Finding Communities with Hierarchical Semantics by Distinguishing General and Specialized Topics

Ge Zhang[1], **Di Jin**[1,*], **Jian Gao**[2], **Pengfei Jiao**[1], **Francoise Fogelman-Soulié**[3] and **Xin Huang**[4]
[1] School of Computer Science and Technology, Tianjin University, Tianjin, China
[2] College of Information Science and Technology, Dalian Maritime University, Dalian, China
[3] School of Computer Software, Tianjin University, Tianjin, China
[4] Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
{zge2016, jindi, pjiao}@tju.edu.cn, gaojian@dlmu.edu.cn
francoise.soulie@outlook.com, xinhuang@comp.hkbu.edu.hk

Abstract

Using network topology and semantic contents to find topic-related communities is a new trend in the field of community detection. By analyzing texts in social networks, we find that topics in networked contents are often hierarchical. In most cases, they have a two-level semantic structure with general and specialized topics, to respectively denote common and specific interests of communities. However, the existing community detection methods ignore such a hierarchy and take all words used to describe node semantics from an identical perspective. This indiscriminate use of words leads to natural defects in depicting networked content in which the deep semantics is not fully utilized. To address this problem, we propose a novel probabilistic generative model. By distinguishing the general and specialized topics of words, our model not only can find community structures more accurately, but also provide two-level semantic interpretation for each community. We train the model by deriving an efficient inference method under the framework of *variational expectation-maximization*. We provide a case study to show the ability of our algorithm in deep semantic interpretability of communities. The superiority of our algorithm for community detection is further demonstrated in comparison with eight state-of-the-art algorithms on eight real-world networks.

1 Introduction

As a fundamental tool for network analysis, community detection has gained more and more attention in the scientific field. Its primary task of community detection is to identify community structures in networks. Community structures correspond to functional modules composed of nodes. The identification of these modules provides a perspective for understanding and analyzing networks. Community detection can be used in many areas such as targeted advertising, protein network analysis, recommendation system, etc.

In the early stages of community detection, a large number of algorithms emerged, only using topology, such as hierarchical clustering [Girvan and Newman, 2002], metric-based algorithms [Yang and Leskovec, 2015], generative models [Wang *et al.*, 2011; Yang and Leskovec, 2013] and statistical inference [Karrer and Newman, 2011; Zhang *et al.*, 2018]. However, when there is noise in network topology, the results of these methods could be improved. In recent years, researchers started integrating network topology and content, since content is beneficial to compensate for noisy topological information. Furthermore, the integration of content information provides the possibility of finding *semantic* communities, i.e. to offer semantic interpretation for each community. Semantic interpretation usually refers to finding the topics that embody the interests or functions of communities. Several algorithms [Pei *et al.*, 2015; Wang *et al.*, 2016; He *et al.*, 2017] have been proposed to find communities with semantic interpretation by using both topology and contents.

However, most existing community detection algorithms that attempt to use contents to find topic words to explain communities have overlooked an important problem, that is, topics in the generation of contents in real life often do not come from a unique level. By analyzing a large number of texts in social networks, we found that topics contained in each document are hierarchical, and in most cases, with a two-level semantic structure. Here we call the first level the *general topic*, reflecting a high-level area of this document, which often covers several specialized topics. We call the second level the *specialized topic*, i.e. the core thought of this document. That way, a document can be summed up with general topic as well as specialized topic. Take a citation network as an example, in which each paper represents a node and the citation relationships between papers represent links. We select a node in this network, for example a classical network community detection article, i.e. [Girvan and Newman, 2002] and analyze its content. Through the statistics of topic words of this paper, we find that there is an obvious two-level topic structure, as shown in Figure 1. To reflect the difference between these two levels of topic words

[*] Corresponding author.

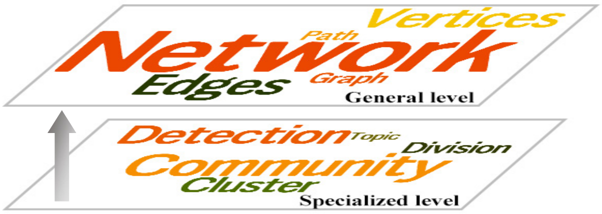


Figure 1: The statistics of two-level topic words in [Girvan and Newman, 2002]. The upper words cloud shows general topic words and the lower shows specialized topic words. Word size represents the frequency of this word in the paper.

more clearly, we divide the partial topic words into two word clouds, to denote general and specialized topic, respectively. As we can see, the general topic words refer to the large area of network analysis. While, the core thought of [Girvan and Newman, 2002] is to provide a new perspective of network analysis, i.e. community detection, which is reflected by the specialized topic words. So, the general and specialized topics work together to form the complete semantics of this paper. Also, a general topic can derive a number of specialized topics though they share different levels of representative words, e.g., "network" is only from the general topic while "community" is only from a specialized topic under this general topic, and they cannot switch.

