Disassortative Degree Mixing and Information Diffusion for Overlapping Community Detection in Social Networks (DMID)

Mohsen Shahriari
Advanced Community Information Systems (ACIS), RWTH Aachen University, Ahornstr. 55, 52056 Aachen, Germany
shahriari@dbis.rwth-aachen.de

Sebastian Krott
Advanced Community Information Systems (ACIS), RWTH Aachen University, Ahonrstr. 55, 52056 Aachen, Germany
krott@dbis.rwth-aachen.de

Ralf Klamma
Advanced Community Information Systems (ACIS), RWTH Aachen University, Ahonrstr. 55, 52056 Aachen, Germany
klamma@dbis.rwth-aachen.de

ABSTRACT

In this paper we propose a new two-phase algorithm for overlapping community detection (OCD) in social networks. In the first phase, called disassortative degree mixing, we identify nodes with high degrees through a random walk process on the row-normalized disassortative matrix representation of the network. In the second phase, we calculate how closely each node of the network is bound to the leaders via a cascading process called network coordination game. We implemented the algorithm and four additional ones as a Web service on a federated peer-to-peer infrastructure. Comparative test results for small and big real world networks demonstrated the correct identification of leaders, high precision and good time complexity. The Web service is available as open source software.

Categories and Subject Descriptors

G.2.2 [**Discrete Mathematics**]: Graph Theory—*Graph Algorithms*; H.2.8 [**Database Management**]: Database Applications—*Data Mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Clustering*

General Terms

Algorithms, theory and experimentation

Keywords

Overlapping community detection; expert identification; information diffusion; Web service

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2741696 .

1. INTRODUCTION

In recent years, researchers have been interested in identifying the more connected and dense parts of graphs, named communities. There is still no well-established definition of the term community but we consider both a traditional and a more modern one. In the classical understanding, communities are considered components in which internal cluster relationships are dense and the relationships among different components are sparse [8]. In a more modern definition online communities are defined as groups of people or nodes which interact with each other based on their needs and interests [22]. First research on community detection algorithms just intended to identify a set of disjoint communities. However, today's online networks, people (or nodes) tend to belong to more than one community. E.g. readers of online media are interested in more than one topic or are member of several news groups. Consequently the detection of overlapping communities has recently gained much attention [13, 4, 5, 6, 7, 11, 17, 29, 32].

There exists a broad variety of methods for identifying overlapping communities. The first category employs global properties of the graph such as e.g. the global clustering coefficient or modularity values to find covers. The modularity optimization by Girvan and Newman is a typical example for this category [18]. So-called local approaches circumvent problems related to global methods like resolution limit and high time complexity. They apply optimization techniques, identify cliques of fixed small sizes or make use of local dynamics through random walk processes and clustering coefficients [13, 24, 30, 31, 10, 9, 2, 32]. One subcategory of local methods for overlapping community detection is based on identifying leaders. At first most influential nodes are detected in the graph and each or a group of them are considered as communities. Afterwards, the other nodes' membership are calculated based on the leaders. Different kinds of metrics and processes like node weights or degree [2, 13], node distances [17] and random walks [24] can be used to determine most influential vertices in the network.

In this manuscript a new local approach for detecting overlapping communities is proposed. It is a two-phase method which utilizes two kinds of dynamics for identifying communities: disassortative degree mixing and information diffusion. Communities form around influential nodes, so identifying these nodes is very important [24]. In networks, nodes can have two states regarding their similarity with their neighbours. One state is assortative, where nodes tend to communicate with similar nodes in terms of rank, degree or other aspects. For example assortative low-degree nodes tend to communicate with other low-degree nodes. Disassortativity on the other hand is a sign of dissimilarity. Disassortative high-rank/high-degree nodes tend to communicate with low-rank/low-degree nodes and vice versa, which is also a characteristic of real world networks [19]. However, for identifying communities, we should look for disassortative hubs. Most real world networks like the internet, citation networks, communication and co-authorship networks have disassortative degree mixing properties. For example, scientists like to cite reputable papers in their field and students like to co-author with well-known and high-rank professors [25]. Numerous methods can be used for detecting disassortative hubs. Here a simple random walk process is proposed. First, we compute a disassortative matrix whose elements are computed based on subtraction of the corresponding node degrees. By row-normalizing this matrix we then obtain a disassortative transition matrix. By performing a random walk process for each node, we compute the local disassortative-ness of each node in the network. Through the disassortative-ness value we identify nodes which largely differ from their neighbours in terms of degree. Among these disassortative nodes we are interested in those which have the highest degrees in the network. Details of disassortative dynamics are explained in the method section. In the second phase of the algorithm the membership degree of each node to the communities is computed. For this purpose a network coordination game based on the dynamics of information diffusion is used. The details of the underlying cascading process are explained in the method section as well.

