Identification of generalized communities with semantics in networks with content

Di Jin[1], Xiaobao Wang[1], Dongxiao He[1], Wenhuan Lu[2*], Francoise Fogelman-Soulié[2], Jianwu Dang[3]
[1]: School of Computer Science and Technology, Tianjin University, Tianjin, China
[2]: School of Computer Software, Tianjin University, Tianjin, China
[3]: School of Information Science, Japan Advanced Institute of Science and Technology, Japan
Email: {jindi, wxbxmt, hedongxiao, wenhuan, soulie}@tju.edu.cn, jdang@jaist.ac.jp

Abstract—**Discovery of communities in networks is a fundamental data analysis task. Recently, researchers have tried to improve its performance by exploiting node contents, and further interpret the communities using the derived semantics. However, the existing methods typically assume that the communities are assortative (i.e. members of each group are mostly connected to other members of the same group), and are unable to find the generalized community structure, e.g. structures with either assortative or disassortative communities (i.e. vertices of the same group have most of their connections outside their group), or a combination. In addition, these methods often assume that the network topology and node contents share the same group memberships, and thus cannot perform well when the contents mismatch with network structure. Also, they are limited to using only one topic to interpret each community. To address these two issues, we propose a new generative probabilistic model which is learned by using a nested expectation-maximization algorithm. It describes the generalized communities (based on network) and the content clusters (based on contents) separately, and further explores and models their correlation to improve as much as possible each of the communities and clusters based on the other. By depicting and utilizing this correlation, our model is not only robust with respect to the above problems, but is also able to interpret each community using more than one topic, which provides richer explanations. We validate the robustness of this proposed new approach on an artificial benchmark, and test its interpretability using a case study analysis. We finally show its definite superiority for community detection by comparing with seven state-of-the-art algorithms on eight real networks.**

Keywords—Social networks, attributed network, community detection, generalized communities, probabilistic model, EM algorithm, semantics

I. Introduction

Complex systems, in which basic units interact with each other and work collectively to form a global object, occur in a large variety of contexts. For example, our lively human society sees people entering into various relationships, and online social media link geographically dispersed users. A network is the simplest representation of such complex systems. It abstracts basic units as nodes and relationships between them as edges. Thanks to this simplification, the network model provides a perspective for understanding complex systems. One of the most important things for understanding complex systems is to identify *communities*. These can be assortative communities (i.e. the members of each group are mostly connected to other members of the same group) [1], disassortative communities (i.e. vertices of the same group have most of their connections outside their group, e.g. in bipartite networks) [2] or a combination [3], and so forth. All of these types of structures are collectively called *generalized community structure* [4]. Such structures help people understanding how the network is organized or predicting its behavior, such as finding the political factions in blog networks [5], or identifying the functional modules in protein-protein interaction networks [6]. Thus, the essence of community detection is to identify sets of nodes with common functions, and its true value lies in revealing meaningful and functional substructures.

Traditional community detection algorithms typically use the network topology alone, and only find the assortative communities. Their basic assumption is that functional communities share a common structural signature, i.e. members of each group are mostly connected to other members of the same group, which allows extracting the assortative communities from networks. A wide variety of community detection algorithms using different theories and techniques has been proposed. They include hierarchical clustering methods [7], modularity-based methods [8], [9], spectral algorithms [10], dynamic algorithms [11], statistical inference-based methods [12], [13], etc. For a comprehensive description readers can refer to the survey of Fortunato [14].

Recently, it has also been noticed that content information (i.e. nodes attributes) is valuable for identifying communities. Individuals with similar attributes are more likely to belong to the same community. Unlike network structure which represents the interconnection of individuals, node attributes focus on individual features and provide another modality of useful information for characterizing the nature of communities. These two data modalities complement each other, and lead to more precise detection of communities. In addition, if one source of information is missing or noisy, the other could make up for it. Thus, algorithms combining these two sources of information have been proposed, e.g. topic model-based methods [15], [16], generative or discriminative models [17], and some heuristic methods [18].

With the further development of community detection based on this richer network representation (network with content), some researchers have realized that significant com-

*Corresponding Author: Wenhuan Lu, wenhuan@tju.edu.cn munity detection should not only accurately discover the underlying communities, but also be able to give interpretable and functional descriptions for these communities. Descriptions are meant to explain why certain nodes belong to a common community, or to indicate the functions or characteristics of these communities. With descriptive abilities, community detection can be more valuable for practical applications. It is only very recently that several descriptive algorithms using network with content have been proposed for this task [19-21].

However, currently, these methods often suffer from the following problems, which limit their abilities in terms of detection performance as well as better interpretation of communities. First, the current methods for combining network topology and content information usually assume that a community is a group of nodes that are densely interconnected (i.e. assortative communities). However, in many cases, this is not in line with reality. For example, in a bipartite author-paper network, vertices represent authors or papers in the network, with edges showing their relationship. A community of authors in the same field typically connects with the corresponding community of papers and the network is thus approximately a bipartite network with disassortative communities. In another example considering a foreign trade network, vertices represent businessmen in various countries, with edges showing the business partnership between them. Because this is a foreign trade network and every businessman can typically connect to foreign businessmen, so the network also has a disassortative community structure. However, it is not easy for current methods to detect either assortative and disassortative communities, or their mixture, i.e. a generalized community structure, especially for the networks with node contents.

In addition, existing methods typically assume that the network structure and node contents share the same group memberships, which is often not the case. For example, social relations in Twitter often directly reflect the users' groups, while the user-generated content is diverse [22]. When the contents do not match well with network communities, especially when the contents are completely useless, the performance of these algorithms will degrade. Take social networks as an example. It is often the case that users tend to communicate frequently over certain topical interests, and form each community based on these. So there may be multiple and overlapping topics corresponding to each community. While using the assumption that the network topoloty and node contents share the same community memberships, the current methods often use only one topic to interpret each community, and hence have limited interpretability.

To deal with the above problems altogether, we provide a probabilistic generative model which is learned by using a nested expectation-maximization (EM) algorithm. It describes the generalized communities (based on network topology) and content clusters (based on node contents) as two separate parts, and then explores and models their correlation to improve, as much as possible, each of the communities and clusters based on the other. We then use this correlation as well as the derived topical interests (of the clusters) to more precisely interpret each of the network communities.