Unfortunately, the existing methods that integrate network topology and contents ignore the two-level structure of topics. As a consequence, ignoring the hierarchy structure will lead to the inaccurate fitting to the semantic contents. Besides, the topology with noise may not be fully compensated by contents, which affects the ability of algorithms to find communities, as well as the richness of semantic interpretation. Therefore, a deep analysis of the hierarchical structure of content semantics can help not only to better find communities but also get richer explanations of communities.

In order to solve the above problem, we propose a novel probabilistic generative model for jointly identifying communities and their two-level semantics at the same time. Our algorithm can automatically identify that words used to describe attributes are derived from either a general topic or a specialized topic. Due to the full utilization of the hierarchy structure of semantic topics in semantic contents, our model not only can accurately find community structures, but also describe communities using both specialized topics (to denote their particular interests) and general topics (to denote their shared features with similar communities). We finally derive an efficient inference method, based on *variational expectation maximization*, to train the model.

Also of note, if more information is considered, we may have more than two topic levels contained in texts. However, considering too many topic levels will often lead to poor matching between topology and contents. This mismatching issue often occurs in community detection when integrating topology and contents [He *et al.*, 2017; Jin *et al.*, 2018]. Fortunately, two-level topics are often sufficient to express the rich semantic of documents [Xie and Xing, 2013], and also provide a good matching between topology and contents. So, in this paper, to simplify the model's complexity, we only consider two-level topics which is a special case of multi-level cases. The contributions of this work are as follows.

1) We find that topics in the generation of contents in real life are often not from a unique level. We design a novel model, which can fit semantic contents more accurately by distinguishing the general and specialized topic words. The hierarchical use of content enables our model to not only find communities with similar interests but also provide two-level semantic interpretation for each community in the network.

2) We propose a Bayesian treatment on the model, and design an effective inference algorithm based on *variational expectation maximization* to train the model.

3) The superiority of our algorithm in finding community structures is evaluated on eight real-world networks by comparing with eight state-of-the-art algorithms. We also present a case study to show its superiority in two-level semantic interpretability over the existing methods.

2 The Model

We develop a novel probabilistic generative model, Two-Level Semantic Community (TLSC). This model works on undirected and unweighted attributed networks.

An *attributed network* G is represented by n nodes and m attributes. The relation among n nodes can be represented by an adjacency matrix $\mathrm{A} = (a_{ij})_{n\times n}$. If there is an edge between nodes v_i and v_j, $a_{ij} = 1$, and 0 otherwise. $\mathrm{W} = (w_{ik})_{n\times m}$ denotes the attributes matrix, if node v_i has the k^{th} attribute, $w_{ik} = 1$, and 0 otherwise. We assume that the numbers of communities, general and specialized topics are c, E and D, respectively. Each community has a specialized topic to denote the interest itself, while some related communities, which belong to a large area, share a general topic. TLSC aims at fitting observed quantities by adjusting latent quantities and model parameters. Table 1 shows the notations of important parameters and Figure 2 the graphical representation of model.

Sets	Sign	Description
X: Set of observed quantities	A	Adjacency matrix
	W	Attribute matrix
	c	Number of communities
	E, D	Number of general and specialized topics
I : Set of latent quantities	z_i	Community assignment of node v_i
	δ_{ik}	1) $\delta_{ik} = 0$: w_{ik} is generated from general topic; 2) $\delta_{ik} = 1$: w_{ik} is generated from specialized topic
	g_{ik}	General topic assignment of attribute w_{ik}
	s_{ik}	Specialized topic assignment of attribute w_{ik}
Π : Set of model parameters having prior	ω_i	Parameter responsible for generating δ_{ik}
	π_r	Probability that a node belongs to community r
	η_{re}	Probability that v_i is in e^{th} general topic given it belongs to r^{th} community
	f_{ed}	Probability that v_i is in d^{th} specialized topic given it belongs to e^{th} generalized topic
Ξ : Set of parameters with no prior	θ_{rl}	Probability that r^{th} and l^{th} communities connected
	β^{g}_{ek}	Probability that e^{th} general topic generates w_{ik}
	β^{s}_{dk}	Probability that d^{th} specialized topic generates w_{ik}

Table 1: Notations used in the paper

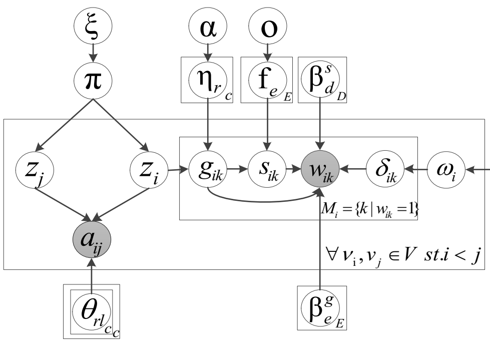


Figure 2: Graphical representation of TLSC

2.1 Generative Process

We first introduce how to generate model parameters (having prior distribution), and then introduce how we use model parameters to generate the latent and observed quantities.

