communities are defined to be strong. An artificial sink T is added to the graph and one calculates the maximum flow from a

source node S to the sink T: the corresponding minimum cut identities the community of node S, provided S shares a sufficiently large number of edges with the other vertices of its community. It is necessary to provide as input the number of groups and their sizes in some cases. So, Algorithms for graph partitioning are not good for community detection. Besides, it is not a reliable procedure using iterative bi-sectioning to split the graph in more pieces.

2. Hierarchical Clustering

Hierarchical clustering is a widely used data analysis tool. The idea behind this clustering is to build a binary tree of data that merges similar groups of points. If the graph is split then it is not easy to know the total number of clusters. If the graph is in hierarchical structure with small groups included within larger groups, in that case hierarchical clustering algorithm may be used.

3. Spectral Clustering

Donath and Hoffmann [3] contributed first on spectral clustering in 1973. They used Eigen Vectors of the adjacency matrix to partition the graph. Spectral clustering makes use of Eigen values of the similarity matrix of the data. The similarity matrix is provided as an input and consists of a quantitative assessment of the relative similarity of each pair of points in the dataset. Andrew Y. Ng et. al in [4] have analyzed the algorithm of spectral clustering as the ideal case and the general case.

B. Divisive Algorithm

1. Newman-Girvan Algorithm:

This algorithm follows the steps stated in below:

i. Calculate the betweenness for all edges in the network.
ii. Remove the edge with the highest betweenness.
iii. Recalculate betweennesses for all edges affected by the removal.
iv. Repeat from step 2 until no edges remain.

If a graph contains groups that are inter-connected to each other and loosely connected by few edges, then all shortest paths between different groups must go along with one of these few edges. Thus, the edges connecting groups will have high edge betweenness. Betweenness can be calculated by using the fast algorithm of Newman, which calculates betweenness for all m edges in a graph of n vertices in time O(mn). Because this calculation has to be repeated once for the removal of each edge, the entire algorithm runs in worst-case time O(n2m). Rattigan et al. proposed a fast version of Newman-Girvan algorithm in 2007.

C. Modularity-Based Methods

High values of modularity represent good partitions of a graph. There are four techniques we have discussed below:

1. Greedy Techniques

Newman proposed first greedy method to maximize modularity. It is a hierarchical clustering method where the graph does not contain edges initially in the graph; edges are added one by one during the procedure.

2. Simulated Annealing

To get global optimization, simulated annealing [5] is probabilistic procedure used in different fields and problems. This procedure consists of the space of possible states looking for the maximum global optimum of a function F. Guimera et al. [6] first applied simulated annealing for modularity optimization. The standard implementation of them [7] combine two types of moves: local moves, where a single node is shifted from one cluster to another randomly; and global moves, which consist of mergers and splits of communities.

3. Extremal Optimization

Boettcher and Percus [8] proposed extremal optimization and it is a heuristic search procedure. This technique is based on the optimization of local variables. In [9] Duch and Arenas used this technique for modularity optimization. Modularity can be measured as a sum over the nodes in the graph. We can get a fitness measure for each node by dividing the local modularity of the node by its degree. Degree of the node does not define the measure.

4. Spectral Optimization

By using the Eigen values and Eigen vectors of a spectral matrix, modularity can be optimized. Wang et al. used community vectors to achieve high-modularity partitions into a number of communities smaller than a given maximum. If the Eigen Vectors is taken corresponding to the two largest Eigen Values, then we can obtain a split of the graph in three clusters. In 2009, Richardson et al. [10] presented a fast technique to achieve graph tri-partitions with large modularity along these lines.

D. Spectral Algorithms

In the previous sub-section, we have learnt the spectral properties of graph matrices that are frequently used in finding the partitions in a graph. In 2005, Slanina and Zhang [11] have shown that if the graph has a clear community structure, then Eigen vectors of the adjacency matrix may be localized. In 2009, Mitrovic and Tadic [12] presented a comprehensive analysis of spectral properties of modular graphs. In 2007, Alves [13] used Eigen values and Eigen vectors of the Laplacian matrix to compute the effective conductances for pairs of nodes in a graph. We compute the transition probabilities by enabling the conductances for a random walker moving on the graph, and from the transition probabilities, we can build a similarity matrix between the node pairs. Hierarchical clustering is applied to join nodes in communities. If we need to compute the whole spectrum of the Laplacian matrix, the time taken by this algorithm is, i.e. the algorithm proposed by Alves [13] is extremely slow.