The paper is organized as follows. In Section II, we introduce the model which is learned from the data. We then present the experimental evaluations of our new approach in Section III. And finally, we conclude with some discussions in Section IV.

II. The Methods

Here we first present a probabilistic generative model for networks with content. We then learn the model using a nested EM algorithm. And finally, we summarize this algorithm and analyze its time complexity.

A. The Generative Model

An attributed network G can be represented by an adjacency matrix A = $(a_{ij})_{n\times n}$ with a_{ij} = 1 if an edge exists between nodes v_i and v_j, or 0 otherwise; and an attribute matrix U = $(u_{it})_{n\times m}$ with u_{it} = 1 if node v_i has w_t , which represents t^{th} attribute's name, as the t^{th} attribute, or 0 otherwise, where n is the number of nodes and m the number of possible attributes. Here we keep the network undirected and unweighted for simplicity, and use the word "community" to denote generalized community in the following.

Now, we have two objectives: 1) to divide the networked data into communities and content clusters respectively, and 2) to find the correlation between the two for the purpose of better interpreting communities using semantics from content clusters.

More precisely, we divide node set V into k generalized communities, called *network communities* (the nodes within communities have similar connection patterns to others [3]). Also, we partition V into k clusters (the same number as for communities) using mainly content information, and call them *content clusters* (the nodes in the same cluster share similar preferences about their attributes, also called *semantic topic* in the area of topic modeling [23]). Each content cluster thus possesses a topic to represent its semantics. Meanwhile, we derive the correlation between network communities and content clusters, and utilize this correlation to improve, as much as possible, each of the community and cluster based on the other. Then, we can further use this correlation as well as the semantic topics (of the content clusters) to better interpret the network communities.

We bring these goals into a unified probabilistic generative model which is compactly represented in Figure 1. The model is specified by three types of quantities. The first type is the observed quantities, which include adjacency matrix A and attribute matrix U. The second is the hidden quantities, including community assignments z, where z_i denotes the label of the community to which node v_i belongs, and topic (or cluster) assignments g where g_i denotes the label of the topic to which node v_i's attributes belong (this is also the label of the content cluster to which node v_i belongs). The last type is model parameters, which include $\gamma = (\gamma_r)_{1\times k}$, where $\gamma_r = p(z_i = r)$ is the probability that node v_i belongs to the r^{th} community; $\Theta = (\theta_{rj})_{k\times n}$, where $\theta_{rj} = p(a_{ij} = 1 | z_i = r)$ denotes the probability that node v_i has a link to node v_j when node v_i lies in the r^{th} community; $\Omega = (\omega_{rs})_{k\times k}$, where $\omega_{rs} = p(g_i = s|z_i = r)$ denotes the probability that node v_i lies in the s^{th} content cluster when it belongs to the r^{th} network community; and $\Phi = (\varphi_{st})_{k\times m}$, where $\varphi_{st} = p(u_{it} = 1|g_i = s)$ is the probability that node v_i has the t^{th} attribute when it belongs to the s^{th} cluster, which is especially suitable for short texts such as those in real social networks. Here each cluster has a topic to denote its semantics. The notation is represented in Figure 1 and is generated as follows.

For each node v_i:
(a) Draw community assignment $z_i \sim \textit{Multinomial}(\gamma)$
(b) For each node v_j with $a_{ij} = 1$:
i. Draw edge $a_{ij} \sim \textit{Multinomial}(\theta_{z_i})$
(c) Draw topic assignment $g_i \sim \textit{Multinomial}(\omega_{z_i})$
(d) For each of the t^{th} attribute with $u_{it} = 1$:
i. Draw attribute $w_t \sim \textit{Multinomial}(\varphi_{g_i})$

Then the probability, or likelihood, that this attributed network G was generated by this model, given the parameters, is:

$$\begin{aligned} &P(\mathrm{A},\mathrm{U}\mid\gamma,\Theta,\Omega,\Phi)\\ &=\sum_{\mathrm{z},\mathrm{g}} P(\mathrm{z}\mid\gamma)P(\mathrm{A}\mid\mathrm{z},\Theta)P(\mathrm{g}\mid\Omega,\mathrm{z})P(\mathrm{U}\mid\Phi,\mathrm{g}) \\ &=\sum_{\mathrm{z},\mathrm{g}}\left(\prod_{i=1}^{n}\gamma_{z_i}\prod_{i=1}^{n}\prod_{j=1}^{n}\left(\theta_{z_i,j}\right)^{a_{ij}}\prod_{i=1}^{n}\omega_{z_i,g_i}\prod_{i=1}^{n}\prod_{t=1}^{m}\left(\varphi_{g_i,t}\right)^{u_{it}}\right)\end{aligned} \tag{1}$$

subject to $\sum_{r=1}^{k}\gamma_r = 1$, $\sum_{j=1}^{n}\theta_{rj} = 1$ for r = 1…k, $\sum_{s=1}^{k}\omega_{rs} = 1$ for r = 1…k, and $\sum_{t=1}^{m}\varphi_{st} = 1$ for s = 1…k.

As we can see, there are mainly 4 parts in Eq. (1). The first two parts are the fitting to the network structure, the third part is the a priori probability that generates the content clusters (with their topics) under the derived network communities, and the fourth part is the fitting to the content information. In general, the fittings to network and content are dominant in the likelihood, while the a priori plays a guiding role to incorporate these two parts and improve the results of each other.

Here remember our motivations. First, θ_{rj} represents the "preferences" of vertices in the r^{th} network community about which other vertices they link to. In particular, we do not assume that members of a community link to nodes v_i which belong to any particular community or communities. They can be in the same community, in several communities or, more generally, distributed over the entire network. That is to say, nodes in the same community have similar link patterns (e.g. the assortative or disassortative structure), so as to find the generalized community structures in networks.

