COMMUNITY DETECTION IN SOCIAL NETWORKS: PROMISING ALGORITHMS

[1]Anupama Chowdhary
Principal, Keen College, Bikaner (Rajasthan), India

Abstract – Online social network is a universal set of sets of people connected via each other densely and in turn these set are connected with each other sparsely. In the graph structure these dense sets are known as communities. Although in literature there is a variety of community definitions based on network structures. Community detecting methods that have appeared in literature are even larger. In this paper we have discussed promising algorithms to detect communities in social networks.

Index terms – centrality, betweenness, modularity, graph partition, hierarchical clustering, spectral clustering, potts model, random walk.

I. INTRODUCTION

Online social networks, such as Twitter, Facebook, have gained huge amount of popularity in this decade. In India there are 23.2 million Twitter users, 194.11 million Facebook users. For world this figure goes to 330 billion for Twitter users, 2 billion for Facebook users. Instagram and Whatsapp are other popular social networks having 800 million and 1.3 billion active users worldwide respectively.

These active users share their information such as personal profiles, holiday photos, and social relationships. As the individual this is a very little data but social network users normally creates groups or follow others and thus are organized in communities. User's behaviour is majorly influenced by these communities rather than his/her friends. Detecting clusters or communities in large real-world graphs such as large social or information networks is a problem of considerable interest. The first analysis of community structure was carried out by Weiss and Jacobson [1], who searched for work groups within a government agency.

A community could be loosely described as a collection of vertices within a graph that are densely connected among them while being loosely connected to the rest of the graph [2]. Community detection is a process of finding a group of users who interact on elements such as published work (stories, write-ups, abstracts etc.), comments, photos, tags and many more. To detect these communities in social networks we need algorithms. This paper reviews about the different community detection algorithms.

II. GENERAL CONCEPTS

Real-world network is a huge network and sometimes, even $O(n^2)$ space and time complexity is unbearable. Network data is basically discrete, so we need algorithms that use the graph properties directly (k-clique, quasi-clique, vertex-betweenness, edge-betweeness etc.). To measure inter-community elementary methods are edge betweenness and random walk betweenness. Following terminology will be of great help to study and analyse the community structure in social networks.

Walk is defined as a sequence of alternating vertices and edges. Walk starts with the source vertex and ends at the target vertex. If the starting vertex is the same as the ending vertex a walk is a **Closed Walk.** A **Trail** is defined as a walk with no repeated edges. A **Path** is defined as an open trail with no repeated vertices. A geodesic is a shortest path between two vertices. A **random walk** is a random process, which describes a path that consists of a succession of random steps.

Indicators of **Centrality** identify the most important vertices within a graph, this helps to identifying the most influential person(s) in a social network, key infrastructure nodes in the internet etc. Centrality concepts were first developed in social network analysis, and many of the terms used to measure centrality reflect their sociological origin [3].

The **betweenness** centrality for each vertex is the number of shortest paths that pass through the vertex; it was devised as a general measure of centrality [4]. In other words betweenness is a variable expressing the frequency of the participation of edges to a process. Edge betweenness is the number of shortest paths between all vertex pairs that run along the edge.

Modularity was designed to measure the strength of division of a network into modules or communities. Networks with high modularity have dense connections between the vertices within modules but sparse connections between vertices in different modules. There are different methods for calculating modularity [5].

III. ALGORITHMS FOR COMMUNITY DETECTION IN SOCIAL NETWORKS

In extremely large network size it is difficult to identifying meaningful community structure in social networks. Sparseness of the networks contributes in the difficulty of the problem. 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 finding have been developed and employed with varying levels of success [6]. Some of the algorithms for community detection in social networks are briefly discussed in this paper.

Graph partition methods

Graph partition is a NP-hard problem so practical solutions are based on heuristics and approximations. This is a type of minimum-cut method, where minimum-cut of a graph is a partition of the vertices of a graph into two disjoint subsets that are joined by at least one edge. One of the oldest algorithms was given by Kernighan-Lin [7] for partition the graph into two equal and unequal sizes of subsets. The complexity of algorithm was $O(n^2 \log n)$, where n is number of vertices. This algorithm was modified by Bruce et al. [8] and was applied periodically to refine the partition.