Generating Model Parameters in Set Π having Prior

We propose a Bayesian treatment for the model generation process. Instead of assuming a fixed value for each parameter in set Π, we treat ω, π, H and F as random variables and place a *prior* distribution over them. We then introduce how to generate these parameters in Π based on some hyper-parameters. Hyper-parameters set Ω include ξ, α, o and γ, which are assumed to be given first.

1. We use a Dirichlet distribution to generate model parameter $\pi = (\pi_1, \pi_2, ..., \pi_c)$, where π_r denotes the proportion of nodes belonging to community *r*, and satisfies the constraint $\pi_r \in [0,1]$ and $\sum_{r=1}^{c} \pi_r = 1$. The density function of Dirichlet distribution is defined as:

$$p(\pi \mid \xi) = [\Gamma(\sum\nolimits_{r=1}^{c} \xi_r) / \prod\nolimits_{r=1}^{c} \Gamma(\xi_r)] \prod\nolimits_{r=1}^{c} \pi_r^{\xi_r - 1}, \quad (1)$$

where $\Gamma(\cdot)$ is a Gamma function. The distribution is parameterized by a positive real *c*-dimensional vector $\xi = (\xi_1, \xi_2, ..., \xi_c)$. We refer to ξ as hyper-parameter.

2. We use a Dirichlet distribution to generate parameter matrix $H = (\eta_{re})_{c \times E}$, in which each row η_r is the general topics distribution over community *r*. H can be also viewed as the matrix of probabilistic transition from communities to general topics, and satisfies $\eta_{re} \in [0,1]$ and $\sum_{e=1}^{e=E} \eta_{re} = 1$. The density function is defined as:

$$p(H \mid \alpha) = \prod_{r=1}^{c} p(\eta_r \mid \alpha) = \prod_{r=1}^{c} [\Gamma(\sum_{e=1}^{E} \alpha_e) / \prod_{e=1}^{E} \Gamma(\alpha_e)] \prod_{e=1}^{E} {\eta_{re}}^{\alpha_e - 1}. \quad (2)$$

The hyper-parameter is an *E*-dimensional vector $\alpha = (\alpha_1, \alpha_2, ..., \alpha_E)$. All communities share the same α.

3. Similar to H, we use a Dirichlet distribution to generate matrix $F = (f_{ed})_{E \times D}$. F is a matrix of probabilistic transition from general topics to specialized topics, in which each row f_e is the specialized topics distribution over general topic *e*, subject to $f_{ed} \in [0,1]$ and $\sum_{d=1}^{d=D} f_{ed} = 1$. The hyper-parameter used to generate f_{ed} is a *D*-dimensional vector $o = (o_1, o_2 ..., o_D)$, defined as:

$$p(F \mid o) = \prod_{e=1}^{E} p(f_e \mid o) = \prod_{e=1}^{E} [\Gamma(\sum_{d=1}^{D} o_d) / \prod_{d=1}^{D} \Gamma(o_d)] \prod_{d=1}^{D} {f_{ed}}^{o_d - 1}. \quad (3)$$

Vector o is shared by all general topics.

4. We use Beta distribution to generate model parameter $\omega = (\omega_1, \omega_2, ..., \omega_n)$, in which ω_i is the parameter of Bernoulli distribution. Through the Bernoulli distribution, we can get the value of δ_{ik}, 0 or 1. There are two hyper-parameters in the Beta distribution, i.e. γ_0 and γ_1.

$$p(\omega \mid \gamma) = \prod_{i=1}^{n} p(\omega_i \mid \gamma_{0,} \gamma_1) = \prod_{i=1}^{n} \frac{\Gamma(\gamma_0 + \gamma_1)}{\Gamma(\gamma_0)\Gamma(\gamma_1)} {\omega_i}^{\gamma_0 - 1} (1 - \omega_i)^{\gamma_1 - 1}. \quad (4)$$

These two hyper-parameters are shared by all nodes.

Generating Observed and Latent Quantities in sets X & I

After the model parameters (having prior) have been generated, we then use model parameters to generate observed and latent quantities. The part is critical in the generation process.