E. Dynamic Algorithms

There are three algorithms we have to discuss in this section: Spin models, Random walk, and Synchronization.

1. Spin Models

In statistical mechanics, the Potts model [14] is the most popular model. This model elaborates a system of spins that can be in Q different states. It favours spin alignment such that all spins are in the same state at zero temperature. That means the interaction is ferromagnetic in this model. The ground state of the system may not be the one where all spins are aligned if antiferromagnetic interactions are also present. But, different spin values coexist in homogeneous clusters in a state. If Potts [14] spin variables are assigned to the nodes of a graph with community structure, then the structural groups could be recovered from like-valued spin clusters of the system while the interactions are between neighboring spins, as there are many interactions inside communities than outside. Based on Potts model [14], in 2004, Reichardt and Bornholdt [15] proposed a method to detect communities that maps the graph onto a zero-temperature Q-Potts model with nearest-neighbour interactions.

2. Random Walk

In 1995, Hughes [16] showed that random walk can be useful to detect the clusters in a graph. If a graph contains several clusters, a random walker spends a long time inside a cluster due to the high intra-connections among all the nodes. All the clustering algorithms based on the random walk can be trivially extended to the case of weighted graphs (O(n3)). In 2005, Zhou and Lipowsky [17] used biased random walkers, where the bias happens to be the fact that walkers usually move towards the nodes sharing a large number of neighbors with the starting node in a graph. A proximity index is defined to show that how much a pair of nodes is closer to all other nodes in the graph. The procedure is called Net Walk to detect the communities in a graph, where

Net Walk is a hierarchical clustering method, where the proximity defines the similarity between nodes. The time complexity of this method is O(n3). In 2008, Weinan et al. [18] described that the best partition of a graph in k communities, where the chain describing a random walk on the meta-graph provides the best approximation of the full random walk dynamics on the whole graph.

3. Synchronization

Synchronization [19] is an excellent process that occurs in the systems and interacts among the units in nature and technology. All the units of the system are in the similar state at every moment while the system is in synchronized state. To detect the communities in a real world network, synchronization can also be applied. In 2007, Boccaletti et al. [20] have designed a method for community detection applying the concept of synchronization.

F. Overlapping Community Detection

A significant characteristic of many complex networks, particularly real-world social networks is community overlapping. A user in a network may belong to more than one community, such as the community of family members, that of friends, and community of co-workers. Further, the number of communities an individual can belong to is essentially unlimited because a person can simultaneously associate with as many groups as he/she wishes. The first work on overlapping communities started by Palla in 2005 [21]. He extended the Girvan Newman's problem to find overlapping communities where each node can belong to one or more communities. Since then, a very high number of algorithms have been proposed with great improvements in time and efficiency. Overlapping community detection algorithms were reviewed and categorized into four classes, namely Clique Percolation Algorithms, Agent and Dynamic based Algorithms, Fuzzy based Algorithms, Local Expansion Algorithms. All these type of approaches for detecting the communities in overlapping social network are all explained below:

1. Clique Percolation

First attempt to deal with overlapping community structure was Clique Percolation method (CPM). It is a deterministic community detection method, which allows for overlapping communities and is purely based on local topological properties of a network [21]. It begins by identifying all cliques of size k in a network. After identification, a new graph is constructed such that each vertex represents one of these k-cliques. Two nodes are connected if the k-cliques which represent them share k-1 members. The connected components in the resultant graph identify which cliques compose the communities. Overlap between communities is possible, since a vertex can be in multiple k-cliques simultaneously. CPM assumes that the graph has large number of cliques and it is suitable only for networks with dense connected parts. If a graph contains few cliques, then it fails to detect meaningful covers.

Weights were added into networks based on the concept of percolating k-cliques with high enough intensity by Farkas [22]. In Weighted CPM (CPMw), a k-clique is included into a community only if it has intensity larger than a fixed threshold value. When compared with CPM, it produces modules with smooth contours with a particular intensity threshold. It expands slightly the modules located by the CPM and may attach to each module additional k-cliques containing weaker links. The first step towards the enhancement of CPM was made by Kumpula [23] with the Sequential Clique Percolation method (SCP). SCP was designed to find clique communities for a given size. It sequentially inserts links to the network and keeps track of the emerging community structure. When links are inserted in order of decreasing weight, the algorithm allows for detecting k-clique communities at chosen threshold levels in a single run and simultaneously produces a dendrogram representation of hierarchical community structure. This algorithm suits more for weighted networks containing hierarchical communities. The computational time of the SCP algorithm scales linearly with the number of k-cliques in the network. SCP is faster than CPM and allows multiple weight thresholds in a single run.