Fiduccia-Mattheyses [9] provide the algorithm with complexity O(P), where P is total number of terminals. It is an iterative min-cut heuristic for partitioning networks its worst case computation time, per pass, grows linearly with the size of the network.

Goldberg-Tarjan [10] use the dynamic tree data structure and achieve $O(nm \log(n^2/m)$ time complexity, where n – number of vertices

and m – number of edges in a graph.

Flake et al. [11a, b] compute point-to-point shortest path throughout the graph and use max flow min-cut theorem. They propose Incremental Shortest Augmentation (ISA) algorithm. For calculation purpose they treated web graph as undirected.

Hierarchical clustering

In this method one defines a similarity measure quantifying some (usually topological) type of similarity between node pairs. Some commonly used measures are cosine similarity, the Jaccard index, the Hamming distance between rows of the adjacency matrix. These measures are used to group similar nodes into communities. Mostly hierarchical clustering methods adopt agglomerative "Bottom-up" clustering process. In the beginning of the agglomerative clustering process, each element is in a cluster of its own. The clusters are then sequentially combined into larger clusters, until all elements end up being in the same cluster. At each step, the two clusters separated by the shortest distance are combined.

R. Sibson [12] proposed SLINK algorithm based on single-linkage with time complexity $O(n^2)$ and space complexity $O(n)$. Gower and Ross [13] used Prism's algorithm, in a variation without binary heaps, that takes time $O(n^2)$ and space $O(n)$ to construct the minimum spanning tree (but not the clustering) of the given items and distances. R. Defays [14] proposed CLINK algorithm based on complete-linkage with time complexity $O(n^2)$ inspired by the similar algorithm SLINK.

Sokal R and Michener [15] proposed an algorithm based on group average that constructs a rooted tree (dendrogram) that reflects the structure present in a pairwise similarity matrix (or dissimilarity matrix). At each step, the nearest two clusters are combined into a higher-level cluster with time complexity $O(n^3)$. Using a heap for each cluster to keep its distances from other cluster reduces its time to $O(n^2 \log n)$. Fionn Murtagh presented some other approaches for special cases, a $O(k3^k n^2)$ time algorithm by Day and Edelsbrunner [16] for k-dimensional data that is optimal $O(n^2)$ for constant k, and another $O(n^2)$ algorithm for restricted inputs, when "the agglomerative strategy satisfies the reducibility property."[17]

Sokal and Michener [18] gives an algorithm that constructs a rooted tree (dendrogram) to reflect the structure present in a pairwise similarity matrix (or dissimilarity matrix). At each step, the nearest two clusters, namely p and q , are combined into a higher-level cluster $p \cup q$. Then, its distance to another cluster s is simply the arithmetic mean of the distances between s and members of $p \cup q$.

Ward's minimum variance method can be defined and implemented recursively by a Lance–Williams algorithm. At each step, it is necessary to optimize the objective function (find the optimal pair of clusters to merge). The recursive formula simplifies finding the optimal pair. The complexity of this method is $O(n^2)$ $O(n \log n)$[19][20].

Divisive Algorithms

Divisive algorithms use "Top-Down" analysis clustering, here initially all the nodes in a network are placed in a single cluster based on the intuition that edges with the highest betweenness values are 'bridges' between communities, groups are subdivided into two group recursively until each node of the network forms a separate cluster of its own. Divisive clustering algorithms provide clearer insights of the main structure of the data, but the larger clusters are generated at the early stage of the clustering process, so we have to rely on heuristic methods.

