Felfli et al. Applied Network Science (2019) 4:85
https://doi.org/10.1007/s41109-019-0216-2

Applied Network Science

RESEARCH Open Access

Community detection and unveiling of hierarchy in networks: a density-based clustering approach

Zineb Felfli*, Roy George, Khalil Shujaee and Mohamed Kerwat

* Correspondence: zfelfli@cau.edu
Department of Cyber Physical Systems, Clark Atlanta University, 223 J. Brawley drive at Fair street, SW, Atlanta, GA 30314, USA

Abstract

The unveiling of communities within a network or graph, and the hierarchization of its members that results is of utmost importance in areas ranging from social to biochemical networks, from electronic circuits to cybersecurity. We present a statistical mechanics approach that uses a normalized Gaussian function which captures the impact of a node within its neighborhood and leads to a density-ranking of nodes by considering the distance between nodes as punishment. A hill-climbing procedure is applied to determine the density attractors and identify the unique parent (leader) of each member as well as the group leader. This organization of the nodes results in a tree-like network with multiple clusters, the community tree. The method is tested using synthetic networks generated by the LFR benchmarking algorithm for network sizes between *500* and *30,000* nodes and mixing parameter between *0.1* and *0.9*. Our results show a reasonable agreement with the LFR results for low to medium values of the mixing parameter and indicate a very mild dependence on the size of the network.

Introduction

Many real world phenomena in the physical, biological and social sciences are complex systems which may be represented as networks. In the field of technology and communication, smart phones, tablets, and other mobile hardware platforms, blogs and instant messaging software and communication services, designed with interoperability and connected to national and global systems, create networks of massive scale and complexity. The analysis of these networks led to the emergence of network science, a multidisciplinary area of research focused on the understanding of information flows and the underlying topological, structural and dynamic aspects of networks. Community structure detection is an important component of network science devoted to the identification of the organizational structure of the network that results from the interactions between its components or members, or nodes in the language of graph theory, and the grouping of these nodes into clusters or communities. A network is said to have 'community structure' if a set of nodes have a higher probability of being linked together compared to nodes of other groups. This means that there exist natural divisions in the network that separate the densely connected nodes from nodes of other groups to which they are less likely to be connected.

Earlier methods of community detection, such as the Kernighan-Lin algorithm (Kernighan & Lin, 1970), the spectral bisection (Fiedler, 1973; Pothen et al., 1990) and hierarchical clustering (Scott, 2000) based on similarity measures do not lend themselves to real-world network data such as Internet and web data and biological and social networks (Newman, 2004). The first community detection algorithm that proved successful in this context was introduced by Girvan and Newman (Girvan & Newman, 2002) and turned the area of community detection to a main pillar of network science research (Schaub et al., 2017). It allowed for the identification of community structures in both social and biological networks by separating them into meaningful clusters. The Girvan-Newman algorithm identifies edges that lie between communities in a network and removes them thereby allowing for the identification of distinct communities in the network. Its limitation stems from the fact that it is relatively slow making it impractical for networks of more than a few thousand nodes. Many new approaches that encompass varying disciplines have been developed since then, including modularity methods (Newman & Girvan, 2004), Bayesian and regularized likelihood approaches (Hofman & Wiggins, 2008; Yan, 2016; Daudin et al., 2008), statistical hypothesis tests (Wang et al., 2017), and spectral methods (Krzakala et al., 2013; Saade et al., 2014).

Despite the number of methods that deal with community structure and community detection and the proliferation of a relatively large number of algorithms, community detection remains a challenging problem and the pursuit of new methodologies that can handle different types of raw data remains highly desirable and an active area of research. However, in the absence of ground truth, it is difficult to assess the quality of any given algorithm, therefore there is great need for robust and firmly established algorithms that serve as the gold standard, a benchmark against which any new algorithm can be measured and tested. One of the earlier and popular benchmarks is that of Girvan and Newman (Girvan & Newman, 2002), referred to as the GN benchmark, although it suffers a number of limitations including the fact that all nodes of the network have the same degree and that the communities are of the same size. This clearly does not describe real-world networks properly, as these are known to have a non-uniform degree distribution, generally following a power-law (Barabási, 2017; Barabási & Albert, 1999), and heterogenous communities. More recently, Lancichinetti, Fortunato and Radicchi developed a more realistic algorithm (Lancichinetti et al., 2008) that assumes power-law distributions for the network degrees and the communities as well as allows for overlapping communities. Today, the algorithm which is referred to as the LFR benchmark, represents the new gold standard for evaluating the performance of newly developed community detection algorithms. Moreover, the LFR algorithm yields true community labels allowing for a simple and easy comparison of communities obtained with different algorithms, thereby making it an ideal benchmark model.

