Fast Optimization of Hamiltonian for Constrained Community Detection

Keisuke Nakata and Tsuyoshi Murata

Department of Computer Science, Graduate School of Information Science and Engineering
Tokyo Institute of Technology
W8-59 2-12-1 Ookayama Meguro Tokyo 152-8552 Japan

Abstract. Various methods for analyzing networks have been proposed. Among them, methods for community detection based on network structures are important for making networks simple and easy to understand. As an attempt to incorporate background knowledge of given networks, a method known as constrained community detection has been proposed recently. Constrained community detection shows robust performance on noisy data since it uses background knowledge. In particular, methods for community detection based on constrained Hamiltonian have advantages of flexibility in output results. In this paper, we propose a method for accelerating the speed of constrained community detection based on Hamiltonian. Our optimization method is a variant of Blondel's Louvain method which is well-known for its computational efficiency. Our experiments showed that our proposed method is superior in terms of computational time, and its accuracy is almost equal to the existing method based on simulated annealing under the same conditions. Our proposed method enables us to perform constrained community detection in larger networks compared with existing methods. Moreover, we compared the strategies of adding constraints incrementally in the process of constrained community detection.

1 Introduction

There are emerging needs for understanding the structures of huge data due to the growing advancement of information technologies. Many of them can be represented as networks, such as friendship networks of social media or hyperlink networks of Web pages. Several attempts have been made for community detection [POM09][For10]; extracting dense subnetworks from given networks. Community detection is important for analyzing and visualizing given networks from mesoscopic viewpoints.

One of the most popular metrics for community detection is modularity [NG04]. It is often used for evaluating the qualities of detected communities compared with the null model. Many community detection methods optimize modularity in order to search for partitions of given networks [CNM04][For10][PKVS12]. As the method for optimizing modularity of large-scale networks, Louvain method [BGLL08] is often employed.

One of the promising directions of community detection is to incorporate constraints on communities to be detected, which is called constrained community detection. In many cases, humans already have some background knowledge on the structure of given networks. Such knowledge should be incorporated in the process of community detection in order to find better communities.

G. Mangioni et al. (eds.), *Complex Networks VI,*
Studies in Computational Intelligence 597, DOI: 10.1007/978-3-319-16112-9_8

Among the approaches of constrained community detection, Reichardt and Bornholdt [RB06] introduced Hamiltonian as a generalization of modularity. Eaton et al. proposed a method for optimizing constrained Hamiltonian [EM12]. Although the method is theoretically good, it is slow since it employs simulated annealing [KJV83] for optimizing constrained Hamiltonian.

This paper extends Louvain method, and proposes a method for fast optimization of constrained Hamiltonian. It is often said that there is a tradeoff between accuracy and speed, but our optimization method satisfies both. It is effective not only for processing large-scale networks but also for performing interactive community detection since users often put some additional constraints after they watched the results of obtained communities. There are many strategies for giving constraints incrementally in the process of community detection, hence we performed experiments comparing some of them.

2 Related Works

This section introduces some basic metrics and notations that are necessary for explaining our proposed method.

2.1 Modularity

Modularity introduced by Newman and Girvan [NG04] is one of the most popular metrics for evaluating the quality of communities extracted from a given network. The metric is computed from the difference between the number of actual edges within communities in a network and the expected value of its null model. Null model of a network is generated by rewiring edges of the network while degrees of all vertices are kept the same as those of the original network. Modularity shows the amount of deviation of the number of edges within communities from random partitions. Therefore, partitions of high modularity are regarded as good from the viewpoint of community detection. The value of modularity Q is defined as follows:

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - P_{ij}\right) \delta\left(C_i, C_j\right), \tag{1}$$

where i and j are indices of nodes, A is an adjacency matrix of the network, $P_{ij} = (k_i k_j)/2m$ is a null model of the network, k_i is the degree of node i, $m = \sum_i k_i/2$ is the number of edges in the network, C_i is the index of the community which node i belongs to, and δ is the Kronecker's delta. In order to detect communities, partitions of high Q values are searched, and it is often called modularity optimization.