2. Fuzzy Based

In non-fuzzy overlapping, each vertex belongs to one or more communities with equal strength with an individual either belonging to a community or not. With fuzzy overlapping, each individual may also belong to more than one communities but the strength of its membership to each community can vary. It is expressed as a belonging coefficient that describes how a given vertex is distributed between communities [Gregory 2011]. Zhang S., Wang R.S., Zhang X.S [24] proposed a method combining spectral mapping, fuzzy clustering and optimization of a quality function. It converts a network to (k-1) dimensional Euclidean space and use the fuzzy c-means algorithm to detect up to k communities. Both detection accuracy and computation efficiency rely on the user specified value k which is the upper bound on the number of communities. Nepusz Y., Petrocz A., Negyessy L., and Bazso F [25] modeled the overlapping community detection as a nonlinear constrained optimization problem which can be solved by simulated annealing methods. It allows each vertex of the graph to belong to multiple communities at the same time. The algorithm determines the optimal membership degrees with respect to a given goal function.

A new measure is also introduced that identifies outlier vertices that do not belong to any of the communities, bridge vertices that belong significantly to more than one single community, and regular vertices that fundamentally restrict their interactions within their own community. A technique that combines disjoint detection methods with local optimization algorithms [26] partitions any algorithm for disjoint community detection. Nodes are added and removed by communities. Variance, the difference of two fitness scores on a community, either including a node or removing node, is computed. The normalized variances form fuzzy membership vectors of a node. A hybrid algorithm with probabilistic approach to detect overlapping communities based on Bayesian non-negative matrix factorization (NMF) to achieve soft partitioning of a network in a computationally efficient manner was proposed by [Psorakis et al. 2011]. An element Vij in matrix 'V' which denotes the count of the interactions that took place between two nodes i and j, is decomposed via NMF as part of the parameter inference for a generative model. The advantage of this method is as it quantifies "how strongly" each individual participates in every other group and does not suffer from the drawbacks of resolution limit.

3. Agent and Dynamic Based

Label Propagation Algorithm (LPA) was proposed by Raghavan et al in 2007, which identifies a community in large networks and runs linearly in the number of edges. Initially, each node is assigned a unique label. After every iteration, vertex updates its label by replacing the label used by the same maximum number of neighbors. The neighbor is chosen randomly. After several iterations, labels get associated with all the members of a community and all vertices with the same label are added to one community. This algorithm uses the network structure alone as its guide and requires neither optimization nor prior information about the communities. But it can detect only disjoint communities. Gregory S proposed a Cluster-Overlap Newman Girvan Algorithm (CONGA) [27] which is an "overlapping" version of existing disjoint community detection algorithm. CONGA extends the algorithm of Girvan and Newman which splits a vertex into two vertices repeatedly during the divisive clustering process. This algorithm considers both split betweenness, defined by the number of shortest paths on the imaginary edge, and also the conventional edge betweenness. A revised version of CONGA [Gregory S 2008] uses local betweenness to optimize the speed. In this algorithm, communities have multiple copies of a vertex which results in overlapping community.

LPA was extended by Gregory S [28], by modifying the nodes to possess multiple labels called Community Overlap Label Propagation Algorithm (COPRA). It allows vertex label to be a set of community identifier and belonging coefficient. Sum of the belonging coefficient of the communities over all neighbors is normalized. After each iteration t, COPRA, synchronously updates label of a vertex based on its neighbors labels in iteration t-1. According to the recent benchmarks of Lancichinetti A. and Fortunato S. [2009], COPRA proves to be the best method for detecting overlapping network communities.

Unnecessary updates in each iteration of LPA should be avoided to improve the execution time in extremely large networks. An algorithm that updates rules and label propagation criteria in LPA, by bookkeeping the information about the boundaries of the

currently existing communities was proposed in 2011 by Xie J., Szymanski B. K. The competition between communities is restricted only to their boundaries after a few iterations. For nodes inside any community, the updates are unnecessary, as long as it does not change to new labels. In the modified neighborhood strength driven LPA algorithm, the additional time required to attempt the updates that are expected to fail to change labels is reduced by book keeping process. When compared with original LPA, it improves the speed and quality of the detected communities. The improvement requires neither any threshold value nor modification of the stop criterion but it is not overlapping.