Our proposed methodology was illustrated in (Felfli et al., 2018) using a synthetic network which consists of 20 nodes. While the small size of the network helps to clearly delineate the steps involved in the clustering process, it does not test its limits. In the present paper, we use larger networks generated by the LFR benchmark in an attempt to establish the validity and utility of our approach and identify its limitations.

Communities, or clusters, can be determined mathematically by identifying density-attractors which are local maxima of the overall density (Hinneburg & Keim, 1998).

Our approach assumes a Gaussian density distribution which is constructed so as to unveil the relative importance (influence) of the various nodes (members) of the network, allowing for the identification of the immediate leader of every member and hence the ranking of all members including the emergence of the group leader. A tree-like network with multiple clusters emerges. In each cluster, the nodes are ranked according to their density value, which for a given node reflects its certainty in attracting interaction with all nodes in the network.

This *clustering* procedure requires the knowledge of the distance metric, i.e. a mapping of the network and its topology via the distances between nodes. These distances can be expressed in terms of path lengths, namely the smallest number of links or edges needed to connect the nodes within the network. Here, we model the interactions among nodes based on the concept of randomized shortest path (RSP) dissimilarity (Kivimäki et al., 2014; Yen et al., 2008; Saerens et al., 2009; Francoisse et al., 2017), which has its foundation in statistical physics and is derived with the constraint of fixed relative "distance" entropy. It was originally inspired by the work of Akamatsu (Akamatsu, 1996) on the decomposition of path choice entropy in g\eneral transport networks, and motivated by the fact that, generally common distances do not take into account the global structure of the graph (Francoisse et al., 2017).

The process of randomization assigns a probability distribution over the set of paths to be followed by each node, so that chance is favored over choice with the advantage of exploration versus efficiency. This reveals the degree of connectedness between nodes where two nodes are considered highly connected or related when they are linked by many, preferably low-cost, paths (Francoisse et al., 2017). The calculation involves the search for the optimal path that minimizes the expected cost obtained by imposing the constraint that the relative entropy has a constant value spread throughout the network. The knowledge of the dissimilarities for the whole network allows for the clustering and ranking of the nodes through the mapping of the resulting distance matrix into a Gaussian kernel matrix.

The paper is organized as follows: in Section 2, the details of the methodology are highlighted; Section 3 presents and discusses the results of benchmarking; Section 4 concludes the paper, evaluates the performance of our algorithm and addresses its limitations.

Approach

The first step in the analysis is to map the network and its topology via the distances between nodes. This step provides a fundamental starting point for interpreting the network and a powerful tool for further exploration of its characteristics using standard multivariate statistics or machine learning methods. To explore the structure and dynamics of the network, we start by modeling the interactions among nodes based on the concept of randomized shortest path (RSP) dissimilarity (Kivimäki et al., 2014; Yen et al., 2008; Saerens et al., 2009; Francoisse et al., 2017). The calculation involves the search for the optimal path that minimizes the expected cost obtained by imposing the constraint that the relative entropy has a constant value spread throughout the network.

The expected cost C is defined in (Kivimäki et al., 2014):

$$C(P_{if}) = \sum_{r\epsilon\mathcal{R}_{if}} P_{if}(r)C(r\,), \tag{1}$$

where the path r that the walker follows in going from node i to node f, belongs to all sets of paths $\mathcal{R}_{if}$ that link these nodes. The relative entropy is simply defined as the Shannon entropy which has the form $-\sum_i p_i \ log\ p_i$ calculated relative to the reference probability, $P_{if}^{ref}(r\,)$ which is the product of the transition probabilities in going from node i to node f along the path r, and $C(r)$ is the cost for path r, namely,

$$H = \sum_{r\epsilon\mathcal{R}_{if}} P_{if}\ (r)\ \ \log P_{if}(r)/P_{if}^{ref}(r\,) \tag{2}$$