3 Generalization of Modularity

Hamiltonian $\mathscr{H}$ [RB06], which is a generalization of modularity (expression (1)), is expressed as follows:

$$\begin{aligned}\mathscr{H} = &-\sum_{i,j} a_{ij}A_{ij}\delta(C_i,C_j) \\ &+\sum_{i,j} b_{ij}(1-A_{ij})\delta(C_i,C_j) \\ &+\sum_{i,j} c_{ij}A_{ij}(1-\delta(C_i,C_j)) \\ &-\sum_{i,j} d_{ij}(1-A_{ij})(1-\delta(C_i,C_j)).\end{aligned} \tag{2}$$

We have to keep in mind that in contrast to modularity, smaller Hamiltonian value means better network partition. In expression (2), Hamiltonian (a) rewards intra-community edges (the first term), (b) penalizes the lack of intra-community edges (the second term), (c) penalizes inter-community edges (the third term), and (d) rewards the lack of inter-community edges (the fourth term), and each is weighted by parameters a, b, c and d, respectively.

If the parameters are set appropriately ($a_{ij} = c_{ij} = 1-\gamma P_{ij}$, $b_{ij} = d_{ij} = \gamma P_{ij}$), expression (2) can be transformed as follows:

$$\mathscr{H} = -2\sum_{i,j}(A_{ij} - \gamma P_{ij})\delta(C_i,C_j) + 2m(1-\gamma). \tag{3}$$

The second term on the right side, $2m(1-\gamma)$, can be ignored because it is independent of the result of community detection. Then the expression is equal to the definition of modularity (expression (1)) times constant value. This means that Hamiltonian is a generalization of modularity.

3.1 Constrained community detection

As a method for performing constrained community detection, Eaton et al. [EM12] proposed an optimization for constrained Hamiltonian, in which a constrained term is added to the above-mentioned Hamiltonian (expression (3)). Constrained term U is composed of (a) u_{ij} which means that a pair of nodes should be in the same community, and (b) $\overline{u}_{ij}$ which means that a pair of nodes should be in different communities:

$$U = \sum_{i,j}\left(u_{ij}\left(1-\delta\left(C_i,C_j\right)\right) + \overline{u}_{ij}\delta\left(C_i,C_j\right)\right). \tag{4}$$

Settings for the values of u_{ij} and $\overline{u}_{ij}$ are discussed in section 5. Constrained Hamiltonian $\mathscr{H}'$ is expressed as follows:

$$\begin{aligned}\mathscr{H}' &= \mathscr{H} + \mu U \\ &= -2\sum_{i,j}\left(\left(A_{ij} - \gamma P_{ij} + \mu\Delta U_{ij}\right)\delta\left(C_i,C_j\right)\right) + K,\end{aligned} \tag{5}$$

where μ is a parameter for balancing Hamiltonian $\mathscr{H}$ and constrained term U, $\Delta U_{ij} = (u_{ij} - \overline{u}_{ij})/2$, $K = 2m(1-\gamma) + \mu\sum_{i,j} u_{ij}$, respectively. K is a constant independent from extracted communities.

Eaton et al. employed simulated annealing [KJV83] in order to optimize expression (5). They claimed that noise-tolerant and accurate constrained community detection is achieved [EM12].

3.2 Louvain Method

Louvain method [BGLL08] is a method known for its fast optimization of modularity. Although Louvain method is a straightforward greedy method, it experimentally showed high accuracy. Louvain method consists of the following two phases:

1. Each node is moved to one of its adjacent communities, and the gain of modularity value after the move is computed. The move that will increase modularity the most will be employed and the node is assigned to the new community, but only if the gain is positive. This process is repeated for every node until no more increase of modularity can be obtained.
2. Each community obtained in step 1 is aggregated to a node, and a new network of aggregated nodes is generated.

The above two phases are repeated iteratively until convergence. In phase 1, only the difference of modularity before and after the move (ΔQ) is computed in order to speedup the computation. When node x is moved from community Y to community Z, the difference of modularity value ΔQ is as follows:

$$\Delta Q = \frac{1}{m}\left(\sum_{i\in Z}(A_{ix}-P_{ix}) - \sum_{i\in Y}(A_{ix}-P_{ix})\right), \tag{6}$$

where k_i in $P_{ij} = (k_i k_j)/2m$ is the sum of weights of all edges that are connected to node i.

In phase 2, each community obtained in phase 1 is regarded as a node and a new network of the nodes is generated. The weight of an edge that connect two nodes in the new network is the sum of the weights of all edges that connect nodes between corresponding two communities before aggregation. The weight of self-loop edge in a new network is equal to the double of the sum of all edges within the community.