Girvan and Newman [21] focused on the concept of betweenness centrality. The betweenness of all edges of the graph can be calculated in a time that scales as $O(mn)$, or $O(n^2)$ on a sparse graph, with techniques based on breadth-first-search[22][23][24]. If the signals flow across random the betweenness of an edge is given by the frequency of the passages across the edge of a random walker running on the graph (random-walk betweenness). Calculation of random-walk betweenness requires the inversion of a $n \times n$ matrix (once), followed by obtaining and averaging the flows for all pairs of nodes. The first task requires a time $O(n^3)$, the second $O(mn^2)$, for a total complexity $O(n^3)$ for a sparse matrix [25]. Grivan-Newman algorithm was further modified by

Tyler et al. [26] they proposed to calculate the contribution to edge betweenness only from a limited number of centres, chosen at random.

Rattigan et al. [27][28] provide a quick approximation of the edge betweenness values using a network structure index, which consists of a set of vertex annotations combined with a distance measure Chen and Yuan [29] proposed to count only non-redundant paths. Holme et al. [30] have used a modified version of the algorithm in which vertices, rather than edges, are removed.

Pinney and Westhead [31] have proposed a modification of the algorithm in which vertices can be split between communities, to find overlapping communities.

Gregory [32] has proposed a similar approach, named CONGA

Modularity Methods

Modularity is a benefit function that measures the quality of a particular division of a network into communities. The modularity maximization method detects communities by searching over possible divisions of a network for one or more that have particularly high modularity. Since exhaustive search over all possible divisions is usually intractable, practical 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 [33][34].

Newman [35] proposed greedy method to maximize modularity. Their algorithm removes edges from the network to split it into communities, the edges removed being identified using one of a number of possible "betweenness" measures and these measures are recalculated after each removal. They also propose a measure for the strength of the community structure found by their algorithms, which gives an objective metric for choosing the number of communities into which a network should be divided.

Guimera et al. [33][34] used simulated annealing for modularity optimization as simulated annealing can give richer framework for heuristic algorithms [36]. Simulated annealing is a stochastic optimization technique that enables to find 'lowcost' configuration without getting trapped in 'high-cost' local minima. Rosvall and Bergstrom [37] optimally compressed the information on the structure of the graph, the optimization of the function is carried out via simulated annealing.

Boettcher et al. [38] used Extremal optimization that complements simulated annealing and this technique was used for modularity by Duch et al. [39]

Spectral Clustering

These techniques make use of the spectrum (eigenvalues) of the similarity matrix of the data to perform dimensionality reduction before clustering in fewer dimensions. 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.

Wang et al. [40] provide the method that combines spectral techniques, a vector partition problem, and the concept of modularity and is a natural extension of bi-partitioning to multiple eigenvectors. If the eigenvectors is taken corresponding to the two largest eigenvalues, then a split of the graph in three clusters can be obtained. Richardson et al. [41] used this technique to partition the graph in three splits with large modularity.

Slanina et al. [42] shown that eigenvectors of the adjacency matrix may be localized, if the graph has a clear community structure. Mitrovic and Tadic [43] study along the line structure–spectra–random-walks and find the relationships between different structural elements of the network and their spectra and the dynamics. Alves [44] provide spectral algorithm, its computational cost is limited to methods in computing the eigenvalues and eigenvectors of symmetric matrices initially it needs $O(n^3)$ operations, with subsequent less expensive iterations $O(n^2)$.

Donetti and Mũnoz [45] focused on the eigenvectors of the Laplacian matrix. The complexity of their algorithm is $O(Cn^2)$.

Donath and Hoffmann [46] contributed first on spectral clustering in 1973. Andrew et. al [47] have analysed the algorithm of spectral clustering as the ideal case and the general case.

Random Walk

A random walk of length k on a possibly infinite graph G with a root say r is a stochastic process with random variables {\displaystyle X_{1},X_{2},\dots ,X_{k}}$AA_1, A_2, \dots, A_k$ such that {\displaystyle $X_{1}=0}AA_1 = 0$ and {\displaystyle {X_{i+1}}}AaA_{i+1} is a vertex chosen uniformly at random from the neighbours of {\displaystyle X_{i}} A_i. Then the number {\displaystyle p_{v,w,k}(G)} $p_{u,v,k}(G)$ is the probability that a random walk of length k starting at u ends at v.