The minimization process constrained by the constant value of the entropy and $\sum_{r\in\mathcal{R}_{if}} P_{if}\ (r) = 1$ leads to a Gibbs-Boltzmann distribution with partition function Z and an optimal distribution

$$P_{if}^{RSP}(r) = \frac{P_{if}^{ref}(r)\ \exp(-\beta\ \mathrm{C}(r\,))}{Z_{if}} \tag{3}$$

where Z_{if} is given by:

$$Z_{if} = \sum_{r\epsilon\mathcal{R}_{if}} P_{if}^{ref}(r)\ exp(-\beta\ C(r\,)) \tag{4}$$

The parameter β, which controls the distribution, plays the role of the inverse-temperature in thermodynamics. It is shown (Francoisse et al., 2017) that, under the Gibbs-Boltzmann distribution, the probability of drawing a path connecting two nodes can easily be computed in closed form by simple matrix inversion. Moreover, in contrast to common distance measures, such as the Shortest Path (SP) (the length of the shortest paths between nodes), and the Commute Time (CT) distance (Akamatsu, 1996) (the expected length of paths that a random walker moving along the edges of the graph takes from one node to the other and back (Kivimäki et al., 2014)), RSP captures the global structure of the network. This is because it is designed as an interpolation between the SP and CT distances, thus preserving their advantages while avoiding their drawbacks. The measure is also shown in experimental results on semi-supervised classification to be competitive with other state-of-the-art approaches (Francoisse et al., 2017).

Once the dissimilarities are computed for the whole network, the ranking of the nodes is determined. The distance matrix is mapped into a Gaussian kernel matrix which is a non-linear Euclidean distance (a radial basis function (RBF) kernel and commonly used in Support Vector Machine classification). An interesting property of this kernel function is that it decreases with distance and ranges between zero and one rendering it a useful metric for weighting observations. In fact, because of its limiting values, it can readily be interpreted as a similarity measure or a density function.

Our density function first introduced in (Bahrami Bidoni & George, 2014) considers the distance between nodes as punishment and captures the impact of a node within its neighborhood. It is therefore equivalent to *an influence* (Hinneburg & Keim, 1998) function which allows for the grouping and ranking of the nodes within the network.

The influence function is applied to each node and the overall density of the network can be expressed as the sum of the influence function of all nodes:

$$f(i,j) = exp\left(\frac{-d(i,j)^2}{2\sigma_j{}^2}\right), \tag{5}$$

where d(i,j) is a distance function between nodes i and j and σ is the parameter. This function must be reflexive and symmetric (Hinneburg & Keim, 1998) therefore we use a symmetrized form of the RSP, the RSP dissimilarity Δ^{RSP}.

Our normalized Gauss influence function takes the form

$$\rho(i) = \sum_{j=1}^{k} exp\left(\frac{-\left(\Delta_{ij}^{RSP}\right)^2}{2\sigma_j^2}\right) / \sum_{i=1}^{k}\sum_{j=1}^{k} exp\left(\frac{-\left(\Delta_{ij}^{RSP}\right)^2}{2\sigma_j^2}\right) \tag{6}$$

The variance σ_j that appears in the Gauss influence function denotes the j^{th} kernel scale-parameter and reflects the impact of node j on its neighborhood. It is optimized internally within the code.

The determination of the density attractors is based on a hill climbing procedure, a local search which determines the solution iteratively. To initiate the climbing procedure, the RSP values for each pair of nodes (distance cost) are sorted in ascending order which leads to a NxN matrix, with each row 'i' representing the closest to the furthest node j from the ith node. Here N is the size of the network. Now, for each node i, we loop over j searching for the closest node that ranks higher than i (with respect to its density) with the constraint that it has a direct link to i. This leads to the identification of the parent of each node, thereby determining a tree-like network, from which the communities are identified. To illustrate this outcome, we show in Fig. 1 the results of our clustering method for LFR- benchmark graphs with $N = 250$ and mixing parameter, $\mu = 0.1, 0.3$ and 0.5.