4 Fast Optimization of Hamiltonian for Constrained Community Detection

Eaton et al. claimed that optimization of constrained Hamiltonian is good for constrained community detection, although they used slow simulated annealing for the optimization. We extended Louvain method (which was originally for optimizing modularity) for the optimization of constrained Hamiltonian in order to speedup constrained community detection.

Our method for optimization is similar to Louvain method, except $\Delta\mathscr{H}'$ is computed in phase 1 in section 3.2 instead of ΔQ. The difference of constrained Hamiltonian $\mathscr{H}'$ before and after node x is moved from community Y to community Z ($\Delta\mathscr{H}'$) is

represented as follows; where C^y is the network partition before the move (when node x belongs to community Y), and C^z is the network partition after the move (when node x belongs to community Z):

$$\Delta\mathscr{H}' = \left(-2\sum_{i,j}\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ij})\,\delta(C_i^z,C_j^z)\right)+K\right) - \left(-2\sum_{i,j}\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ij})\,\delta(C_i^y,C_j^y)\right)+K\right). \tag{7}$$

Since the communities of other nodes except x is the same (if $i \neq x$ and $j \neq x$ then $\delta(C_i^z,C_j^z) = \delta(C_i^y,C_j^y)$), the following equation holds:

$$\begin{aligned}\frac{\Delta\mathscr{H}'}{2} = &-\sum_i\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix})\,\delta(C_i^z,C_x^z)\right)\\ &-\sum_j\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{xj})\,\delta(C_x^z,C_j^z)\right)\\ &+\sum_i\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix})\,\delta(C_i^y,C_x^y)\right)\\ &+\sum_j\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{xj})\,\delta(C_x^y,C_j^y)\right).\end{aligned} \tag{8}$$

Since A, P and ΔU are symmetric matrices[1], the following equation holds:

$$\begin{aligned}\frac{\Delta\mathscr{H}'}{2} = &-2\sum_i\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix})\,\delta(C_i^z,C_x^z)\right)\\ &+2\sum_i\left((A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix})\,\delta(C_i^y,C_x^y)\right).\end{aligned} \tag{9}$$

If nodes i and x are in different communities, $\delta(C_i,C_x) = 0$. Otherwise, if they are in the same community, $\delta(C_i,C_x) = 1$. Therefore the following equation holds:

$$\Delta\mathscr{H}' = -4\left(\sum_{i\in Z}(A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix}) - \sum_{i\in Y}(A_{ij}-\gamma P_{ij}+\mu\Delta U_{ix})\right). \tag{10}$$

Expression (10) is computed in our proposed method in order to perform constrained community detection. If the parameter μ is set to $\mu = 0$, the term ΔU is cancelled out in expressions (5) and (10), and our method is the same as the normal community detection without considering constraints. If the parameter μ is set to a large value, ΔU dominates the behavior of $\mathscr{H}'$, and the communities that only focus on constraints will be extracted.

Since computational cost of expression (10) is almost the same as that of expression (6), the efficiency of our proposed method for optimizing constrained Hamiltonian is expected to achieve the same level as Louvain method.

[1] In the case of an undirected network, A and P are always symmetric. Blondel's original Louvain method is basically for undirected networks.

Table 1. Networks used in our experiments

Network	#nodes	#edges	#communities
Karate [Zac77]	34	78	2
Polbooks [Kre]	105	441	3
Polblogs [AG05]	1,222	16,714	2

5 Experiments

Table 1 shows the networks that were used for our experiments. Correct communities are known in advance as the ground-truth labels for each of them. Parameters are set as follows: $\mu = 2$, $\gamma = 1$, and $P_{ij} = k_i k_j / 2m$.

We focus on the constraints of assigning a positive integer l_i (as community label) to node i. A label of an unconstrained node is assigned as $l_i = -1$. Values of u_{ij} and $\overline{u}_{ij}$ are set as follows:

$$u_{ij} = \begin{cases} 1 & (\text{when } l_i = l_j \neq -1), \\ 0 & (\text{otherwise}), \end{cases} \tag{11}$$

$$\overline{u}_{ij} = \begin{cases} 1 & (\text{when } l_i \neq l_j \wedge l_i \neq -1 \wedge l_j \neq -1), \\ 0 & (\text{otherwise}). \end{cases} \tag{12}$$