Furthermore, the correlation matrix Ω, which is the transition probability matrix from the network communities to content clusters, plays a vital role to achieve our goals. To be specific, when the network and node contents match well in terms of community structure, correlation matrix Ω will be almost an identity matrix, we can then easily improve the results by incorporating both information sources. Even if the network communities and content clusters do not match well, by exploiting the relationship Ω, the model can still utilize what information is present to improve results. While, if content clusters do not match with network communities at all, correlation matrix Ω will be very fuzzy, so that the model can almost automatically ignore the content, returning results based on network alone. In addition, we can also use the correlation matrix Ω, combined with the topics derived, to interpret each community with more than one topic, which is thus more precise.

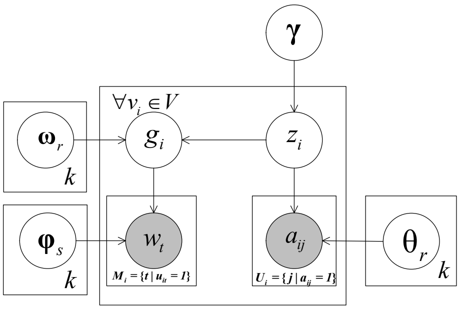


Fig. 1. Diagram of the generative model

B. Fitting to the Data

Our goal is now, given the observed data, to maximize the likelihood in (1) to find the best-fit parameters. Rather than maximizing (1) itself, we instead maximize the log likelihood.

$$L = \log\sum_{\mathrm{z},\mathrm{g}} P(\mathrm{z}\mid\gamma)P(\mathrm{A}\mid\mathrm{z},\Theta)P(\mathrm{g}\mid\Omega,\mathrm{z})P(\mathrm{U}\mid\Phi,\mathrm{g}) \tag{2}$$

Since direct maximization of (2) is nontrivial, we adopt an expectation-maximization (EM) algorithm. By applying Jensen's inequality to (2), we obtain the expected log likelihood.

$$\begin{aligned} &L \ge \overline{L}\\ &=\sum_{\mathrm{z}} q(\mathrm{z})\log\frac{\sum_{\mathrm{g}} P\left(\mathrm{z}\mid\gamma\right)P(\mathrm{A}\mid\mathrm{z},\Theta)P(\mathrm{g}\mid\Omega,\mathrm{z})P(\mathrm{U}\mid\Phi,\mathrm{g})}{q(\mathrm{z})}\\ &=\sum_{i=1}^{n}\sum_{r=1}^{k} q_{ir}\left(\log\gamma_r + \sum_{j} a_{ij}\log\theta_{rj} + \log\left(\sum_{s=1}^{k}\omega_{rs}\prod_{t=1}^{m}\left(\varphi_{st}\right)^{u}\right) - \log q_{ir}\right)\end{aligned} \tag{3}$$

where $q(\mathrm{z})$ is any distribution over community assignments z such that $\sum_{\mathrm{z}} q(\mathrm{z}) = 1$, $q_{ir} = \sum_{\mathrm{z}} q(\mathrm{z})\delta_{z_i r}$ is the marginal probability within $q(\mathrm{z})$ that node v_i belongs to community r, and δ_{rs} is the Kronecker delta.

Then, the maximum of $\overline{L}$ with respect to all possible choices of distribution $q(\mathrm{z})$ will be obtained when $L = \overline{L}$, which, following Jensen's inequality, is when

$$q\left(\mathrm{z}\right) = \frac{\sum_{\mathrm{g}} P\left(\mathrm{z}\mid\gamma\right)P(\mathrm{A}\mid\mathrm{z},\Theta)P(\mathrm{g}\mid\Omega,\mathrm{z})P(\mathrm{U}\mid\Phi,\mathrm{g})}{\sum_{\mathrm{z},\mathrm{g}} P\left(\mathrm{z}\mid\gamma\right)P(\mathrm{A}\mid\mathrm{z},\Theta)P(\mathrm{g}\mid\Omega,\mathrm{z})P(\mathrm{U}\mid\Phi,\mathrm{g})} \tag{4}$$

Thus, the maximization of likelihood L with respect to γ, Θ, Ω and Φ to obtain optimal parameter values is equivalent to a maximization of its lower bound $\overline{L}$ with respect to both $q(\mathrm{z})$ (making $L = \overline{L}$) and the parameters. The EM algorithm for performing such a double maximization is to repeatedly maximize with respect to first $q(\mathrm{z})$ (i.e. the E-step) and then γ, Θ, Ω and Φ (i.e. the M-step), which is known to monotonically converge to a local maximum.

For the E-step, we need to make $L = \overline{L}$. So, from Eq. (3) we can get the optimal q_{ir}'s by using:

$$
\begin{aligned}
q_{ir} &= \sum_{\mathrm{z}} q(\mathrm{z})\delta_{z_i r} \\
&= P\left(z_i = r \mid \mathrm{A}, \mathrm{U}, \Omega, \Phi, \gamma, \Theta\right) \\
&= \frac{P\left(\mathrm{A}, z_i = r, \mathrm{U} \mid \Omega, \Phi, \gamma, \Theta\right)}{P\left(\mathrm{A}, \mathrm{U} \mid \Omega, \Phi, \gamma, \Theta\right)} \\
&= \frac{\gamma_r \prod_{j=1}^{n}\left(\theta_{rj}\right)^{a_{ij}} \sum_{s=1}^{k} \omega_{rs} \prod_{t=1}^{m}\left(\varphi_{st}\right)^{u_{it}}}{\sum_{r=1}^{k} \gamma_r \prod_{j=1}^{n}\left(\theta_{rj}\right)^{a_{ij}} \sum_{s=1}^{k} \omega_{rs} \prod_{t=1}^{m}\left(\varphi_{st}\right)^{u_{it}}}
\end{aligned}
\tag{5}
$$

However, the M-step is nontrivial because the expected log-likelihood $\overline{L}$ contains some latent quantities *s*.

1) M-step with a nested EM process.