The proposed method and the four algorithms including SSK [24], CLiZZ [17], MONC [6] and Link Communities [30] are implemented on top of a java-based peer-2-peer infrastructure [1] and provided as open source software. Among the above mentioned baseline approaches, SSK and CLiZZ are leader-based local OCD algorithms. They are suitable for both directed and weighted networks. SSK uses a random walk process for the detection of leaders [24]. Besides CLiZZ finds leaders based on a distance matrix and membership of non-leader nodes are computed iteratively [17]. Moreover, MONC is also a local based method which performs based on community expansion. In other words, it starts with some initial communities (seeds) and it adds nodes to communities based on a fitness function [6]. Finally, Link Communities algorithm is an approach for OCD based on link partitioning. It aims to increase the link density inside communities by creating communities of edges as opposed to minimizing the communities' external edges. This is achieved by detecting communities of edges rather than of nodes by using the so called similarity index [30]. We ran the algorithms on different offline and online networks to demonstrate the usefulness of our approach. It proves to be competitive in terms of both precision and time complexity. In summary the paper makes the following contribution:

A novel two-phase method of overlapping community detection is proposed and experimented on small, large real world and synthetic networks. Not only the algorithm is competitive in comparison to others but also it can identify most influential nodes correctly.

[1] https://github.com/rwth-acis/LAS2peer

2. USE OF TERMS AND VARIABLES

Overlapping community detection is a graph-theoretic problem. We are trying to identify communities in a network (or graph) $G = (V, E)$ which is a tuple containing a set of nodes (or vertices) V and a set of edges (or links) E. The total number of nodes is denoted as $n = |V|$ and the total number of edges as $m = |E|$. A node j is called a neighbour of node i if the two are directly connected by an edge. The set of all neighbours of i is called the (open) neighbourhood of i, denoted $N(i)$. The degree $deg(i)$ of node i is given by the amount of edges which are incident to i. In the case of the original (disjoint) community detection problem the goal is to identify a partition of V into disjoint communities, i.e. each node of the graph is member of exactly one community. In OCD, one single node can be part of different communities. Hence a community structure is given by a cover $\Gamma = (C_1, C_2, ..., C_l)$ which is a tuple containing the (generally not disjoint) identified communities. The number of communities in cover Γ is denoted $|\Gamma|$. Often we want to be more specific and define to what degree a node is member of which community. Therefore we assign each node i a membership vector M_i of dimension l. The j-th value of this node determines to what extent i is part of C_j.

3. PROPOSED TWO-PHASED METHOD

In this section, the new method of overlapping community detection is explained.

3.1 Phase I: Disassortative Random Walk

In the first phase, a random walk is used to identify disassortative hubs. A hub is a node which has a high degree. A network is disassortative if high-degree nodes are mainly connected to low-degree nodes, and vice versa. We consider a node to be disassortative if its degree strongly differs from the degree of most of its neighbours. So a disassortative hub is a central node, i.e. a node with a high degree, whose neighbours are mostly low-degree nodes. Consequently, such nodes can be regarded as local leaders, since they have an extraordinarily high level of communication compared to their surrounding nodes.