As a metric for measuring the similarity between extracted communities C and correct communities C', normalized mutual information (NMI) [SG03] is used:

$$\mathrm{NMI}(C, C') = \frac{\sum_c \sum_{c'} n_{cc'} \log \frac{n_{cc'} \cdot n}{n_c \cdot n_{c'}}}{\sqrt{\left(\sum_c n_c \log \frac{n_c}{n} \right) \left(\sum_{c'} n_{c'} \log \frac{n_{c'}}{n} \right)}}, \tag{13}$$

where c and c' are indices of communities C and C', n is the number of nodes, $n_{cc'}$ is the number of nodes that belong to both c and c', and n_c and $n_{c'}$ are the number of nodes that belong to c and c', respectively. The more C and C' are similar, the larger their NMI is. C is set to the extracted communities and C' is set to the correct communities in order to measure the accuracy of community detection.

5.1 Comparison of Our Proposed Method, Simulated Annealing Method and Louvain Method

We can consider two cases for constrained community detection: (1) all constraints are given in advance, and (2) constraints are given incrementally. This subsection discusses the former case for comparing our proposed method, simulated annealing method, and Louvain method.

Figure 1 shows comparisons of accuracy using Karate network, Polbooks network and Polblogs network. X axis is the ratio of randomly added/deleted edges (as noise)

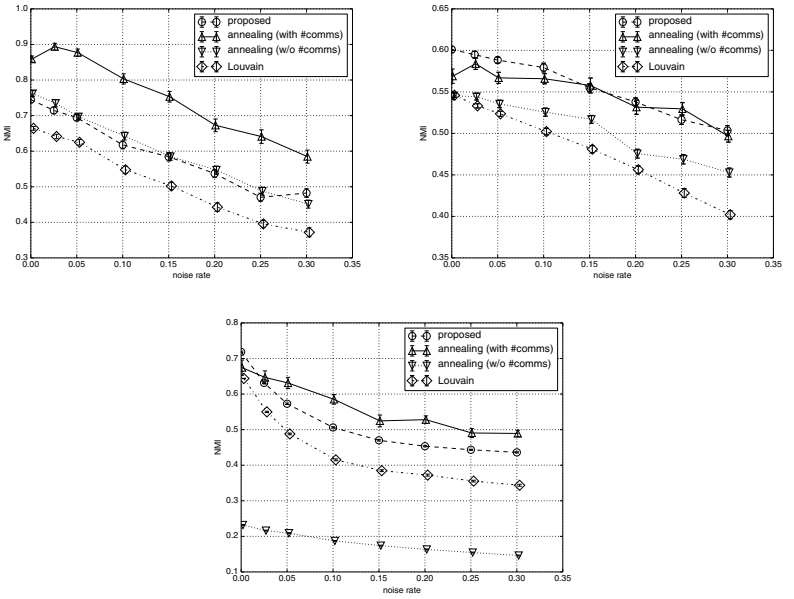

Fig. 1. Accuracies of our proposed method, simulated annealing method and Louvain method using Karate network (top left), Polbooks network (top right), and Polblogs network (bottom).

with keeping the degree distributions, and Y axis is NMI. In our proposed method and simulated annealing method, 20% of nodes are randomly selected and their ground-truth labels are given as constraints. Error bars show standard errors.

As Figure 1 shows, our proposed method is almost as accurate as simulated annealing method, if the number of communities is not given. In [EM12], the number the of ground-truth communities is given to simulated annealing method (triangular solid line in Figure 1). We also performed experiments with simulated annealing method without giving the number of communities (reversed-triangular dotted line in Figure 1), in order to compare it with our proposed method in the same condition. It was already pointed out that Louvain method is effective for optimizing modularity compared with other optimization methods [BGLL08], which is consistent with this result.

Figure 2 shows the comparisons of computational times of three methods. X axis is the same as Figure 1, and Y axis is the computational time (seconds). This showed that our proposed method is significantly faster than simulated annealing.

These results showed that our proposed method is almost as accurate as simulated annealing, and is much faster. This enables us to process large-scale networks.