1. We first sample the community label z_i of each node v_i from a multinomial distribution independently. The multinomial distribution is defined as:

$$p(z_i = r \mid \pi) = \pi_r, r = 1, 2, ..., c. \quad (5)$$

2. Given the community labels z_i and z_j of nodes v_i and v_j, respectively, we sample indicator a_{ij} from a Bernoulli distribution, defined as:

$$p(A \mid \Theta, z) = \prod_{i<j} p(a_{ij} \mid d_i d_j \theta_{z_i z_j}) = \prod_{i<j} (d_i d_j \theta_{z_i z_j})^{a_{ij}} (1 - d_i d_j \theta_{z_i z_j})^{1 - a_{ij}}. \quad (6)$$

This describes the fitting of the model to network topology from the degree-corrected stochastic block model [Karrer and Newman, 2011], where $\Theta = (\theta_{rl})_{c \times c}$ is the block matrix, and d_i is the degree of v_i. This model performs well in fitting network topology.

3. Given the community label z_i of node v_i, we need to sample the general topic label g_{ik} of attribute w_{ik} of node v_i from a multinomial distribution, defined as:

$$p(G \mid H, z) = \prod_{i=1}^{n} \prod_{k=1}^{m} p(g_{ik} \mid \eta_{z_i})^{w_{ik}} = \prod_{i=1}^{n} \prod_{k=1}^{m} (\eta_{z_i, g_{ik}})^{w_{ik}}. \quad (7)$$

The meaning of H has been explained in last subsection.

4. Then, in order to determine whether each attribute w_{ik} of node v_i is generated from a general topic or a specialized topic, we utilize a binary variable δ_{ik} from a Bernoulli distribution parameterized by ω_i, defined as:

$$p(\Delta \mid \omega) = \prod_{i=1}^{n} \prod_{k=1}^{m} p(\delta_{ik} \mid \omega_i) = \prod_{i=1}^{n} \prod_{k=1}^{m} {\omega_i}^{\delta_{ik}} (1 - \omega_i)^{1 - \delta_{ik}}. \quad (8)$$

The value of δ_{ik} determines the generative process in the next step. That is:

1) If $\delta_{ik} = 0$, w_{ik} will be generated by the general topic. Recall that, in step 3 of this subsection we have identified this general topic. So here we need to generate the attribute w_{ik} of node v_i. To be specific, we sample each attribute from a multinomial distribution, defined as:

$$p(W \mid B^g, G) = \prod_{i=1}^{n} \prod_{k=1}^{m} p(w_{ik} \mid \beta^g_{g_{ik}})^{w_{ik}(1 - \delta_{ik})} = \prod_{i=1}^{n} \prod_{k=1}^{m} (\beta^g_{g_{ik}, k})^{w_{ik}(1 - \delta_{ik})}. \quad (9)$$

In $B^g = (\beta^g_{ek})_{E \times m}$, $\beta^g_{ek} = p(w_{ik} = 1 \mid g_{ik} = e)$ denotes the probability that the e^{th} general topic generates the k^{th} attribute, which is independent of node v_i subject to $\sum_{k=1}^{m} \beta^g_{g_{ik}, k} = 1$ and $\beta^g_{g_{ik}, k} \in [0,1]$.

2) If $\delta_{ik} = 1$, w_{ik} will be generated by a specialized topic, given the general topic label g_{ik} of the attribute w_{ik}. So first we need to sample the specialized topic label from a multinomial distribution, which is defined as:

$$p(\mathrm{S}\mid\mathrm{F},\mathrm{G})=\prod_{i=1}^{n}\prod_{k=1}^{m}p(s_{ik}\mid \mathrm{f}_{g_{ik}})^{w_{ik}\delta_{ik}}=\prod_{i=1}^{n}\prod_{k=1}^{m}(f_{g_{ik},s_{ik}})^{w_{ik}\delta_{ik}}. \tag{10}$$

The specific meaning of F has been explained in step 3 of the last subsection. We then generate the attribute w_{ik} of node v_i from a multinomial distribution, defined as:

$$p(\mathrm{W}\mid\mathrm{B}^{s},\mathrm{S})=\prod_{i=1}^{n}\prod_{k=1}^{m}p(w_{ik}\mid \beta^{s}_{s_{ik}})^{w_{ik}\delta_{ik}}=\prod_{i=1}^{n}\prod_{k=1}^{m}(\beta^{s}_{s_{ik},k})^{w_{ik}\delta_{ik}}. \tag{11}$$

In $\mathrm{B}^{s}=(\beta^{s}_{dk})_{D\times m}$, $\beta^{s}_{dk}=p(w_{ik}=1\mid s_{ik}=d)$ denotes the probability that the d^{th} specialized topic generates the k^{th} attribute of node v_i, subject to $\sum_{k=1}^{m}\beta^{s}_{s_{ik},k}=1$ and $\beta^{s}_{s_{ik},k}\in[0,1]$.

Here the choice of Dirichlet and Beta distribution as priors for π, H, F and ω are not arbitrary. The Dirichlet and Beta distribution are conjugate priors of multinomial and Bernoulli distribution, respectively. It will give a closed-form expression for the posterior and provide mathematical convenience when we derive inference.