Pascal Pons and Matthieu Latapy [48] propose Walktrap algorithm based on random walk, which runs in time $O(mn^2)$ and space (n^2) in the worst case, and in time $O(n^2 \log n)$ and space $O(n^2)$ in most real-world cases. Here n – number of vertices and m – number of edges.

Rosvall and Bergstrom [49] optimally compressed the information on the structure of the graph; the optimization of the function is carried out via Minimum Description Length [50] of the random walk.

Zhou-Lipowsky [51] measured the proximity index between two nearest-neighboring vertices of a network by a biased Brownian motion (random walk) on the network with computational time $O(n^3)$. Based on this proximity measure, they constructed Netwalk algorithm.

Weinan et al. [52] propose a strategy for partition of a graph in k-communities along the lines of optimal prediction for the Markov chains associated with the dynamics on networks, 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.

Potts Model

Phase space of the Potts model is divided into three regions; ferromagnetic, super-paramagnetic and paramagnetic phases. The region of interest is that corresponding to the super-paramagnetic one, where domains of aligned spins appear. 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.

Ronhovde and Nussinov [53] proposed a method based on the minimization of the Hamiltonian of a Potts-like spin model, where the spin state represents the membership of the node in a given community. The method is rather fast, its complexity is slightly super-linear $O(L^{1.3})$ for community detection and $O(L^{1.3} \log N)$ for the multi-resolution algorithm, where L is the number of edges and N is the number of nodes in the system.

Reichardt and Bornholdt [54] discuss an algorithm for community detection in complex networks based on a modified q-state Potts model. Communities appear as domains of equal spin value near the ground state of the system, which is approximated through Monte-Carlo optimization. Only local information is used to update the spins which makes parallelization of the algorithm straightforward and allows the application to very large networks.

IV. CONCLUSION

General concepts related to social media and community detection are discussed in the beginning of the paper. Further we have classified the available community detection algorithms in seven categories and briefly discussed the algorithms or methods under these categories. In this paper graph partition, hierarchical clustering, divisive algorithms, modularity methods, spectral clustering, random walk, and Potts model based methods and algorithms are illustrated.

REFERENCES

[1] R.S. Weiss, E. Jacobson, A method for the analysis of the structure of complex organizations, Am. Sociol. Rev. 20 (1955) 661668.

[2] J. Bagrow and E. Bolt, "Local method for detecting communities". Physical Rev. E., Volume 72 No. 4, p 046108, 2015.

[3] Newman, M.E.J. 2010. Networks: An Introduction. Oxford, UK: Oxford University Press.

[4] Freeman, Linton (1977). "A set of measures of centrality based on betweenness". Sociometry. 40: 35–41. doi:10.2307/3033543

[5] Newman, M. E. J. (2006). "Modularity and community structure in networks". Proceedings of the National Academy of Sciences of the United States of America. 103 (23): 8577–8696. arXiv:physics/0602124

[6] M. A. Porter; J.-P. Onnela; P. J. Mucha (2009). "Communities in Networks" . Notices of the American Mathematical Society. 56: 1082–1097, 1164–1166.

[7] Kernighan, B.W.; Lin, S., "An Efficient Heuristic Procedure for Partitioning Graphs" Bell System Technical Journal, 49: 2. February 1970 pp 291-307.. ()

[8] Hendrickson, B.; Leland, R. (1995). A multilevel algorithm for partitioning graphs. Proceedings of the 1995 ACM/IEEE conference on Supercomputing. ACM. p. 28.

[9] Fiduccia, C. M., & Mattheyses, R. M. (1988, June). A linear-time heuristic for improving network partitions. In Papers on Twenty-five years of electronic design automation (pp. 241-247). ACM.