Unnecessary updates in each iteration of LPA should be avoided to improve the execution time in extremely large networks. An algorithm that updates rules and label propagation criteria in LPA, by bookkeeping the information about the boundaries of the currently existing communities was proposed in 2011 by Xie J., Szymanski B. K. The competition between communities is restricted only to their boundaries after a few iterations. For nodes inside any community, criteria in LPA, by bookkeeping the information about the boundaries of the currently existing communities was proposed in 2011 by Xie J., Szymanski B. K. The competition between communities is restricted only to their boundaries after a few iterations. For nodes inside any community, the updates are unnecessary, as long as it does not change to new labels. In the modified neighborhood strength driven LPA algorithm, the additional time required to attempt the updates that are expected to fail to change labels is reduced by book keeping process. When compared with original LPA, it improves the speed and quality of the detected communities. The improvement requires neither any threshold value nor modification of the stop criterion but it is not overlapping. In order to determine how the node spreads its information to others and to process the information received from other nodes in dynamic process, In 2012, Xie J. and Szymanski B.K proposed a Speaker-Listener Label Propagation Algorithm (SLPA) method to mimic people's preference of spreading most frequently discussed opinions. In SLPA, when the node serves as an information provider it is treated as speaker and while consuming information it acts as a listener [29]. Typically, a node can hold as many labels as it likes depending on what it has learned from the underlying network structure. In LPA, a node updates its label completely forgetting the old knowledge by Raghavan et al. [30]; Gregory [2010]. But, SLPA provides each node with a memory to store received information (i.e., labels) and accumulates its knowledge from repeated observation of labels. When a node observes more labels, the more likely it will spread this label to other nodes. SLPA collects only label information that reflects the underlying network structure during the evolution.

Chen et al., [2010] proposed a game-theoretic framework where community formation is played by selfish agents on the social network. Initially each agen's social interaction is known and fixed. Every agent has an intrinsic utility that associates with the communities it joins and those it does not. The two components of the utility function are gain and loss. Agent aims to maximize their own utility. A Nash equilibrium of the game can be readily interpreted into a community structure of the network, the communities every node belongs to in a Nash equilibrium is the output of this algorithm.

4. Local Expansion and Optimization

Algorithms utilizing local expansion and optimization rely on a local benefit function that characterizes the quality of a densely connected group of nodes. Baumes et al. iteratively improved the candidate cluster of CONGA by a two phase method whereby a network is first broken into a number of disjoint "seed" communities and then adding vertices to and removing vertices from the candidate set until its density is maximized [31]. It depends on finding a local maximum of density. Lancichinettia A., Fortunato S, proposed LFM method which finds both overlapping communities and the hierarchical structure [32]. The node is distributed to different communities after finding the highest fitness value through local optimization. Several visits may happen to one node which places the node in more than one community. The size of each community is decided by tuning the resolution parameter, which leads to meaningful hierarchical communities. The only difference between this algorithm and that of Baumes [31] is that a seed community is simply a vertex that is not yet assigned to any community. This algorithm provides

a general framework that yields a large class of algorithms by choosing a different expression for the fitness function or a different optimization procedure of the fitness as a single cluster.

In a recent attempt, Goldberg et al. [33] proposed Connected Iterative Scan (CIS) method for finding overlapping communities by connectedness and local optimality properties of a community. This method consists of repeated scans where each of them is based on "seeds" obtained in the previous scan. It examines each node of the network, adding or removing it to the set. As a result, the density of the set is increased. Until the set is locally optimal with respect to a defined density metric, the scans are repeated [34]. However, the results may not be deterministic due to its un-predetermined choice of initial seed nodes.

Order Statistics and Local Optimization method (OSLAM) is the first method capable to detect clusters accounting for edge directions, edge weights, overlapping communities, hierarchy and community dynamics [35]. This algorithm is based on a fitness measure, a score that is tightly related to the significance of the clusters in the configuration model. First it looks for significant clusters until convergence. Then it analyzes the resulting set of clusters, trying to detect their internal structure or possible unions thereof. Finally, it detects the hierarchical structure of the clusters. It generally finds different depths in different hierarchical branches. In particular, OSLOM is superior on directed graphs and in the detection of strongly overlapping clusters.