2.2 Model Formulation

We note that the generative process implicitly makes a number of conditional independence assumptions among all parameters. In Figure 2, each rectangle denotes the repetition of the enclosed structure, where the number of repetitions is indicated by the subscript. The set of conditional independence assumptions can be readily read off the graph. A node is independent of all its non-descendants given its parent nodes.

Given the hyper-parameters and the model parameters without prior, we decompose the joint distribution over other parameters using the probability chain rule and apply the conditional independence assumptions, as follows:

$$p(\mathrm{A},\mathrm{W},\mathrm{z},\Delta,\mathrm{G},\mathrm{S},\pi,\mathrm{H},\mathrm{F},\omega\mid\Theta,\mathrm{B}^{g},\mathrm{B}^{s},\xi,\alpha,\mathrm{o},\gamma)$$
$$=\left(\begin{array}{l}p(\pi\mid\xi)p(\mathrm{H}\mid\alpha)p(\mathrm{F}\mid\mathrm{o})p(\omega\mid\gamma)p(\mathrm{z}\mid\pi)p(\mathrm{A}\mid\Theta,\mathrm{z})\\ \times p(\Delta\mid\omega)p(\mathrm{G}\mid\mathrm{H},z)p(\mathrm{S}\mid\mathrm{F},\mathrm{G})p(\mathrm{W}\mid\mathrm{B}^{g},\mathrm{G},\mathrm{B}^{s},\mathrm{S},\Delta)\end{array}\right). \tag{12}$$

The sub functions in (12) have all been defined in (1) - (11). We abbreviate (12) to $p(\mathrm{X},\mathrm{I},\Pi\mid\Xi,\Omega)$ in the following.

3 Learning the Model

We give an efficient variational expectation-maximization (EM) algorithm to train the model. We first introduce our variational inference process, and then our algorithm.

3.1 Variational Inference

The Bayesian model proposed in the previous section defines a joint distribution $p(\mathrm{X},\mathrm{I},\Pi\mid\Xi,\Omega)$. Based on this model, the problem of clustering observed quantities X = (A, W) can be transformed into a standard probabilistic inference problem, i.e. finding the *maximum a posteriori* (MAP) configuration of the latent quantities conditioning on A and W, the posterior function is then defined as:

$$p(z,\mathrm{G},\mathrm{S},\Delta,\pi,\mathrm{H},\mathrm{F},\omega\mid\mathrm{A},\mathrm{W},\Theta,\mathrm{B}^{g},\mathrm{B}^{s},\alpha,\mathrm{o},\gamma,\xi).$$

For brevity, we abbreviate this to $p(\mathrm{I},\Pi\mid\mathrm{X},\Xi,\Omega)$, where

$$p(\mathrm{I},\Pi\mid\mathrm{X},\Xi,\Omega)=\frac{p(\mathrm{X},\mathrm{I},\Pi\mid\Xi,\Omega)}{\sum_{\mathrm{I}}\iiiint p(\mathrm{X},\mathrm{I},\Pi\mid\Xi,\Omega)d\Pi}. \tag{13}$$

Since the calculation of (13) is intractable, our basic idea is to approximate the posterior by a novel variational distribution function *q*. According to the theory of variational optimization, the variational distribution *q* can be defined as:

$$q(\mathrm{I},\Pi\mid\Xi')\text{=}q(\mathrm{z},\Delta,\mathrm{G},\mathrm{S},\pi,\mathrm{H},\mathrm{F},\omega\mid\tilde{\Phi},\tilde{\mathrm{T}},\tilde{\mathrm{P}},\tilde{\Sigma},\tilde{\xi},\tilde{\mathrm{A}},\tilde{\mathrm{O}},\tilde{\Lambda}) \tag{14}$$
$$=q(\mathrm{z}\mid\tilde{\Phi})q(\Delta\mid\tilde{\mathrm{T}})q(\mathrm{G}\mid\tilde{\mathrm{P}})q(\mathrm{S}\mid\tilde{\Sigma})q(\pi\mid\tilde{\xi})q(\mathrm{H}\mid\tilde{\mathrm{A}})q(\mathrm{F}\mid\tilde{\mathrm{O}})q(\omega\mid\tilde{\Lambda})$$

Here Ξ' is a set of *variational parameters*. The sub distributions in (14) take exactly the similar parametric forms as the sub functions in (1) - (11). The variational parameters in set Ξ' are free to vary, while the hyper-parameters in Ω are fixed throughout the generative process. Three variational parameters need to be emphasized. The first is $\tilde{\Phi}=(\tilde{\varphi}_{ir})_{n\times c}$, where $\tilde{\varphi}_{ir}=p(z_i=r)$ is the posterior of node v_i belonging to community *r*. The second is $\tilde{\mathrm{P}}=(\tilde{\rho}_{ik,e})_{n\times m\times E}$, which denotes the probability of w_{ik} belonging to general topic *e*. The third is $\tilde{\Sigma}\text{=}(\tilde{\sigma}_{ik,d})_{n\times m\times D}$, which denotes the probability that w_{ik} belongs to specialized topic *d* on the premise that $\delta_{ik}=1$.