Results

Networks ranging in size from $N = 500$ through $N = 30{,}000$ were generated using the LFR benchmark (Lancichinetti et al., 2008). All calculations were performed on a Windows server with a dual Intel-Xeon processor and 1536 GB of RAM. The LFR algorithm requires the input of the two exponents, β and γ which control respectively, the power

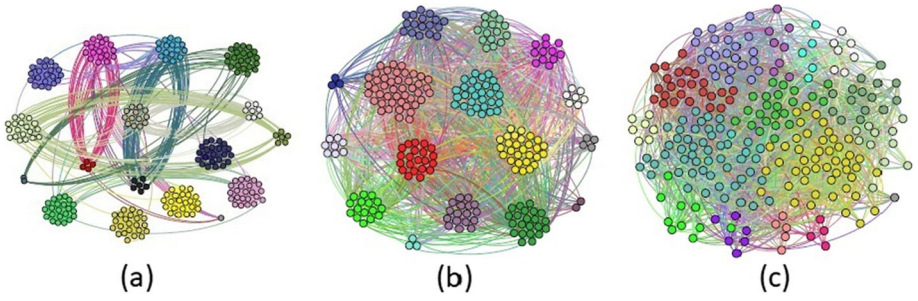

Fig. 1 Results of our clustering algorithm corresponding to LFR graphs with $N = 250$ and (**a**) $\mu = 0.1$, (**b**) $\mu = 0.3$ and (**c**) $\mu = 0.5$

law that governs the network degrees and that for the communities present in a given network. This is consistent with the scale-free nature of most real networks (Barabási, 2017). Moreover, in a real-world network members or nodes do not necessarily belong to a single community. This fact is taken into account in the LFR algorithm via a *mixing parameter* μ, which reflects overlap between communities, i.e. the fraction of a node's links or edges that are external to its assigned cluster. It is clearly more difficult to detect clusters with increasing μ as this corresponds to poorly defined boundaries between clusters. Different clustering algorithms may lead to different clustering assignments. A number of similarity metrics have been introduced in order to more efficiently quantify the performance of a given algorithm against the gold standard, here the LFR benchmark. The most frequently used metrics are the normalized mutual information (NMI) based on the Shannon entropy of information theory, and the adjust Rand index (ARI) which counts the number of pairs of nodes assigned to the same cluster, then normalizes the result to the total number of pairs of nodes within the network. Another version of the NMI is the adjusted mutual information (AMI) which we use in the present work along with the ARI, and two other metrics, the homogeneity score (HS) and completeness score (CS). The HS expresses the degree to which all clusters contain only nodes which are members of a single class. CS reflects the degree to which all nodes that are members of a given class are members of the same cluster. All metrics are normalized so as to fall between a value of zero and one for no agreement and perfect agreement respectively, with the gold standard.

The first part of the algorithm extracts from the frequency matrix the relevant information to calculate of the cost function and the randomized shortest path dissimilarity. In the calculation of the RSP dissimilarity, the parameter β must be provided instead of the relative entropy; the value used here is $\beta = 0.9$. The density value of each node is determined, the density attractors extracted allowing for the identification of the clusters. and their respective leaders. Within each cluster, the unique parent of each node is identified leading to a ranking of all nodes. We generated directed unweighted networks using the LFR benchmark for $\beta = 3$, $\gamma = 2$ and with N ranging from 500 to $30{,}000$. The dependence of the average degree on the network size and, in the case of scale free networks, to the degree distribution exponent is quite loose (Orman et al., 2013). Therefore, we considered values of the average degree, $<k>$ ranging from 3 through 30, then selected those that yielded optimal mutual information scores. The results are depicted in Fig. 2.

The agreement with the results of the LFR benchmark, quantified by four similarity measures, are virtually independent of the size of the network (for the sizes considered here), and deteriorate considerably for values of the mixing coefficient starting at $\mu = 0.5$. The benchmarking procedure allowed us to assess the quality of our present formulation. Work on extending our methodology so that overlapping communities are taken into account, is in progress. The aim is to achieve results with greater accuracy, particularly when the mixing between communities becomes important.