5.2 Experiments on Large-Scale Networks

Table 2 shows the large-scale networks which we experimented with. Because there was no ground-truth label for them, it is impossible to give constraints from ground-truth labels or to measure the accuracy with NMI. However we tried to detect communities

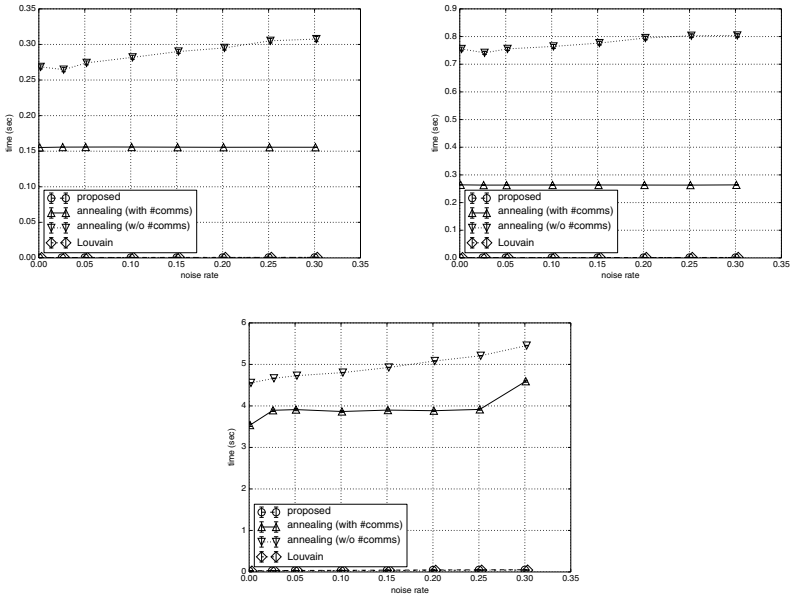

Fig. 2. Computational times of our proposed method, simulated annealing method and Louvain method using Karate network (top left), Polbooks network (top right) and Polblogs network (bottom)

from them with our proposed method and simulated annealing method without giving constraints in order to check the computational costs of them.

The results are shown in Table 3. It implies that our proposed method is very fast on large-scale networks.

5.3 Incremental Constrained Community Detection

This section discusses how to give constraints incrementally during the optimization of constrained Hamiltonian. Suppose there are no constraints at the initial stage, and constraints are given one by one and then constrained community detection is performed based on the constraints given so far. Since giving too many constraints manually is

Table 2. Large-scale networks used in our experiments

Network	#nodes	#edges	#communities
Power [WS98]	4,941	6,594	unknown
Dblp [YL12]	317,080	1,049,866	unknown[2]

[2] Dblp network has the overlapping and nested ground-truth communities, but that is not suitable because we assume that each node must belong to exactly one community.

Table 3. Computational times (second) for large-scale networks of our proposed method and simulated annealing method

	Annealing	Proposed
Power	3.016	0.056
Dblp	143.111	12.720

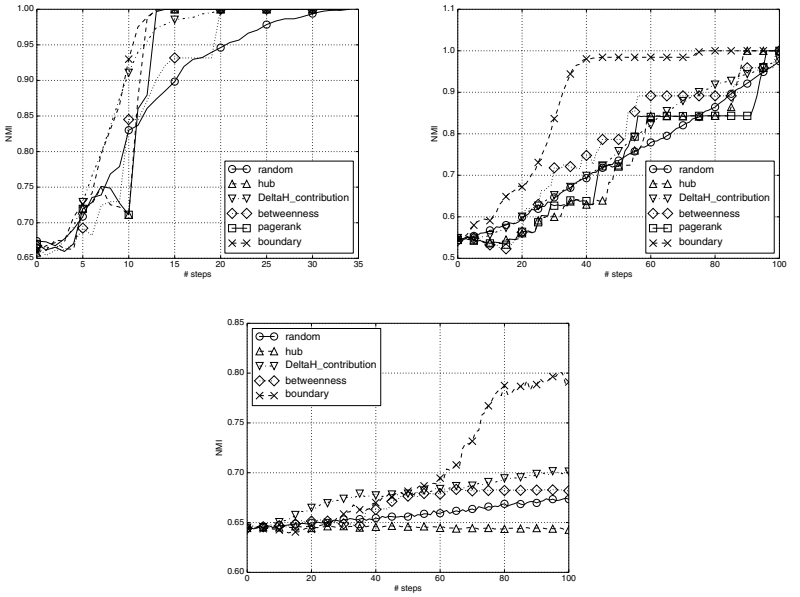

Fig. 3. Incremental addition of constraints and corresponding NMI using Karate network (top left), Polbooks network (top right) and Polblogs network (bottom)

unrealistic, we have to think about the strategies for selecting nodes that should be constrained.