Our goal is now to find the variational function $q(\mathrm{I},\Pi\mid\Xi')$ closest to the real posterior $p(\mathrm{I},\Pi\mid\mathrm{X},\Xi,\Omega)$. We adopt the Kullback-Leibler (KL) divergence to measure the distance between the variational function and real posterior, defined as

$$KL(q\parallel p)=\sum_{\mathrm{I}}\iiiint q(\mathrm{I},\Pi\mid\Xi')\log\frac{q(\mathrm{I},\Pi\mid\Xi')}{p(\mathrm{I},\Pi\mid\mathrm{X},\Xi,\Omega)}d\Pi.$$

Note that the KL divergence is a function of the variational parameters $\Xi'=\{\tilde{\Phi},\tilde{\mathrm{T}},\tilde{\mathrm{P}},\tilde{\Sigma},\tilde{\xi},\tilde{\mathrm{A}},\tilde{\mathrm{O}},\tilde{\Lambda}\}$ and model parameters $\Xi=\{\Theta,\mathrm{B}^{g},\mathrm{B}^{s}\}$. Our problem is to find the optimal state of these parameters that minimizes this KL divergence. However, since the calculation of the KL divergence involves the real posterior $p(\mathrm{I},\Pi\mid\mathrm{X},\Xi,\Omega)$, this cannot be solved directly. So instead of directly minimizing the KL divergence, we solve an equivalent maximization problem. The objective function of this maximization problem is defined as:

$$\tilde{L}(q)=\sum_{\mathrm{I}}\iiiint q(\mathrm{I},\Pi\mid\Xi')\log\frac{p(\mathrm{I},\Pi,\mathrm{X}\mid\Xi,\Omega)}{q(\mathrm{I},\Pi\mid\Xi')}d\Pi. \tag{15}$$

Since the sum of these two functions is a constant, that is:

$$KL(q\parallel p)+\tilde{L}(q)=\log p(\mathrm{X}).$$

In order to maximize the objective function $\tilde{L}(q)$, we need to take the derivatives of $\tilde{L}(q)$ with respect to the variational parameters $\Xi'=\{\tilde{\Phi},\tilde{\mathrm{T}},\tilde{\mathrm{P}},\tilde{\Sigma},\tilde{\xi},\tilde{\mathrm{A}},\tilde{\mathrm{O}},\tilde{\Lambda}\}$ and model parameters $\Xi=\{\Theta,\mathrm{B}^{g},\mathrm{B}^{s}\}$, and set these derivatives to zeros.

$$\nabla\tilde{L}(q)=(\frac{\partial\tilde{L}}{\partial\tilde{\xi}},,\frac{\partial\tilde{L}}{\partial\tilde{\varphi}},\frac{\partial\tilde{L}}{\partial\tilde{\rho}},\frac{\partial\tilde{L}}{\partial\tilde{\sigma}},\frac{\partial\tilde{L}}{\partial\tilde{\alpha}},\frac{\partial\tilde{L}}{\partial\tilde{o}},\frac{\partial\tilde{L}}{\partial\tilde{\tau}},\frac{\partial\tilde{L}}{\partial\tilde{\lambda}},\frac{\partial\tilde{L}}{\partial\theta},\frac{\partial\tilde{L}}{\partial\beta^{g}},\frac{\partial\tilde{L}}{\partial\beta^{s}})=0.$$

We get the expressions of parameters needing to be updated.

$$\tilde{\xi}_r=\xi_r+\sum_{i=1}^{n}\tilde{\varphi}_{ir} \tag{16}$$

$$\begin{aligned}\tilde{\varphi}_{ir}\propto\exp\{&\sum_{\substack{j=1\\ j\neq i}}^{n}\sum_{l=1}^{c}\tilde{\varphi}_{jl}[a_{ij}\log d_i d_j\theta_{rl}+(1-a_{ij})\log(1-d_i d_j\theta_{rl})]\\ &+[\psi(\tilde{\xi}_r)-\psi(\sum_{r'=1}^{c}\tilde{\xi}_{r'})]+\sum_{k=1}^{m}\sum_{e=1}^{E}w_{ik}\tilde{\rho}_{ik,e}[\psi(\tilde{\alpha}_{re})-\psi(\sum_{e'=1}^{E}\tilde{\alpha}_{re'})]\}\end{aligned} \tag{17}$$