Intrinsic Longitudinal Community Detection (iLCD) is capable of detecting both static and temporal communities [36]. The algorithm updates the existing communities by adding a new node if it's number of second neighbors and number of robust second neighbors are greater than expected values. A new community is formed if the minimum pattern is detected by the new edges. If the similarity between two communities (ratio of nodes in common) is high, then communities are merged. The algorithm depends on two parameters for adding a node and merging two communities. Local optimization algorithms have gained more attention in the recent years.

This was some of the background details of work that has been done on community detection in social networks and the field is still a new one and needs to have a much more work in it.

COMPARISON AMALYSIS OF ALGORITHMS

Community detection is actually a clustering process. A comparative overview of various basic clustering methods is given below in Table 1:

***Table 1**: Overview of various Clustering Methods*

Method	General Characteristics
Partitioning Methods	• Find mutually exclusive clusters of spherical shape • Distance-based • May use mean or median (etc.) to represent cluster Centre • Effective for small to medium-size datasets
Hierarchical Methods	• Clustering is hierarchical decomposition(i.e. multiple levels) • Cannot correct erroneous merges or splits • May incorporate other techniques like micro-clustering or consider object "linkages"
Density Based Methods	• Can find arbitrarily shaped clusters • Clusters are dense regions of objects in space to be separated by low-density regions • Cluster density: Each point must have a minimum number of points within its "neighbourhood" • May filter out outliers
Grid-Based Methods	• Use a multi-resolution grid data structure • Fast Processing time(typically independent of the number of data objects , yet dependent on grid size)

1. Clique Percolation Method (CPM)

A k-clique is just any collection of 'k' vertices in which every possible edge is present. In otherwords a clique is, as you say, an induced subgraph in which every vertex is connected to every other vertex. Cliques may be confined in one another, in fact, every subgraph of a clique is necessarily itself a clique. So if you have a 4-clique, then each of its four subgraphs with three vertices are cliques as well.

The clique percolation method builds up the communities from k-cliques, which correspond to complete sub-graphs of 'k' nodes. (e.g., a k-clique at $k = 3$ is equivalent to a triangle). Two k-cliques are considered adjacent if they share $k - 1$ nodes. A community is defined as the maximal union of k-cliques that can be reached from one other through a series of adjacent k-cliques. Such communities can be best interpreted with the help of a k-clique template (an object isomorphic to a complete graph of 'k' nodes). Such a template can be placed onto any k-clique in the graph and rolled to an adjacent k-clique by relocating one of its nodes and keeping its other $k - 1$ nodes fixed. Thus, the k-clique communities of a network are all those sub-graphs that can be fully explored by rolling a k-clique template in them.

This definition allows overlaps between the communities in a natural way, as illustrated in Fig.1, showing four k-clique communities at 'k' = 4. The communities are color-coded and the overlap between them is emphasized in red. The definition above is also local: if a certain sub-graph fulfills the criteria to be considered as a community, then it will remain a community independent of what happens to another part of the network far away. In contrast, when searching for the communities by optimizing with respect to a global quantity, a change far away in the network can reshape the communities in the unperturbed regions as well. Furthermore, it has been shown that global methods can suffer from a resolution limit problem, where the size of the smallest community that can be extracted is dependent on the system size. A local community definition such as here circumvents this problem automatically.

Since even small networks can contain a vast number of k-cliques, the implementation of this approach is based on locating all maximal cliques rather than the individual k-cliques [37]. This inevitably requires finding the graph's maximum clique, which is an NP-hard problem. (We emphasize to the reader that finding a maximum clique is much harder than finding a single maximal clique.) This means that although networks with few million nodes have already been analyzed successfully with this approach, the worst case runtime complexity is exponential in the number of nodes.

There are a number of implementations of clique percolation. The clique percolation method was first implemented and popularized by CFinder [38] (freeware for non-commercial use) software for detecting and visualizing overlapping communities in networks. The program enables customizable visualization and allows easy strolling over the found communities. The package contains a command line version of the program as well, which is suitable for scripting.