Now we need to perform the maximization of $\overline{L}$ in (3) over the parameters, with a fixed q_{ir}. Maximization of the γ and Θ is straightforward. Differentiating $\overline{L}$ with respect to γ_r, subject to the normalization condition $\sum_{r=1}^{k} \gamma_r = 1$, gives

$$
\gamma_r = \frac{1}{n}\sum_{i=1}^{n} q_{ir} \tag{6}
$$

And then, computing the derivative, setting the result to zero and satisfying to the normalization condition $\sum_{j=1}^{n} \theta_{rj} = 1$ for $r = 1\ldots k$, we find that the maximum with respect to θ_{rj} is obtained for

$$
\theta_{rj} = \frac{\sum_{i=1}^{n} q_{ir} a_{ij}}{\sum_{i=1}^{n} q_{ir} d_i} \tag{7}
$$

where $d_i = \sum_{j=1}^{n} a_{ij}$.

However, maximization of $\overline{L}$ with respect to Ω and Φ is a little more complicated since $\overline{L}$ contains latent variables *s*. Again, we use a nested expectation-maximization (EM) process, and apply Jensen's inequality to (3):

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{r=1}^{k} q_{ir} \log \sum_{s=1}^{k} \omega_{rs} \prod_{t=1}^{m} \left(\varphi_{st}\right)^{u_{it}} \geq \\
&\qquad \sum_{i=1}^{n}\sum_{r=1}^{k} q_{ir} \sum_{s}^{k} t_{ir}^{s} \log \frac{\omega_{rs} \prod_{t=1}^{m}\left(\varphi_{st}\right)^{u_{it}}}{t_{ir}^{s}}
\end{aligned}
\tag{8}
$$

where t_{ir}^{s} can be any distribution, subject to $\sum_{s=1}^{k} t_{ir}^{s} = 1$. Note that here we ignore the terms in $\overline{L}$ which can be regarded as constant with respect to Ω and Φ.

The exact equality of (8), and hence the maximum of the right-hand side is achieved when:

$$
t_{ir}^{s} = \frac{\omega_{rs} \prod_{t=1}^{m}\left(\varphi_{st}\right)^{u_{it}}}{\sum_{s=1}^{k} \omega_{rs} \prod_{t=1}^{m}\left(\varphi_{st}\right)^{u_{it}}} \tag{9}
$$

As before, we can maximize the left-hand side of (8) by repeatedly maximizing the right-hand side with respect to t_{ir}^{s} using (9) and with respect to Ω and Φ by differentiation. Differentiating the right-hand side of (8) with respect to ω_{rs}, subject to $\sum_{s=1}^{k} \omega_{rs}$ for $r = 1\ldots k$, gives:

$$
\omega_{rs} = \frac{\sum_{i=1}^{n} q_{ir} t_{ir}^{s}}{\sum_{i=1}^{n} q_{ir}} \tag{10}
$$

Similarly, differentiating with respect to φ_{st}, subject to $\sum_{t=1}^{m} \varphi_{st} = 1$ for $s = 1\ldots k$, gives:

$$
\varphi_{st} = \frac{\sum_{i=1}^{n}\sum_{r=1}^{k} q_{ir} t_{ir}^{s} u_{it}}{\sum_{i=1}^{n}\sum_{r=1}^{k} q_{ir} t_{ir}^{s} K_i} \tag{11}
$$

where $K_i = \sum_{t=1}^{m} u_{it}$. Then, the optimal Ω and Φ can be calculated by iterating (9), (10) and (11) from an initial condition, alternatively, until convergence.

C. *Algorithm Summary and Time Complexity*

We summarize this nested EM algorithm in Algorithm 1 as follows. Here we let numbers of iterations T_1 = 100 and T_2 = 20. In the early stages of this algorithm, T_2 = 20 gives rather crude values for posterior probabilities q_{ir} 's and parameters Ω and Φ, but these values would not be particularly good under any situation, no matter how many steps were used, because of the poor current solution. In the later stages of this algorithm, 20 steps are enough to ensure good convergence. In addition, as the algorithm may converge to local minima, we ran it 20 times and reported the result with largest likelihood which is often stable and obtain good results.

```
Algorithm 1: Nested EM algorithm
Input: A, U and $k$
Output: $q_{ir}$'s, Ω and Φ
Initialize γ, Θ, Ω and Φ randomly
For $t_1$ = 1: $T_1$ //main EM
    Update one-node marginal probabilities $q_{ir}$'s via (5)
    Update γ and Θ using (6) and (7)
    For $t_2$ = 1: $T_2$ //nested EM
        Update $t_{ir}^{s}$ 's, Ω and Φ using (9), (10) and (11)
    End
End
```

When we get the optimal q_{ir}'s, Ω and Φ, we can use q_{ir}'s where q_{ir} is the posterior probability that node v_i belongs to community *r* to find the network community structure. Thereafter, we can use Φ, where φ_{st} is the probability that topic *s* selects the t^{th} attribute (or word), to derive the topic for each content cluster. Also, we can further use correlation matrix Ω, where ω_r is the distribution of the content clusters (and their topics) over community *r*, to find the dominant topics for each community, and thus we can interpret the communities more precisely.

Now we give the complexity analysis of this algorithm taking into account data sparsity, i.e. matrices A and U. First, the time to update q_{ir}'s, γ and Θ once via (5), (6) and (7) is $2ek+fk^2$, nk and $2ek$ respectively, where n is the number of nodes, k the number of communities and clusters, e the number of edges in the network, $f = \sum_{i=1}^{n} K_i$ the number of content attributes and $K_i = \sum_{t=1}^{m} u_{it}$ the number of attributes of node v_i. Then, the time to compute t_{ir}^{s} 's, Ω and Φ once via (9),

(10) and (11) is fk^2, nk^2, and fk^2, respectively. Also, the time to compute the likelihood function once is $2nk+2ek+fk^2$. Thus, the time complexity of this algorithm is $O(ek+fk^2)$, which is nearly linear on large and sparse networks with content.

III. Experiments

Here we first give an artificial benchmark to evaluate the effectiveness of the proposed new method in terms of our first motivation. We then use an online music system to validate whether this method can interpret the communities more precisely in a manner suitable for mismatched networked data. Finally, we test its performance on eight real attributed networks, and compare it with seven state-of-the-art methods. This is to show whether our new method is able to find the generalized community structure (e.g. assortative or disassortative communities, or their mixture) which would make it more flexible, and whether it can utilize the content information to better improve community detection results.