$$\begin{aligned}\tilde{\rho}_{ik,e}\propto\exp\{&\sum_{r=1}^{c}\tilde{\varphi}_{ir}[\psi(\tilde{\alpha}_{re})-\psi(\sum_{e'=1}^{E}\tilde{\alpha}_{re'})]+(1-\tilde{\tau}_{ik})\log\beta^{g}_{e,k}\\ &+\sum_{d=1}^{D}\tilde{\sigma}_{ik,d}\tilde{\tau}_{ik}[(\psi(\tilde{o}_{ed})-\psi(\sum_{d'=1}^{D}\tilde{o}_{ed'})]\}\end{aligned} \tag{18}$$

$$\tilde{\sigma}_{ik,d} \propto \exp\{\sum_{e=1}^{E} \tilde{\tau}_{ik}\tilde{\rho}_{ik,e}[\psi(\tilde{o}_{ed})-\psi(\sum_{d'=1}^{D}\tilde{o}_{ed'})]+\tilde{\tau}_{ik}\log\beta^{s}_{dk}\} \quad (19)$$

$$\tilde{\alpha}_{re} = \alpha_e + \sum\nolimits_{i=1}^{m}\sum\nolimits_{k=1}^{m} w_{ik}\tilde{\varphi}_{ir}\tilde{\rho}_{ik,e} \quad (20)$$

$$\tilde{o}_{ed} = o_d + \sum\nolimits_{i=1}^{n}\sum\nolimits_{k=1}^{m} w_{ik}\tilde{\tau}_{ik}\tilde{\rho}_{ik,e}\tilde{\sigma}_{ik,d} \quad (21)$$

$$\tilde{\lambda}_{i0} = \gamma_0 + \sum_{k=1}^{m} w_{ik}\tilde{\tau}_{ik}\ ,\quad \tilde{\lambda}_{i1} = \gamma_1 + \sum_{k=1}^{m} w_{ik}(1-\tilde{\tau}_{ik}) \quad (22)$$

$$\begin{aligned}\tilde{\tau}_{ik} = \{1+\exp\{\psi(\tilde{\lambda}_{i1})-\psi(\tilde{\lambda}_{i0})+\sum_{e=1}^{E}\tilde{\rho}_{ik,e}\log\beta^{g}_{e,k}\\ -\sum_{e=1}^{E}\sum_{d=1}^{D}\tilde{\rho}_{ik,e}\tilde{\sigma}_{ik,d}[\psi(\tilde{o}_{ed})-\psi(\sum_{d'=1}^{D}\tilde{o}_{ed'})]-\sum_{d=1}^{D}\tilde{\sigma}_{ik,d}\log\beta^{s}_{d,k}\}\}^{-1}\end{aligned} \quad (23)$$

$$\theta_{rl} \sim \sum_{i<j} a_{ij}\tilde{\varphi}_{ir}\tilde{\varphi}_{jl} \quad (24)$$

$$\beta^{g}_{ek} \sim \sum_{i=1}^{n} w_{ik}(1-\tilde{\tau}_{ik})\tilde{\rho}_{ik,e}\ ,\quad \beta^{s}_{dk} \sim \sum_{i=1}^{n} w_{ik}\tilde{\tau}_{ik}\tilde{\sigma}_{ik,d} \quad (25)$$

where $\psi(\cdot)$ is the Digamma function.

3.2 Iterative Optimization Algorithm

The process of TLSC is shown in Alg. 1. We can use: 1) matrix $\tilde{\Phi}$ to derive the final community label of each node, 2) matrix H and F to find the general topic of each community and which specialized topics belong to the same general topic, 3) the product of H and F (i.e. H×F) to find the specialized topic of each community, and 4) B^{g} and B^{s} to find the key words generated by two-level topics, respectively.

Alg. 1: Process of TLSC

```
Input: A, W, $c$, $E$, $D$, a threshold $\varepsilon$, $count_{\max}$
Output: $\tilde{\Phi}, \mathrm{H}, \mathrm{F}, \mathrm{B}^{g}, \mathrm{B}^{s}$
1. Initialize variational and model parameters randomly
2. $count \leftarrow 1$
3. repeat:
    (a) Update $\tilde{\xi}, \tilde{\Phi}, \tilde{\mathrm{P}}, \tilde{\Sigma}, \tilde{\mathrm{A}}, \tilde{\mathrm{O}}, \tilde{\mathrm{T}}, \tilde{\Lambda}, \Theta, \mathrm{B}^{g}, \mathrm{B}^{s}$ via (16) - (25)
    (b) Compute $\tilde{L}(q^{(count)})$
    (c) $count \leftarrow count+1$
  Until $\tilde{L}(q^{(count)}) - \tilde{L}(q^{(count-1)}) < \varepsilon$ or $count > count_{\max}$
3. Compute H and F using the derived $\tilde{\Phi}, \tilde{\mathrm{P}}, \tilde{\Sigma}$
```

4 Experiments

We first use an online music system to assess the interpretabilities of this model. We then evaluate our approach on 8 real networks in comparison with 8 state-of-the-art methods.