We begin by defining a disassortative matrix

$$AS_{ij} = |deg(i) - deg(j)| \tag{1}$$

so that AS_{ij} corresponds to the disassortativity for nodes i and j which are directly connected. For the execution of a random walk we row-normalize this matrix, resulting in the following transition matrix

$$T_{ij} = \frac{AS_{ij}}{\sum_{k=1}^{|N|} AS_{ik}} \tag{2}$$

Consequently, we can perform a random walk for calculating a disassortativity vector DA^t at time t with

$$DA^{t+1} = DA^t \times T \tag{3}$$

The probability of choosing each path in this process is commensurate with its level of disassortativity of the path. So if one edge connects two nodes that are highly disassortative, then the probability of choosing this edge in the random walk process will be high as well. For initializing the values of the disassortativity vector we use a uniform distribution

$$DA_i^0 = \frac{1}{|N|} \tag{4}$$

After a finite number of t^* iterations this process converges, giving us the local disassortativity value $DA_i^{t^*}$ of each node i. Since we are searching for disassortative nodes with a high degree, we still have to consider additional information. Hence we continue by calculating a normalized degree vector

$$DRN_i = \frac{deg(i)}{\max_{k \in V} deg(k)} \tag{5}$$

using the infinity norm. Combining node degrees and disassortativity values, we define the leadership of node i as

$$LS_i = DA_i \times DRN_i \tag{6}$$

We name LS the leadership vector and call a node i a local leader if

$$LS_i > LS_j \quad \forall j \in N(i) \tag{7}$$

Node j is said to be a follower of node i if it is connected to i and i is a local leader. In order to obtain the most influential leaders from LL, the set of local leaders, we compare the number of each leader's followers with the average number of followers. We define the average follower degree as

$$AFD = \frac{\sum_{i \in LL} |FL(i)|}{|LL|} \tag{8}$$

where $FL(i)$ is the set of followers of local leader i. We finally consider a local leader i to be a global leader, i.e. a community leader, if its number of followers is above average such that holds

$$|FL(i)| > AFD \tag{9}$$

3.2 Phase II: Cascading Behaviour

In order to compute the assignment of remaining non-leader nodes to communities a cascading behaviour is used. The cascading behaviour which is used here is a network coordination game. To make things more clear, if all nodes in the network are communicating with a behaviour B and suddenly one leader of a community adopts a new behaviour A, we are interested in identifying to what extent this behaviour is cascaded through the network. If we run this game with different initial sets, in our case each consisting of one leader, we can also identify overlapping nodes. Since we have $|L|$ leaders in the network, we play this game $|L|$ times. In each game some nodes are affected by the leader's new behaviour A. If we select a suitable threshold for each node then its membership degree can be computed based on the time after which it adopts behaviour A. More precisely, for each leader a cascade set is created. Cascade sets for different leaders can overlap. Hence nodes belonging to several different cascade sets are overlapping nodes and are part of all the corresponding communities. Moreover, nodes which change their behaviour in the first iterations will have a higher degree membership for the leader's community than nodes which resist this new behaviour and join the set A in the final iterations.

To illustrate this, let us consider an example node i that currently has behaviour B and possesses three neighbours. If we assume that only one of its neighbours works with behaviour B, but two with behaviour A, then it would be more profitable for the node i to adopt behaviour A because this way it could collaborate more easily with a higher number of nodes.

More precisely, for a node i and a behaviour A we define the pay-off for that node for adopting the behaviour as the percentage of its neighbours working with behaviour A

$$p_A(i) = \frac{|\{j \in N(i) : j \text{ has behavior } A\}|}{|N(i)|} \tag{10}$$

A node will adopt a new behaviour if the pay-off for that behaviour is higher than a given profitability threshold α.

We now do this label propagation once for each leader L_j. Initially all nodes apart from L_j will have behaviour B, whereas the leader starts with behaviour A. Generally speaking, the sooner a node adopts the new behaviour, the more profitable this behaviour results for it. Hence we can determine the membership of a node i of the community represented by leader L_j by the number of iterations t_i it took for i to adopt the leader's behaviour

$$M_{ij} = \frac{1}{t_i^2} \tag{11}$$

To reduce the dependency of nodes that accept a new behaviour at a later time we propose the function in equation (11) which was also used in the experiments. The game ends after an iteration in which no additional node adopts the new behaviour. For determining a meaningful profitability threshold a binary search is performed over all possible threshold values between the bounds 0 and 1. In each iteration of the binary search the game is played once for each leader using the average value of the two current bounds as the profitability threshold. If all nodes are assigned to at least one community, the upper bound is decreased, otherwise the the lower bound will increase. The iteration count for the binary search phase must be predefined, for the experiments we used 10 iterations. The final outcome is a membership matrix M where M_{ij} defines the membership degree of node i of community j.