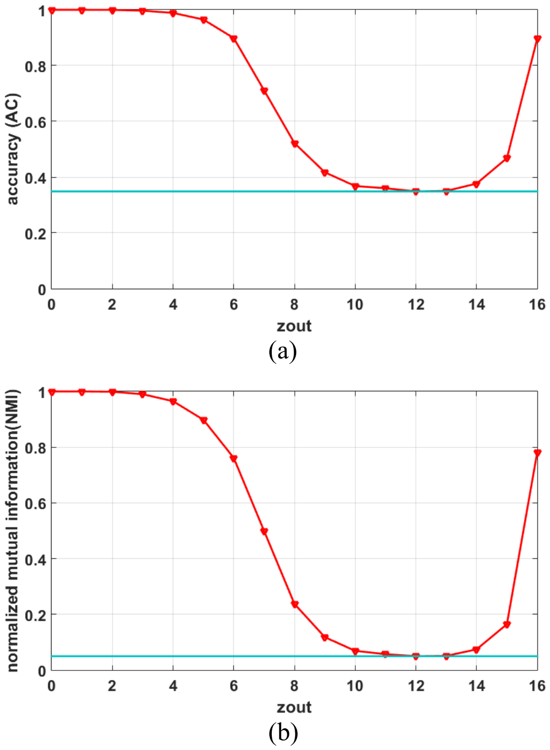
A. *Artificial Benchmarks*

We first introduce the benchmark used. First, we use the Newman's model [1] to generate networks. The graph consists of 128 nodes divided into 4 same size communities. Each node has on average z_{in} edges connecting it to members of the same community and z_{out} edges to members of other communities, with $z_{\text{in}} + z_{\text{out}} = 16$. Notice that p_{in} (= $z_{\text{in}}/32$) > p_{out} (= $z_{\text{out}}/96$) means that the connection probability of nodes within the community is larger than that between communities, i.e. assortative community structure; on the contrary, $p_{\text{in}} < p_{\text{out}}$ (i.e. $z_{\text{out}} > 12$) corresponds to the case when generated networks have disassortative communities; and $p_{\text{in}} = p_{\text{out}}$ (i.e. $z_{\text{out}} = 12$) denotes that the networks do not have any community structure.

Thereafter, we generated a $4h$-dimensional binary attributes for each node v_i (i.e. u_i) to form 4 content nodes clusters, corresponding to the 4 network communities. To be specific, for each node within the s^{th} cluster, we use a binomial distribution with mean $\rho_{\text{in}} = h_{\text{in}}/h$ to generate a h-dimensional binary vector as its $((s-1) \times h + 1)$-th to $(s \times h)$-th attributes, and generate the rest of the attributes using a binomial distribution with mean $\rho_{\text{out}} = h_{\text{out}}/(3h)$. Then, $\rho_{\text{in}} > \rho_{\text{out}}$ means that these generated h-dimensional attributes are associated with the s^{th} cluster with higher probability, while the remaining $3h$ attributes are irrelevant (or say noise attributes). Here we set the dimension of attributes $4h = 200$, and the average number of attributes w_t with $u_{it} = 1$ for each node v_i to be $h_{\text{in}} + h_{\text{out}} = 16$.

Here we try to validate our first motivation, which is to detect the generalized community structure. We set $h_{\text{out}} = 8$, and vary z_{out} from 0 to 16 with an increment of 1. Note that, when z_{out} increases from 0 to 12, the assortative structure becomes weaker and weaker. Especially, when $z_{\text{out}} = 12$ (meaning $p_{\text{in}} = p_{\text{out}}$), the topology does not have any community property. In contrast, when z_{out} increases from 12 to 16, the disassortative structure becomes more and more clear.

The results are shown in Figure 2. The blue line denotes the minimum of all points. As we can see, when $z_{\text{out}} = 12$, the accuracy values of our method in terms of both AC and NMI (see definitions below) is minimum. This is because the topology does not have any community property when $z_{\text{out}} = 12$, and thus the task of community detection is only based upon the content information. But when z_{out} continues to increase, the values of AC and NMI of our method both increase, which corresponds to the disassortative community structure. So to sum up, these results validate that: our method can detect not only assortative communities, but also find disassortative ones (i.e. the *generalized community structure*), which makes it fit to find real generalized community structures.



(a)

(b)

Fig. 2. (a) The AC (accuracy) index [24] of our method as a function of the nodes average outside-community degree (z_{out}). (b) The NMI (normalized mutual information) index [24] of our method as a function of the average of nodes outside-community degree (z_{out}).

B. *A Case Study on Last.fm*

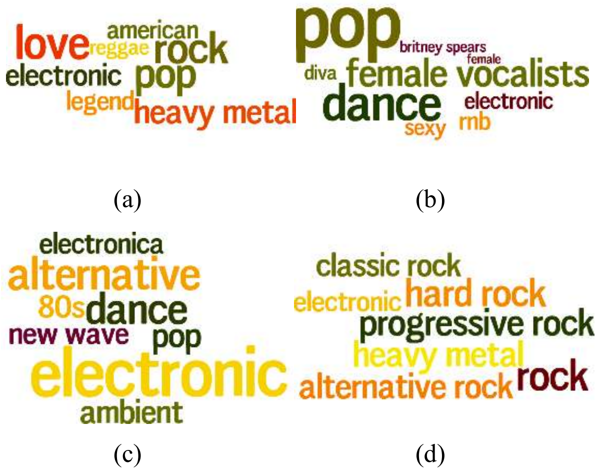
Here we try to further validate our second motivation, namely, whether the derived correlation (Ω) between network communities and the topics of content clusters (i.e. the distribution of each community's topics) can help us better interpret communities. This would make our method more suited for situations when the node contents do not match well with network communities. Here we used the dataset [25] from an online music system *Last.fm*, where 1,892 users are connected in a social network generated from *Last.fm* "friend" relations. Each user has 11,946-dimensional attributes, including a list of most listened music artists, and tag assignments. As the network does not have ground-truth of communities, we set the number of communities to 38 as was done in [26].