Discussion and conclusions

In the present work we propose an approach to community detection which inherently leads to hierarchization within the network. A density-based formulation, which permits

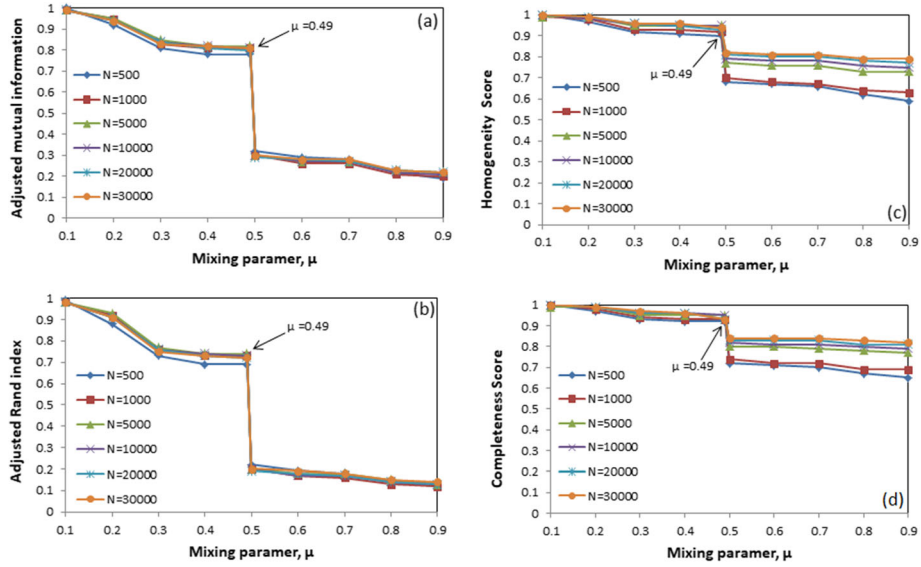

Fig. 2 Variation of (**a**) the adjusted mutual information (AMI), **b** adjusted Rand index (ARI), **c** homogeneity score (HS) and (**d**) completeness score (CS) with mixing parameter, μ for network sizes between *500* and *30,000*

the weighting of different correlation types has been applied to describe the relationship between nodes linked by edges and identify their relative importance.

The benchmarking process allowed us to assess the performance of our approach. The results indicate a very mild dependence on the size of the network, a reasonable agreement with the LFR results for low to medium values of the mixing parameter, but a clear deterioration for $\mu \geq 0.5$. We did not check the limitation of the methodology as a function of the size of the network (beyond *30,000*). This is part of our future work, which also includes the development of an algorithm that considers overlap between communities with the main objective of improving the quality of our results as the mixing parameter, μ increases from *0.5*.

As it stands the method has many positive aspects. In particular, the communities emerge naturally from the clustering method, without the knowledge of their size or number. The method may also be used to model the temporal evolution of network degrees, i.e. the re-wiring of nodes based on the condition of preferential attachment (Barabási & Albert, 1999; Felfli et al., 2018). This contributes to gaining insight into the dynamic and structural changes that occur with time within the network and the re-organization of members.

Finally, the algorithm can equally handle undirected and directed networks and a priori, should work well for high-dimensional data sets. The ranking of nodes via the assignment of scores defined by their density values, the identification of the community and group leaders is important from a cyber intelligence standpoint. It leads to the discovery of nodes (benign or otherwise) operating in concert, the unveiling of controller nodes, and facilitates the exploration of the propagation mechanism of malware and opens the possibility of dismantling or disrupting malicious network structures.

Abbreviations

ARI: Adjust Rand Index; BA: Barabasi-Albert; CI: Completeness Index; HI: Homogeneity Index; LFR: Lancichinetti, Fortunato and Radicchi; NGC: Northrop Grumman Corporation; NMI: Normalized Mutual Information

Acknowledgments
This research is funded in part by NSF Grant No. FAIN-1901150, Department of Energy and NGC under Contract Number DE-NA 0002686, Contract No. 2017-2007 respectively. Any opinions, findings, and conclusions expressed here are those of the author(s) and do not reflect the views of the sponsor(s).

Authors' contributions
ZF performed the density-based calculation using the LFR data; wrote the paper. RG contributed to the discussions and the analysis of the data; wrote algorithm for the evaluation of the various scores (ARI, NMI, HI and CI). KS contributed to the discussions and the analysis of the data. MK put together the LFR benchmark algorithm in python and generated the LFR network data. All authors gave final approval for publication.