3.3 DMID Time Complexity and its Extension for Directed/Weighted Graphs

This algorithm can be easily tuned for weighted and directed networks. In the first phase, the nodes' weighted in-degrees will be used instead of the regular degrees to construct the disassortativity matrix and the leadership vector. In the second phase, the definition of the pay-off value changes to

$$p_A(i) = \frac{|\{j \in S(i) : j \text{ has behavior } A\}|}{|S(i)|} \tag{12}$$

where $S(i)$ is the set of successors of node i.

For analysing time complexity let us define $c(l)$ as the size of the community generated by the l-th leader. If we assume that each node is member of only a constant amount of communities and further that the sizes of the communities are equal, it can be approximated that the worst case time complexity is about $O(m\frac{n}{|\Gamma|})$.

4. EXPERIMENTAL SET UP AND DATA SETS

In this section, firtly we introduce the datasets which are used in the experiments and then we investigate and compare the results of DMID and other algorithms on synthetic and real world networks.

Table 1: Type, number of nodes and edges of networks used in the experiments. Number of edges is shown after making the graph undirected meaning that a backward edge was added for any directed edge. Information regarding the small datasets Zachary, Sawmill, Sawmill Strike and Dolphins is not listed.

Graph	dblp [21]	email [23]	facebook [12]	hamsterster [1]	internet [16]	jazz [20]	Power grid [28]
Nodes	1959	1133	4039	2000	6474	198	4941
Edges	16354	10902	176468	32196	25144	5484	13188
Type	Citation	Social	Social	Social	Tech	Social	Tech