After one run of our algorithm, we found 11 communities with one dominant topic each, as well as 27 communities with more topics. After a detailed analysis, we find that most of these communities are topically meaningful. But due to limited space here, we only give three examples: 2 communities with one dominant topic each, and 1 community with two topics. The word clouds formed by the dominant attributes of each topic are shown in Figure 3. The size of a word is proportional to the probability that it belongs to this topic.

Our first example is the 1st community which possesses one dominant topic, topic 23, as shown in Figure 3(a). As we can see, this community is a group of fans of reggae music. To be more specific, in the word cloud of topic 23, “american” and “rock” show that reggae music has become important in the mainstream of American rock music. “love” is a theme that the reggae music always expresses. Reggae music evolved from “pop” music. “electronic” and “heavy metal” are also related to reggae music.

The second example is the 11th community whose main topic is topic 10, as shown in Figure 3(b). This topic is mainly related to Britney Spears, a famous US female singer, so that the community may be a group of her fans. Britney Spears is sexy, so “female vocalists”, “female”, “diva” and “sexy” all appear here. Besides, the words “pop”, “dance”, “rnb” and “electronic” also correspond to Britney Spear’s music style.

The third example is community 19 which contains two dominant topics, topics 18 and 23. To be specific, topic 18, as shown in Figure 3(c), is highly related to “ambient” which is a kind of electronic music. It has 80s trance style of electronic dance music, so “80s” and “dance” both appear here. “alternative” and “new wave” are in the nature of ambient. Topic 23 shown in Figure 3(d) mainly refers to rock, as discussed above in Figure 3(a). The sound of rock music is mainly produced by electric guitar. Rock music is a kind of electronic music actually. Other words like “hard rock”, “progressive rock” and “alternative rock” are all highly related to rock music. It is worth noting that these two topics, which correspond to ambient and rock, respectively, both belong to electronic music, although being different branches. Therefore, this community will be a group of fans of ambient and rock under the style of electronic music.



(a) (b)

(c) (d)

Fig. 3. Three examples to interpret communities. (a) is the word cloud of topic 23 which is the main topic of the 1st community. (b) is topic 10, the main topic of the 11th community. (c) and (d) are the main topics (topic 18 and 23) of the 19th community.

So, this case study analysis not only validates that our derived communities are topically meaningful, but also shows that allowing communities with more than one topic can help us interpreting and understanding them better (as in the third example). In summary, this further supports our hypothesis that users tend to communicate frequently over certain topical interests and form communities based on those interests.

C. *Real-world Networks*

Here we want to show whether our method is more flexible for community identification owing to the fact that we can find the generalized community structure (e.g. assortative, disassortative communities, or their mixture); and also whether we can utilize content information to improve the community detection results. For these purposes, we evaluate the performance of our method on eight real-world attributed networks with ground truth of communities ([29], [30]). These networks include social networks, information networks as well as citation networks, scaling from hundreds to tens of thousands of nodes, as shown in Table 1.

1) *Evaluation Metrics*

Because the networks used have ground-truth communities, we can adopt accuracy (AC) [24] and normalized mutual information (NMI) [24] to compare the detected and ground-truth communities.

If the set of detected communities is C and the set of ground-truth communities is C^*, and accuracy AC is defined as:

$$AC(C,C^*)=\frac{\sum_{i=1}^{n}\delta\left(map\left(C_i^*\right),map\left(C_i\right)\right)}{n} \tag{12}$$

where $\delta(a, b)$ is the delta function that equals to 1 if $a = b$ and 0 otherwise, and $map(C_i^*)$, resp. $map(C_i)$, is the function that maps each community C_i^*, resp. C_i to the index of the community i belongs to in C^*, resp. C, n is the number of nodes.

On the other hand, the normalized mutual information (NMI) is defined as:

$$NMI(C,C^*)=\frac{MU(C,C^*)}{\max(H(C),H(C^*))} \tag{13}$$

where

$$H(C)=\sum_{C_i}P(C_i)\log(P(C_i)) \tag{14}$$

is the entropy of the set of communities C and where $P(C_i)=|C_i|/|C|$ and:

$$MU(C,C^*)=\sum_{C_i,C_j^*}p\left(C_i,C_j^*\right)\log\frac{p(C_i,C_j^*)}{p(C_i)p(C_j)} \tag{15}$$

is the mutual information between C and C* and where

$$P(C_i,C_j^*)=\left|C_i\cap C_j^*\right|\Big/\left|C_i\right| \tag{16}$$

However, some of the baseline methods used in our evaluations provide overlapping community structures which cannot be compared in terms of AC and NMI. Thus, we also adapt the metrics used in [20] to evaluate overlapping communities, namely F-score or Jaccard similarity, i.e., we evaluated a set of detected communities C with ground-truth communities C^* by

$$\frac{1}{2|C^*|}\sum\nolimits_{C_i^*\in C^*}\max\nolimits_{C_j\in C}\delta\left(C_i^*,C_j\right) + \frac{1}{2|C|}\sum\nolimits_{C_j\in C}\max\nolimits_{C_i^*\in C^*}\delta\left(C_i^*,C_j\right) \tag{17}$$

where $\delta\left(C_i^*,C_j\right)$ is a similarity measure (F-score or Jaccard) between C_i^* and C_j.

2) *Experimental Evaluation*

We consider three types of community detection methods for comparison. The first type includes DCSBM [13] and BigCLAM [27], which use network topology alone. The second includes SMR [28], using only node contents. And the third includes Block-LDA [15], PCL-DC [17], CESNA [20], and DCM [19], which combine both network structure and node contents. As shown in Tables 2 and 3, there are two families of methods: some produce non-overlapping communities (Table 2) while others produce overlapping communities (Table 3). We thus want to see whether our method outperforms each of these three types and two families of community detection methods. All of these methods require the number of communities to be specified. We set it the same as that of the ground truth, so that each model detects the same number of communities. We use default values for other parameters of these algorithms, as offered by their original authors.

As shown in Table 2, our method outperforms all the baseline algorithms (for finding disjoint communities) on 5 and 6 out of 8 networks in terms of AC and NMI, respectively. As shown in Table 3, our method performs best on all the 8 networks, in terms of F-score and Jaccard, compared with all the baselines for detecting overlapping communities. In addition, our method also performs the second best on Washington in terms of AC, as well as on Pubmed in terms of NMI.