[10] A V Goldberg , R E Tarjan, "A new approach to the maximum flow problem", Journal of the ACM 35, 921, 1988

[11] Flake, G. W., S. Lawrence, and C. L. Giles, "Efficient Identification of Web Communities", Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM Press, Boston, USA),pp. 150-160, 2000

[12] Flake, G. W., S. Lawrence, C. Lee Giles, and F. M. Coetzee, "Self-Organization and Identification of Web Communities" IEEE Computer 35, 66, 2002

[13] R. Sibson (1973). "SLINK: an optimally efficient algorithm for the single-link cluster method" (PDF). The Computer Journal. British Computer Society. 16 (1): 30–34. doi:10.1093/comjnl/16.1.30.

[14] Gower, J. C.; Ross, G. J. S. (1969), "Minimum spanning trees and single linkage cluster analysis", Journal of the Royal Statistical Society, Series C, 18 (1): 54–64, JSTOR 2346439, MR 0242315.

[15] D. Defays (1977). "An efficient algorithm for a complete link method" (PDF). The Computer Journal. British Computer Society. 20 (4): 364–366. doi:10.1093/comjnl/20.4.364.

[16] Sokal R and Michener C (1958). "A statistical method for evaluating systematic relationships". University of Kansas Science Bulletin. 38: 1409–1438.

[17] Day, William H. E.; Edelsbrunner, Herbert (1984-12-01). "Efficient algorithms for agglomerative hierarchical clustering methods". Journal of Classification. 1 (1): 7–24. ISSN 0176-4268. doi:10.1007/BF01890115.

[18] Murtagh F (1984). "Complexities of Hierarchic Clustering Algorithms: the state of the art". Computational Statistics Quarterly. 1: 101–113.

[19] Clauset, Aaron; Cristopher Moore; M. E. J. Newman (2008-05-01). "Hierarchical structure and the prediction of missing links in networks". Nature. 453(7191): 98–101. ISSN 0028-0836. PMID 18451861. doi:10.1038/nature06830.

[20] Ward, J. H., Jr. (1963), "Hierarchical Grouping to Optimize an Objective Function", Journal of the American Statistical Association, 58, 236–244

[21] F. Murtagh, "Expected time complexity results for hierarchal clustering algorithms which use cluster centres", Information Processing letters 16(1983) 237-241, North-Holland.

[22] Girvan M. and Newman M. E. J., "Community structure in social and biological networks", Proc. Natl. Acad. Sci. USA 99, 7821–7826 (2002)

[23] M.E.J. Newman, M. Girvan, "Finding and evaluating community structure in networks", Phys. Rev. E 69 (2) (2004) 026113.

[24] U. Brandes, "A faster algorithm for betweenness centrality", J. Math. Sociol. 25 (2001) 163-177.

[25] T. Zhou, J.-G. Liu, B.-H. Wang," Notes on the calculation of node betweenness", Chin. Phys. Lett. 23 (2006) 2327-2329

[26] M.E.J. Newman," A measure of betweenness centrality based on random walks", Soc. Netw. 27 (2005) 39-54

[27] J.R. Tyler, D.M. Wilkinson, B.A. Huberman, "Email as spectroscopy: Automated discovery of community structure within organizations", in: Communities and Technologies, Kluwer, B.V., Deventer, The Netherlands, 2003, pp. 81-96.

[28] M.J. Rattigan, M. Maier, D. Jensen, "Graph clustering with network structure indices", in: ICML '07: Proceedings of the 24th International Conference on Machine Learning, ACM, New York, NY, USA, 2007, pp. 783-790

[29] M.J. Rattigan, M. Maier, D. Jensen, "Using structure indices for efficient approximation of network properties", in: T. Eliassi-Rad, L.H. Ungar,

[30] J. Chen, B. Yuan," Detecting functional modules in the yeast protein-protein interaction network", Bioinformatics 22 (18) (2006) 2283-2290

[31] P. Holme, M. Huss, H. Jeong, "Subnetwork hierarchies of biochemical pathways", Bioinformatics 19 (4) (2003) 532-538

[32] J.W. Pinney, D.R. Westhead, "Betweenness-based decomposition methods for social and biological networks", in: Interdisciplinary Statistics and Bioinformatics, Leeds University Press, Leeds, UK, 2006, pp. 87-90.