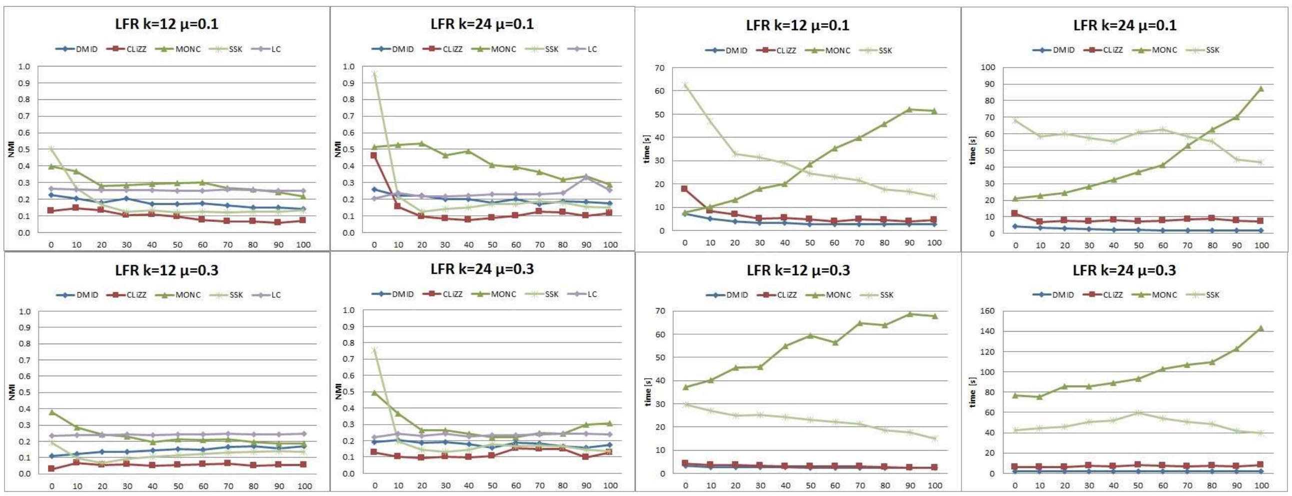

Figure 2: This figure comprises both NMI measure and CPU time calculated for different algorithms on the LFR benchmark networks with different parameter settings (average degree 12, 24 and mixing parameter 0.1, 0.3 and percentage of overlapping nodes ranging from 0 to 100 percent. The y axis for NMI and CPU time respectively indicates the NMI value and CPU time and x axis shows the percentage of overlapping nodes. For each overlap value we generated 100 networks and averaged over the results. Because of the high time complexity of the Link Communities algorithm (LC), unlike the others it was only run on one network for each parameter setting. Moreover, running time of LC was too high to be plotted.

Table 2: Results of extended modularity and running time on different real-world networks for various algorithms. The upper part of the table indicates the modularity values and the lower part shows the running times. NaN indicates that the algorithm could not finish in appropriate running time on the corresponding network.

Graph	dblp	dolphins	email	facebook	hamsterster	internet	jazz	powergrid	sawmill	sawmill strike	zachary
DMID	0.42	0.514	0.342	0.825	0.288	**0.522**	*0.346*	0.643	**0.685**	**0.752**	**0.691**
CLiZZ	*0.428*	*0.522*	0.248	*0.882*	*0.368*	0.225	0.000	*0.681*	0.455	0.00	*0.593*
MONC	0.21	0.3218	0.0886	0.00	0.173	NaN	0.201	0.398	0.364	0.496	0.332
SSK	**0.485**	**0.551**	*0.428*	**0.941**	0.366	*0.406*	**0.413**	**0.701**	*0.588*	*0.702*	0.592
LC	0.349	0.359	**0.441**	NaN	**0.511**	0.080	0.338	0.090	0.340	0.368	0.237
DMID	*110.77*	*0.38*	**5.50**	*3101.55*	734.19	*2522.62*	0.203	*3039.97*	*0.04*	0.028	**0.01**
CLiZZ	**84.91**	0.41	*7.59*	**528.90**	**29.54**	**2483.98**	**0.14**	400037	0.108	*0.01*	**0.01**
MONC	159	0.44	109	3808	*87.4*	NaN	0.29	**62.4**	**0.02**	**0.009**	0.07
SSK	3911	**0.36**	129.9	3528.4	7406	6545	0.65	204797	0.23	0.02	0.02
LC	12500	0.54	3074	NaN	133531	48736	304	5082	0.06	*0.01*	0.04

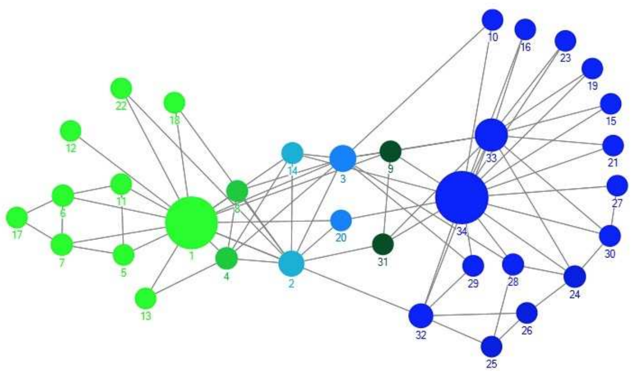

Figure 1: Results of DMID on Zachary.

4.1 Data Sets

Some information about the number of nodes, edges and types of networks are shown in table 1.

4.2 Results on Zachary Karate Club

Zachary Karate Club [27] network is popular for testing community detection algorithms. This network consists of 34 nodes which are members of a karate club and 78 edges which represent friendship among these nodes. As can be observed in figure 1, for Zachary DMID detected two communities, node 1 and node 34 as the most influential nodes. In real setting, the author identified two communities, node 1 as an instructor and node 34 as the president of the club. As the author noted, this network was split into two communities as a result of a conflict between instructor and the president. Our algorithm was able to identify exact leaders. Also it identified overlapping nodes in the network (9, 31, 14, 3, 2 and 20). However, their memberships of the leaders are different. Nodes 3 and 20 have equal membership degrees for both leaders.