However, the strong performance of our method is not obvious, as it would be entirely possible that combining two sources of data would confuse the algorithm and degrade the overall performance. While the strong performance of our algorithm remains to be further investigated, we believe it may be due to two main properties: 1) our method can find generalized community structure (e.g. assortative, disassortative communities, or their mixture) which makes it more flexible for community identification; 2) our method does not assume that the network and node contents share the same community memberships, leading to the fact that even if the content clusters do not match well with network communities, the method can still utilize the content information as much as possible to improve the community detection results.

TABLE 1. DATASETS USED. N IS THE NUMBER OF NODES, E THE NUMBER OF EDGES, M THE NUMBER OF ATTRIBUTES, AND K THE NUMBER OF COMMUNITIES.

Datasets	*n*	*e*	*m*	*k*	Descriptions [29], [30]
Texas	187	328	1,703	5	The WebKB network consists of 4 subnetworks
Cornell	195	304	1,703	5	from 4 universities,
Washington	230	446	1,703	5	which are Texas, Cornell, Washington and
Wisconsin	265	530	1,703	5	Wisconsin, respectively
Twitter	171	796	578	7	Largest subnetwork (id 629863) in Twitter data
Facebook	1,045	26,749	576	9	Largest subnetwork (id 107) in Facebook data
Cora	2,708	5,429	1,433	7	A Cora citation network
Pubmed	19,729	44,338	500	3	Publications in PubMed on diabetes

TABLE 2. COMPARISON OF ALGORITHMS WITH DISJOINT COMMUNITY STRUCTURES, IN TERMS OF AC AND NMI, RESPECTIVELY. BOLD REPRESENTS BEST RESULT, AND THE MARKS AFTER THE RESULT OF OUR METHOD CORRESPONDS TO ITS RANK IN TERMS OF PERFORMANCE, IF IT IS NOT 1ST.

Metrics (%)	Methods	Datasets							
		Texas	Cornell	Washington	Wisconsin	Twitter	Facebook	Cora	Pubmed
AC	DCSBM	48.09	37.95	31.80	32.82	60.49	45.19	38.48	53.64
	SMR	47.54	31.79	49.77	40.84	38.27	52.46	30.28	39.95
	Block-LDA	54.10	46.15	39.17	49.62	35.80	37.66	25.52	49.01
	PCL-DC	38.80	30.26	29.95	30.15	56.79	51.04	34.08	63.55
	Ours	**61.20**	**46.16**	46.54(2)	37.68(3)	**61.73**	**59.83**	**43.28**	48.55
NMI	DCSBM	16.65	9.69	9.87	3.14	57.48	43.38	17.07	12.28
	SMR	3.55	8.45	7.3	7.21	3.26	14.90	1.18	0.0367
	Block-LDA	4.21	6.81	3.69	10.09	0	9.28	1.41	6.58
	PCL-DC	10.37	7.23	5.66	5.01	52.64	38.63	17.54	26.84
	Ours	**19.73**	**10.70**	**13.66**	4.54	**57.52**	**50.37**	**20.28**	13.48(2)

TABLE 3. COMPARISON OF ALGORITHMS WITH OVERLAPPING COMMUNITY STRUCTURES, IN TERMS OF F-SCORE AND JACCARD, RESPECTIVELY. BOLD REPRESENTS BEST RESULT.

Metrics (%)	Methods	Datasets							
		Texas	Cornell	Washington	Wisconsin	Twitter	Facebook	Cora	Pubmed
F-score	BigCLAM	20.64	13.23	13.35	12.84	39.79	40.06	18.89	7.72
	CESNA	23.54	23.48	21.91	23.17	43.72	49.05	31.05	27.97
	DCM	11.15	14.38	12.45	10.45	10.57	39.21	3.43	0.38
	Ours	**36.64**	**33.51**	**31.05**	**30.02**	**51.55**	**51.70**	**42.15**	**46.46**
Jaccard	BigCLAM	12.18	7.18	7.25	7.01	26.13	28.94	10.89	4.04
	CESNA	13.57	13.47	12.40	13.14	29.63	38.18	19.10	16.26
	DCM	6.03	7.95	6.72	5.54	5.75	28.46	1.76	0.19
	Ours	**25.77**	**21.58**	**19.34**	**18.39**	**38.40**	**39.75**	**27.67**	**31.03**

IV. CONCLUSION AND DISCUSSION

In this paper, we propose a probabilistic generative model for attributed networks, learned via a nested EM algorithm, which is designed for finding generalized community structures. The model describes network communities and content clusters separately, and then explores and models their relationship to improve, as much as possible, each of the communities and the clusters (with their topics) based on the other. By doing so, its performance will not degrade when the content does not match well with network communities, even if the content is completely useless or harmful for community identification. Furthermore, we can also use the derived correlation between network communities and content clusters, as well as the topical interests (from the clusters) to better interpret each community using more than one topic. We validate these two goals using an artificial benchmark as well as a case study analysis on Last.fm. We finally validate the effectiveness of our new algorithm to detect communities on eight real attributed networks, and compare with seven state-of-the-art methods.

We thus show that our method works well on a variety of networks, demonstrating its versatility to exploit both network structure and content. Obtaining an interpretation of communities (i.e. semantics) is a significant achievement of our work and may be useful to site managers for marketing purposes.

One might now be concerned that our method finds network communities mainly based on network topology, but gives node content an auxiliary role. This is partly true and it is often believed that social relations reflect the user interests directly. For example, an Obama supporter will be more likely to follow the Democrats. Therefore, social relations can serve as a good indicator for communities. But the user-generated content in social networks such as Twitter is diverse, and therefore it may be suitable as an auxiliary role.

In addition, our method needs the number of communities to be given, which may be difficult to determine in practice. This is a so-called model selection problem, which may be solved using cross-validation or hierarchical Bayesian methods. Also, the number of communities may be different from the number of clusters (or topics). Besides, the number of communities c and topics k may be different, and our method is also suitable when $c \neq k$. But we will leave these as our main future work.

ACKNOWLEDGMENT

This work was supported by Natural Science Foundation of China (61502334, 61772361, 61471259, 61673293).