Figure 3 shows the results of incremental addition of constraints and corresponding NMI values after constrained community detection was performed with our proposed method. X axis is the number of constraints, and Y axis is NMI. Lines in the Figure correspond to the following strategies for giving constraints:

random: Nodes are selected randomly.
hub: Nodes are selected in descending order of their degrees.
DeltaH_contribution: Nodes are selected in descending order of expression (10).
betweenness: Nodes are selected in descending order of betweenness.
pagerank: Nodes are selected in descending order of PageRank[PBMW99].
boundary: Nodes adjacent to different communities are selected.

The top left of Figure 3 shows that the performances of DeltaH_contribution and boundary are good when the number of constraints are less than ten. Among them,

boundary strategy is the best since it quickly reaches the highest NMI value. The top right and bottom of Figure 3 also shows that boundary is the best strategy. The results show that the order of adding constraints matters for an accurate constrained community detection. Based on the above results, we can conclude that the boundary strategy is the best in our list of surveyed strategies. This strategy gives constraints to the nodes that are located at the boundaries of different communities. It makes sense because giving constraints to such marginal nodes is expected to enhance the accuracies of community detection.

As for the strategy for adding constraints to nodes, the uncertain sampling [LG94] is often employed. The strategy is to select nodes whose degree of "wrongness" are the biggest. It has been pointed out that humans' strategies are often superior to uncertain sampling. This means that the performance of humans' interactive constrained community detection is expected to be better than the results shown in Figure 3.

6 Conclusion

This paper extends Louvain method for optimizing constrained Hamiltonian. Our proposed method is much faster than the existing simulated annealing method, without any compromise in accuracy. In addition, we performed some experiments on incremental constrained community detection and compare the strategies for giving constraints.

The followings are left for our future work.

Firstly, appropriate values of parameters such as γ, μ, u, $\overline{u}$ should be discussed further. We have used the same values that are used in Eaton's paper [EM12]. But theoretical and experimental optimization for these parameters have yet to be solved. μ controls the strength of overall constrained term, and u, $\overline{u}$ controls the strength of each constraint. Hence u, $\overline{u}$ can be set to the degree of user's confidence on each constraint. Another direction of this research is to set the weight of each constraint automatically.

Secondly, good strategies for giving constraints should be discussed further. It might be good to observe and imitate humans' heuristic strategies for accurate constrained community detection. The final goal of our research is to develop an environment of network analysis that would allow an interactive feedback from users, and this would give more insights into the performance of interactive community detection.

References

AG05. Adamic, L.A., Glance, N.: The political blogosphere and the 2004 u.s. election: Divided they blog. In: Proceedings of the 3rd International Workshop on Link Discovery, LinkKDD 2005, pp. 36–43. ACM, New York (2005)

BGLL08. Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008(10), P10008 (2008)

CNM04. Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Phys. Rev. E 70, 066111 (2004)

EM12. Eaton, E., Mansbach, R.: A spin-glass model for semi-supervised community detection. In: Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI 2012), July 22-26, pp. 900–906. AAAI Press (2012)

For10. Fortunato, S.: Community detection in graphs. Physics Reports 486(3-5), 75–174 (2010)

KJV83. Kirkpatrick, S., Gelatt Jr., C.D., Vecchi, M.P.: Optimization by simulated annealing. Science 220, 671–680 (1983)

Kre. Krebs, V.: Books about us politics. Nodes represent books about US politics sold by the online bookseller Amazon.com. Edges represent frequent co-purchasing of books by the same buyers, as indicated by the "customers who bought this book also bought these other books" feature on Amazon

LG94. Lewis, D.D., Gale, W.A.: A sequential algorithm for training text classifiers. In: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1994, pp. 3–12. Springer-Verlag New York, Inc., New York (1994)

NG04. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E 69, 26113 (2004)

PBMW99. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab (November 1999) Previous number = SIDL-WP-1999-0120

PKVS12. Papadopoulos, S., Kompatsiaris, Y., Vakali, A., Spyridonos, P.: Community detection in social media. Data Mining and Knowledge Discovery 24(3), 515–554 (2012)

POM09. Porter, M.A., Onnela, J.-P., Mucha, P.J.: Communities in networks. Notices of the AMS 56(9) (2009)

RB06. Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. Phys. Rev. E 74, 016110 (2006)

SG03. Strehl, A., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research 3, 583–617 (2003)

WS98. Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks. Nature 393, 440–442 (1998)

YL12. Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. CoRR, abs/1205.6233 (2012)

Zac77. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research 33, 452–473 (1977)