4.3 Results on Synthetic Networks

For a better analysis of the precision and running time, we also used the LFR networks proposed by Lancichinetti [14]. We set the mixing parameter to $\mu \in \{0.1, 0.3\}$, the average degree to $k \in \{12, 24\}$ and increased the percentage of overlapping nodes stepwise from 0% to 100%. For each parameter combination, we generated 100 networks and averaged over the results. Figure 2 indicates extended Normalized Mutual Information (NMI) measure and running time for these networks. The extended NMI is a variant of the original NMI that is a metric for determining the quality of partitions [3], i.e. for disjoint community detection. Later it was adapted by Lancichinetti for measuring covers as well [15]. The NMI is knowledge-driven, i.e. based on the comparison with ground truth data sets.

As it can be observed in figure 2, DMID has higher NMI values in comparison to CLiZZ and SSK. As for LFR $k = 12$, $\mu = 01$ and LFR $k = 12$, $\mu = 03$, it is competitive with MONC and LC except in some cases where the NMI is a little bit lower. On the other hand CPU times indicate a better time complexity of DMID. LC is not illustrated because the high running times did not allow to plot it. In all parameter settings for the LFR networks, CPU time of DMID is better than that of both MONC and SSK. Only in LFR $k = 24$ and $\mu = 01$, the MONC algorithm outperforms others.

4.4 Results on Real-World Networks

We ran SSK, CLiZZ, Link Community (LC), DMID and MONC on the networks of table 1. Table 2 indicates the extended modularity measure and running time for these networks [2]. Extended modularity is a statistical measure for the evaluation of covers without any given ground truth datasets [26]. Concerning extended modularity, DMID has the highest values on the four datasets Zachary (0.691), Sawmill Strike (0.752), Sawmill (0.685) and Internet (0.524) and also has the second rank among the algorithms for Jazz (0.346). One can observe that DMID is very competitive over different datasets regarding both the quality of the detected community structure and the time complexity. Although SSK sometimes is better regarding modularity, DMID beats it in terms of time complexity. Moreover, it seems that CLiZZ has good time complexity in comparison to other algorithms. However, DMID defeats CLiZZ with respect to modularity. Finally, it is worth mentioning that DMID has further advantageous since it can be used e.g. to detect global and local leaders as well as node hierarchies.

5. CONCLUSIONS AND FUTURE WORKS

In this paper a two-phase overlapping community detection algorithm is proposed, which uses two social processes named disassortative degree mixing and information diffusion. The proposed algorithm has several advantages. Firstly, it detects leaders as disassortative hubs and based on these dynamics it seems that it can better identify communities in networks with disassortative degree mixing property. In future works, we would like to investigate the effects of network structure by using graphs with different assortativity values. Also, in this implementation only one leader started the cascade and all the nodes used the same threshold value to determine when to change their behaviour. In further steps we would like to change this algorithm to allow multiple leaders for each community in the first phase and different threshold values for the other non-leader nodes in the second phase. Secondly, this algorithm can uncover the hierarchical structure in the network. Thirdly, via this algorithm, overlapping nodes can be identified and due to the soft partitioning one can identify the membership degrees of each node for all communities. Moreover, in terms of time complexity, since the algorithm only considers local information, it can be implemented in a decentralized environment. Hence, another aspect of future work is implementing the algorithm with the Map-Reduce framework and running it on larger networks.

6. ACKNOWLEDGMENTS

The work has received funding from the European Commission's FP7 IP Learning Layers under grant agreement no 318209.

[2]On the internet network MONC and on the facebook network LC did not finish because of high running times

7. REFERENCES

[1] Hamsterster full network dataset - konect, August, 2014.

[2] D. Chen, Y. Fu, and M. Shang. An efficient algorithm for overlapping community detection in complex networks. In *Intelligent Systems, 2009. GCIS '09. WRI Global Congress on*, volume 1, pages 244–247, May 2009.

[3] L. Danon, J. Duch, A. Diaz-Guilera, and A. Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, P09008, 2005.