REFERENCES

[1] M. Girvan, and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Natl. Acad. Sci. USA*, vol. 12, p. 7821-7826, Apr. 2002.

[2] P. Holme, F. Liljeros, C. R. Edling, and B. J. Kim, "Network bipartivity," *Phys. Rev. E.*, vol 68, p. 056107, Nov. 2003.

[3] M. E. J. Newman and E. A. Leicht, "Mixture models and exploratory analysis in networks," *Proc. Natl. Acad. Sci. USA*, vol. 104, p. 9564-9569, Jun. 2007.

[4] M. E. J. Newman and T. P. Peixoto, "Generalized communities in networks," *Phys. Rev. Lett.*, vol. 115, p. 088701, May. 2015.

[5] P. J. Bickel and P. Sarkar, "Hypothesis testing for automated community detection in networks," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol.78, p. 253-273, May, 2015.

[6] S. Vlaic, C. Tokarski-Schnelle, M. Gustafsson, U. Dahmen, R. Guthke, and S. Schuster, "*Module Discoverer: Identification of regulatory modules in protein-protein interaction networks*," bioRxiv, 119099, March, 2017.

[7] S. Jia, L. Gao, Y. Gao, J. Nastos, Y. Wang, X. Zhang, and H. Wang, "Defining and identifying cograph communities in complex networks," *New Journal of Physics*, vol 17, p. 013044, Jan. 2015.

[8] L. Yang, X. Cao, D. He, and W. Zhang, "Modularity based community detection with deep learning," In *Proceedings of 25th International Joint Conference on Artificial Intelligence* (IJCAI'16), New York, USA, July 9-15, 2016, pp. 2083-2089.

[9] G. Pan, W. Zhang and Z. Wu, "Online community detection for large complex networks," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence* (IJCAI'13), Beijing, China, August 03-09, 2013, pp. 1903-1909.

[10] B. Yang, J. Liu and J. Feng, "On the spectral characterization and scalable mining of network communities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, p. 326-337, Feb. 2012.

[11] M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and Lambiotte. R., "Memory in network flows and its effects on spreading dynamics and community detection," *Nature Communications*, vol 5, p. 4630, Aug. 2014.

[12] A. Anandkumar, R. Ge, D. Hsu, and S. M. Kakade, "A tensor approach to learning mixed membership community models," *Journal of Machine Learning Research*, vol. 15, p. 2239-2312, Oct. 2014.

[13] B. Karrer and M. E. J. Newman, "Stochastic blockmodels and community structure in networks," *Phys. Rev. E*, vol. 83, p. 016107, Jan, 2011.

[14] S. Fortunato and D. Hric, "Community detection in networks: A user guide, "*Physics Reports*, vol. 659, p. 1-44, Nov. 2016.

[15] R. Balasubramanyan and W. W. Cohen, "Block-LDA: Jointly modeling entity-annotated text and entity-entity links," In *Proceedings of 11th SIAM International Conference on Data Mining* (SIAM'11), Hilton Phoenix east, Arizona, USA, April 28-30, 2011, pp. 450-461.

[16] R. Nallapati, A. Ahmed, E. Xing, and W. Cohen, "Joint latent topic models for text and citations," In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD'08), Las Vegas, USA, August 24-27, 2008, pp. 542-550.

[17] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: A discriminative approach," In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*(KDD'09), Paris, France, June 28-July 1, 2009, pp. 927-936.

[18] Y. Ruan, D. Fuhry and S. Parthasarathy, "Efficient community detection in large networks using content and links," In *Proceedings of the 22nd International World Wide Web Conference* (WWW'13), Rio de Janeiro, Brazil, May 13-17, 2013, pp. 1089-1098.

[19] S. Pool, F. Bonchi, and M. Leeuwen, "Description-driven community detection." *ACM Transactions on Intelligent Systems and Technology*, vol. 5, p.1-28, Apr. 2014.

[20] D. He, Z. Feng, and D. Jin, "Joint Identification of Network Communities and Semantics via Integrative Modeling of Network Topologies and Node Contents," In *Proceedings of the 31th AAAI Conference on Artificial Intelligence*(AAAI'17), San Francisco, California USA, February 4–9, 2017, pp. 116-124.

[21] L. Liu, L. Xu, Z. Wang, and E. Chen, "Community detection based on structure and content: A content propagation perspective," In *Proceedings of the 15th IEEE International Conference on Data Mining* (ICDM'15), Atlantic City, New Jersey, USA, November 14-17, 2015, pp. 271-280.

[22] Y. Pei, N. Chakraborty, and K. Sycara, "Nonnegative matrix tri-factorization with graph regularization for community detection in social networks," In *Proceedings of the 24th International Joint Conference on Artificial Intelligence* (IJCAI'15), Buenos Aires, Argentina, July 25-31, 2015, pp. 2083-2089.

[23] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J. Mach. Learn. Res.*, vol. 3, p. 993-1022, Jan. 2003.

[24] H. Liu, Z. Wu, X. Li, D. Cai, and T. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Trans. Pat. Anal. Mach. Intel.*, vol. 34, p. 1299-1311, Jul. 2012.

[25] I. Cantador, *The 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems.* http://ir.ii.uam.es/hetrec2011/datasets.html.

[26] X. Wang, D. Jin, X. Cao, L. Yang and W. Zhang, "Semantic community identification in large attribute networks," In *Proceedings of the 30th AAAI Conference on Artificial Intelligence* (AAAI'16), Phoenix, Arizona USA, February 12-17, 2016, pp. 172-178.

[27] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining* (WSDM'13), Rome, Italy, February 04 - 08, 2013, pp. 587-596.

[28] H. Hu, Z. Lin, J. Feng and J. Zhou, "Smooth representation clustering," In *Proceedings of the 27th Conference on Computer Vision and Pattern Recognition* (CVPR'14), Columbus, Ohio, USA, June 24-27, 2014, pp. 3834-3841.

[29] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad, "Collective classification in network data," *AI Magazine*, vol. 29, p. 93-106, Jan, 2008.

[30] J. Leskovec, *Stanford Network Analysis Project.* http://snap.stanford.edu, 2016.