Funding
This research is funded in part by the Department of Energy and NGC under Contract Number DE-NA 0002686, Contract No. 2017–2007 respectively. Any opinions, findings, and conclusions expressed here are those of the author(s) and do not reflect the views of the sponsor(s).

Availability of data and materials
The data and material are available upon request.

Competing interests
The authors declare that they have no competing interests.

Received: 7 January 2019 Accepted: 30 May 2019
Published online: 22 October 2019

References

Akamatsu T (1996) Cyclic flows, markov process and stochastic traffic assignment. Transportation Res B 30(5):369–386
Bahrami Bidoni Z, George R (2014) Discovering Community Structure in Dynamic Social Networks using the Correlation Density Rank. ASE BigData/SocialCom/Cybersecurity Conference, Palo Alto
Barabási A-L (2017) Network science. Cambridge University Press, Cambridge
Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286:509–512
Daudin J-J, Picard F, Robin S (2008) A mixture model for random graphs. Stat Comput 18(2):173–183
Felfli Z, George R, Shuiaee K, Kerwat M (2018) Computing Ranking and Dynamics in Social Networks. In: Proceedings of the 5th International Conference on Social Networks Analysis, Management and Security (SNAMS2018)
Fiedler M (1973) Algebraic connectivity of graphs. Czech Math J 23:298–305
Francoisse K, Kivimaki I, Mantrach A, Rossi F, Saerens M (2017) A bag-of-paths framework for network data analysis. Neural Netw 90:90–111
Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci U S A 99:7821–7826
Hinneburg A, Keim DA (1998) An efficient approach to clustering in large multimedia databases with noise. KDD 98:58–65
Hofman JM, Wiggins CH (2008) Bayesian approach to network modularity. Phys Rev Lett 100(25):258701
Kernighan BW, Lin S (1970) An efficient heuristic procedure for partitioning graphs. Bell Sys Techn J 49:291
Kivimäki I, Shimbo M, Saerens M (2014) Developments in the theory of randomized shortest paths with a comparison of graph node distances. Physica A: Stat Mech Appl 393:600–616
Krzakala F, Moore C, Mossel E, Neeman J, Sly A, Zdeborov L, Zhang P (2013) Spectral redemption in clustering sparse networks. Proc Natl Acad Sci 110(52):20 935–20 940
Lancichinetti A, Fortunato S, Radicchi F (2008) Benchmark graphs for testing community detection algorithms. Phys Rev E 78:046110
Newman MEJ (2004) Detecting community structure in networks. Eur Phys J B 38:321. https://doi.org/10.1140/epjb/e2004-00124-y
Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. Phys Rev E 69(2):026113
Orman GK, Labatut V, Cherifi H (2013) Towards realistic artificial benchmark for community detection algorithms evaluation. Int J Web Based Commun 9(3):349–370
Pothen A, Simon H, Liou K-P (1990) Partitioning sparse matrices with eigenvectors of graphs. SIAM J Matrix Anal Appl 11:430–452
Saade A, Krzakala F, Zdeborov L (2014) Spectral clustering of graphs with the Bethe Hessian. Adv Neural Info Proc Sys 27:406–414
Saerens M, Achbany Y, Fouss F, Yen L (2009) Randomized shortest-path problems: two related models. Neural Comput 21(8):2363–2404
Schaub MT, Delvenne JC, Rosvall M, Lambiotte R (2017) The many facets of community detection in complex networks. Appl Netw Sci 2:4. https://doi.org/10.1007/s41109-017-0023-6
Scott J (2000) Social network analysis: a handbook, 2nd edn. Sage, London
Wang YR, Bickel PJ et al (2017) Likelihood-based model selection for stochastic block models. Ann Stat 45(2):500–528
Yan X (2016) Bayesian model selection of stochastic block models. In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). IEEE, pp 323–328
Yen L, Mantrach A, Shimbo M, Saerens M (2008) A family of dissimilarity measures between nodes generalizing both the shortest-path and the commute-time distances. In: Proceedings of the 14th SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2008), pp 785–793

Publisher's Note