[4] T. S. Evans. Clique graphs and overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment*, (12), 2010.

[5] T. S. Evans and R. Lambiotte. Line graphs of weighted networks for overlapping communities. *European Physical Journal B*, 77(2):265–272, 2010.

[6] F. Havemann, M. Heinz, A. Struck, and J. Glaser. Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels. *Journal of Statistical Mechanics: Theory and Experiment*, 2011.

[7] M. Fan, K.-C. Wong, T. Ryu, T. Ravasi, and X. Gao. Secom: A novel hash seed and community detection based-approach for genome-scale protein domain identification. *PLOS ONE*, 7(6), 2012.

[8] 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.

[9] S. Gregory. A fast algorithm to find overlapping communities in networks. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases - Part I*, ECML PKDD '08, pages 408–423, Berlin, Heidelberg, 2008. Springer-Verlag.

[10] S. Gregory. Finding overlapping communities in networks by label propagation. *New Journal of Physics*, 12(10):1–21, 2010.

[11] J. Huang, H. Sun, J. Han, and B. Feng. Density-based shrinkage for revealing hierarchical and overlapping community structure in networks. *Physica A-Statistical Mechanics and its Applications*, 390(11):2160–2171, 2011.

[12] J. McAuley and J. Leskovec. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems 25*, pages 548–556.

[13] D. Jin, B. Yang, C. Baquero, D. Liu, D. He, and J. Liu. A markov random walk under constraint for discovering overlapping communities in complex networks. *Journal of Statistical Mechanics-Theory and Experiment*, 1303.5675, 2011.

[14] A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80(1):016118, July 2009.

[15] A. Lancichinetti, S. Fortunato, and J. Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.

[16] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. KDD '05, pages 177–187, New York, NY, USA, 2005. ACM.

[17] H. Li, J. Zhang, Z. Liu, L. Chen, and X. Zhang. Identifying overlapping communities in social networks using multi-scale local information expansion. *Eur. Phys. J. B*, 85(6):190, 2012.

[18] M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(026113), 2004.

[19] M. E. J. Newman. Mixing patterns in networks. *Phys. Rev. E*, (67), 2003.

[20] P. Gleiser and L. Garrido. Advances in complex systems. 6(565), 2003.

[21] M. Pham, D. Kovachev, Y. Cao, and R. Klamma. Enhancing academic event participation with context-aware and social recommendations. In *ASONAM 2012*, pages 457–464.

[22] J. Preece. *Online Communities: Designing Usability and Supporting Sociability.* John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 2000.

[23] R. Guimerà, L. Danon, A. Díaz-Guilera, and F. Giralt, and A. Arenas. Self-similar community structure in a network of human interactions. *American Physical Society*, 68(6):065103, 2003.

[24] A. Stanoev, D. Smilkov, and L. Kocarev. Identifying communities by influence dynamics in social networks. *Phys. Rev. E*, 84:046102, Oct 2011.

[25] L. Šubelj and M. Bajec. Model of complex networks based on citation dynamics. In *Proceedings of the WWW Workshop on Large Scale Network Analysis 2013 (LSNA '13)*, pages 527–530.

[26] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri. Extending the definition of modularity to directed graphs with overlapping communities. *J. Stat. Mech. (2009) P03024.*

[27] W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.

[28] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.

[29] Z. Wu, Y. Lin, H. Wan, S. Tian, and K. Hu. Efficient overlapping community detection in huge real-world networks. *Physica A-Statistical Mechanics and its Applications*, 391(7):2475–2490, 2012.

[30] J. Xie, S. Kelley, and B. K. Szymanski. Overlapping community detection in networks: the state of the art and comparative study. *CoRR*, abs/1110.5813, 2013.

[31] J. Xie, B. K. Szymanski, and X. Liu. Slpa: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In *ICDMW 2012*, pages 344–349.

[32] Y. Zhang, J. Wang, Y. Wang, and L. Zhou. Parallel community detection on large networks with propinquity dynamics. *KDD-09*, pages 997–1006, 2009.