[33] S. Gregory, "An algorithm to find overlapping community structure in networks", in: Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, PKDD 2007, Springer-Verlag, Berlin

[34] L. Danon; J. Duch; A. Díaz-Guilera; A. Arenas (2005). "Comparing community structure identification". J. Stat. Mech. 2005 (09): P09008. doi:10.1088/1742-5468/2005/09/P09008

[35] R. Guimera; L. A. N. Amaral (2004). "Functional cartography of complex metabolic networks". Nature. 433 (7028): 895–900. PMC 2175124. PMID 15729348. doi:10.1038/nature03288

[36] M.E.J. Newman, M. Girvan, "Finding and evaluating community structure in networks", Phys. Rev. E 69 (2) (2004) 026113.

[37] Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing", Science, New Series, Vol. 220, No. 4598. (May 13, 1983), pp. 671-680.

[38] M. Rosvall and C. T. Bergstrom," An information-theoretic framework for resolving community structure in complex networks" Proc. Natl. Acad. Sci. USA 104, 7327 (2007).

[39] Boettcher, S., and A. G. Percus, "Optimization with Extremal Dynamics", Phys. Rev. Lett. 86, 5211, 2001

[40] Duch, J., and A. Arenas, "Community detection in complex networks using Extremal Optimization", Phys. Rev. E 72(2), 027104, 2005.

[41] Wang, G., Y. Shen, and M. Ouyang, "A vector partitioning approach to detecting community structure in complex networks", Comput. Math. Appl. 55(12), 2746, 2008

[42] Richardson, T., P. J. Mucha, and M. A. Porter, "Spectral tripartitioning of networks", 2009, Physical Review E, 80(3), 036111

[43] Slanina, F., and Y.C. Zhang, "Referee networks and their spectral properties", Acta Phys. Pol. B 36, 2797, 2005

[44] Mitrovic, M., and B. Tadic," Spectral and Dynamical Properties in Classes of Sparse Networks with Mesoscopic Inhomogeneities", Phys. Rev. E 80(2), 026123, 2009

[45] Alves, Nelson A. "Unveiling community structures in weighted networks." Physical Review E 76.3 (2007): 036101.

[46] L. Donetti and M. A. Mũnoz, "Detecting network communities: a new systematic and efficient algorithm", J. Stat. Mech. P10012 (2004).

[47] Donath, W., and A. Hoffman, "lower bounds for partitioning of graphs", IBM Journal of Research and Development 17(5), 420, 51973

[48] A.Y. Ng, M.I. Jordan, Y. Weiss, "On Spectral Clustering: Analysis and Algorithm", Stanford AI Lab.

[49] Pascal Pons and Matthieu Latapy, "Computing Communities in Large Networks Using Random Walks", Journal of Graph Algorithms and Applications http://jgaa.info/ vol. 10, no. 2, pp. 191–218 (2006)

[50] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure ", Proc. Natl. Acad. Sci. USA 105, 1118 (2008)

[51] P. D. Gr¨unwald, I. J. Myung, and M. A. Pitt, Advances in Minimum Description Length: Theory and Applications (MIT Press, Cambridge, USA, 2005)

[52] Zhou, H., and R. Lipowsky, "Network Brownian Motion: A New Method to Measure Vertex-Vertex Proximity and to Identify Communities and Subcommunities", Computational Science - ICCS 2004. Lecture Notes in Computer Science, vol 3038. Springer, Berlin, Heidelberg.

[53] Weinan, E., T. Li, and E. Vanden-Eijnden, "Optimal partition and effective dynamics of complex networks", Proc. Natl. Acad. Sci. USA 105, 7907, 2008

[54] P. Ronhovde and Z. Nussinov, Phys. Rev. E 80, 016109 (2009)

[55] Reichardt, J., and S. Bornholdt, "Detecting fuzzy community structures in complex networks with a Potts model", Phys. Rev. Lett. 93(21), 218701, 2004