2. Girvan–Newman algorithm

The Girvan–Newman algorithm detects communities by progressively removing edges from the original network. Subsequently, the connected components of the remaining network are the actual communities. Instead of trying to construct a measure that tells us which edges are the most central to communities, the Girvan–Newman algorithm focuses on edges that are most likely "between" communities.

The Girvan–Newman algorithm extends this definition to the case of edges, defining the "edge betweenness" of an edge as the number of shortest paths between pairs of nodes that run along it. If there is more than one shortest path between a pair of nodes, each path is assigned equal weight such that the total weight of all of the paths is equal to unity. If a network contains communities or groups that are only loosely connected by a few inter-group edges, then all shortest paths between different communities must go along one of these few edges. Thus, the edges connecting communities will have high edge betweenness (at least one of them). By removing these edges, the groups are separated from one another and so the underlying community structure of the network is revealed.

The algorithm steps for community detection are summarized below:

i. The betweenness of all existing edges in the network is calculated first.
ii. The edge with the highest betweenness is removed.
iii. The betweenness of all edges affected by the removal is recalculated.
iv. Steps 2 and 3 are repeated until no edges remain.

The end result of the Girvan–Newman algorithm is a dendrogram. As the Girvan–Newman algorithm runs, the dendrogram is produced from the top down (i.e. the network splits up into different communities with the successive removal of links). The leaves of the dendrogram are individual nodes.

3. Hierarchical clustering

In data mining and statistics, hierarchical clustering, also known as Hierarchical Cluster Analysis (HCA) is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical clustering generally fall into two categories:

I. ***Agglomerative***: This is a "bottom up" approach wherein each observation starts in its own cluster and pairs of clusters are merged as one moves up in the hierarchy.
II. ***Divisive:*** This is a "top down" approach wherein all observations start in one cluster and splits are performed recursively as one moves down in the hierarchy.

In general, the merges and splits are determined in a greedy manner. The results of hierarchical clustering are usually presented in a dendrogram.

In general case, the complexity of agglomerative clustering is O(n2log(n)) which makes it too slow for large data sets. Divisive clustering with an exhaustive search is O(2n), which is even worse.

4. Modularity

Modularity is the fraction of edges that fall within given groups minus the expected fraction if edges are distributed at random. The value of the modularity lies in the range of -1/2 and +1. It is

positive if the number of edges within group exceeds the number expected on the basis of a chance. For a given division of the network's vertices into some modules, modularity reflects the concentration of edges within modules compared with random distribution of links between all nodes regardless of modules.

CONCLUSION

The data exhibits a lot of characteristics and associations in an online social network and as a result of it, there are countless ways to uncover the hidden community structure. This community detection in online social networks takes us through a varied featured community structure and accordingly provides a broader way for a community analyst to analyze the data over varied data dimensions.

REFERENCES

[1] M. Girvan and M. Newman, "Community structure in social and biological networks",Proceedings of the National Academy of Sciences of the United States of America, 99(12):7821–7826, 2002.

[2] Goldberg, A. V., &. Trajan, R. E. (1988) A New Approach to the Maximum Flow Problem, J

[3] Donath, W. E. and Hoffman, A. J. (1973). Lower bounds for the partitioning of graphs. IBM J. Res. Develop. 17 420–425. MR0329965

[4] Andrew, Y, Ng and et al. (2001). "Natural and Synthetic", NIPS'01 Proceedings of 14th International Conference on Neural Information Processing Systems:,pp: 849-856.

[5] Kirkpatrick, S., Gelatt, C. D., &Vecchi, M. P. (1983). Optimization by simulated annealing. science, 220(4598), 671-680.

[6] Guimera, R., Sales-Pardo, M., &Amaral, L. A. N. (2004). Modularity from fluctuations in random graphs and complex networks. Physical Review E, 70(2), 025101.

[7] Guimera, R., &Amaral, L. A. N. (2005). Functional cartography of complex metabolic networks. nature, 433(7028), 895.

[8] Boettcher, S., &Percus, A. G. (1999, July). Extremal optimization: Methods derived from co-evolution. In Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation-Volume 1 (pp. 825-832). Morgan Kaufmann Publishers Inc..

[9] JordiDuch and Alex Arenas. Community identification using extremal optimization. Physics Review E, 72(027104), 2005.

[10] Richardson, T., Mucha, P. J., & Porter, M. A. (2009). Spectral tripartitioning of networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *80*(3), [036111]. DOI: 10.1103/PhysRevE.80.036111

[11] Slanina, F., & Zhang, Y. C. (2005). Referee networks and their spectral properties. In ActaPhysicaPolonica B (p. 2797).

[12] Mitrović, M., &Tadić, B. (2009). Spectral and dynamical properties in classes of sparse networks with mesoscopicinhomogeneities. Physical Review E, 80(2), 026123.

[13] Alves, N. A. (2007). Unveiling community structures in weighted networks. Physical Review E, 76(3), 036101.

[14] Wu, F. Y. (1982). The potts model. Reviews of modern physics, 54(1), 235.

[15] Reichardt, J., &Bornholdt, S. (2004). Detecting fuzzy community structures in complex networks with a Potts model. Physical Review Letters, 93(21), 218701.
[16] Hughes, B. D. (1995). Random walks and random environments.
[17] Zhou, H., &Lipowsky, R. (2005). Dynamic pattern evolution on scale-free networks. Proceedings of the National Academy of Sciences, 102(29), 10052-10057.
[18] Weinan, E., Li, T., &Vanden-Eijnden, E. (2008). Optimal partition and effective dynamics of complex networks. Proceedings of the National Academy of Sciences, 105(23), 7907-7912.
[19] Pikovsky, A., Rosenblum, M., Kurths, J., &Kurths, J. (2003). Synchronization: a universal concept in nonlinear sciences(Vol. 12). Cambridge university press.
[20] Boccaletti, S., Ivanchenko, M., Latora, V., Pluchino, A., &Rapisarda, A. (2007). Detecting complex network modularity by dynamical clustering. Physical Review E, 75(4), 045102.
[21] Palla, G., Derényi, I., Farkas, I., &Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. Nature, 435(7043), 814.
[22] Farkas, I., Ábel, D., Palla, G., &Vicsek, T. (2007). Weighted network modules. New Journal of Physics, 9(6), 180.
[23] J. M. Kumpula, M. Kivela, K. Kaski, J. Saramaki, "Sequential algorithm for fast clique percolation," Physical Review E, Vol. 78, p. 026109, 2008.
[24] Zhang S., Wang R.S., Zhang X.S, Identification of overlapping community structure in complex networks using fuzzy c-means clustering, Physics A, Vol. 374, 2007, pp. 483-490.
[25] Nepusz Y., Petrocz A., Negyessy L., and Bazso F, Fuzzy communities and the concept of bridgeness in complex networks, Physics Review E 77, 2008, pp. 016107.
[26] Wang X., Jiao L., and Wu J., Adjusting from disjoint to overlapping community detection of complex networks, Physica A., Vol. 388, 2009, pp. 5045–5056.
[27] Gregory S, An algorithm to find overlapping community structure in networks, In Proc. the 11th PKDD, Sept. 2007,pp.91-102
[28] Gregory S , Finding overlapping communities in networks by label propagation. New Journal of Physics, vol. 12, 2010, pp.103018.
[29] Xie, J., Szymanski, B. K., and Liu X, SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process, In Proceedings of IEEE ICDM Workshop on DMCCI, 2011, pp. 344–349.
[30] Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. Physical review E, 76(3), 036106.
[31] Baumes J., Goldberg M., Krishnamoorthy M., Magdon-Ismail M., and Preston N, Finding communities by clustering a graph into overlapping subgraphs, In Proceedings of IADIS Applied Computing, 2005, pp. 97–104.
[32] Lancichinettia A., Fortunato S, Community detection algorithms: A comparative analysis, Physics Review E, Vol. 80, 2009, 056117.
[33] Goldberg, Y., &Elhadad, M. (2010, June). Easy first dependency parsing of modern Hebrew. In Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages (pp. 103-107). Association for Computational Linguistics.
[34] Goldberg M., Kelley S., Magdon-Ismail M., Mertsalov K., and Wallace A, Finding overlapping communities in social network, SocialCom 2010.
[35] Lancichinetti A., Radicchi F., Ramasco J. J., and Fortunato S., Finding statistically significant communities in networks, PLoS ONE, 2011, Vol. 6(4): e18961.

[36] Cazabet R., Amblard F., and Hanachi C, Detection of overlapping communities in dynamical social networks, In Proceedings of SOCIALCOM, 2010, pp. 309–314.
[37] https://en.wikipedia.org/wiki/Clique_percolation_method#cite_note-cpm_nature-1
[38] http://www.cfinder.org/