4.1 A Case Study

The dataset we use in this case study analysis is the British online music platform *Last.fm* [Cantador, 2016]. The dataset has 1,892 users and 11,946 attributes, the connections between them form a friendship network. Since no ground-truth is known regarding user communities in the network, we set the number of communities and specialized topics to 38 (c = 38, D = 38) as done in [Wang *et al.*, 2016]. We also performed some experiments to vary the number of general topics, and found that highly overlapping general topics will appear when the number of general topics is greater than 4. So we set this number to 4 (E = 4). We found four groups of communities under different general topics. The two-level semantics of these four groups of communities are shown in Figures 3, 4, 5 and 6, respectively. Due to space limitation, we only show some of the communities in each group.

Figure 3 shows a group of communities of electronic music lovers. These words in general topic #1 can be used to describe all types of electronic music. The communities sharing this general topic are fans of different branches of electronic music. Community #16 is a group of "high techno" music lovers, and the highest probability word in this community is "techno". Community #33 is a group of fans of "dubstep" music, and the origin of dubstep is related to "post-punk". "New wave" is also a branch of electronic music, shown as community #29. Community #27 gathers the "lounge" music fans, and this form of music is also called "chill-out". "Trance" fans gather in community #19.

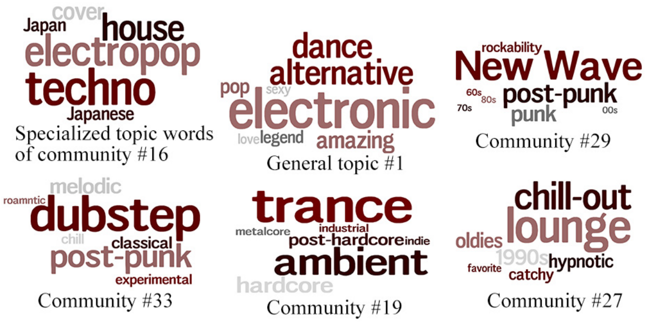


Figure 3: The first group of communities corresponding to general topic #1. The word cloud in the top center shows key words of general topic #1. The surrounding five word clouds show specialized topic words of communities #16, #33, #29, #27 and #19, respectively. Word sizes are proportional to the probability that they belong to a general topic or specialized topic.

Communities in Figure 4 are all related to rock music. Words of general topic #2 reflect detailed keywords of rock music. Specially, community #1 gathers "heavy-metal" music lovers. "Punk" fans gather in community #30. "Progressive-rock" fans are in community #6. Community #12 is a group of "alternative-rock" fans.

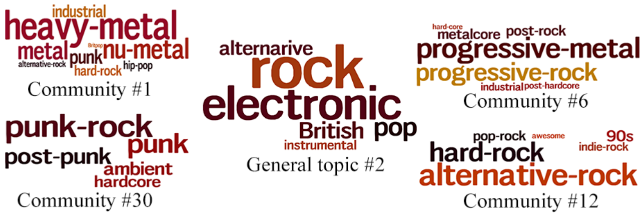
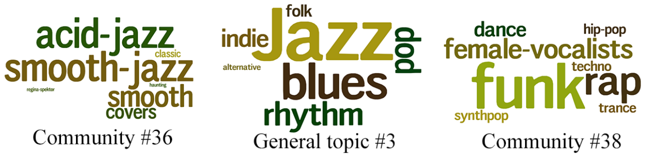


Figure 4: The second group of communities corresponding to general topic #2. The word cloud in the center shows key words of general topic #2. The surrounding word clouds show the specialized topic of communities #1, #30, #6 and #12, respectively.

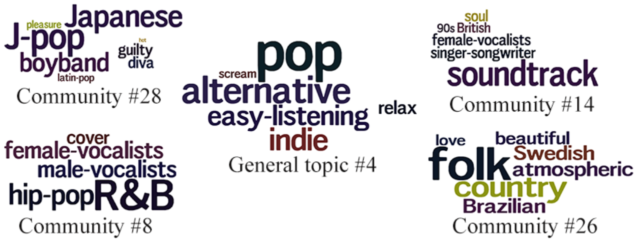
According to the words of general topic #3 in Figure 5, we can realize that community #36 and #38 both belong to "jazz" music. The "acid-jazz" and "smooth-jazz" shown by the specialized topic words of community #36 both are fusion jazz. Community #38 gathers lovers of "funk".



Community #36 General topic #3 Community #38

Figure 5: The third group of communities related to general topic #3. The word cloud in the center shows key words of general topic #3. The surrounding two word clouds denote specialized topic words of communities #36 and #38, respectively.

Words of general topic #4 in Figure 6 show that this is a group of pop music enthusiasts. Specialized topic of community #28 is about Japanese pop, generally called "J-pop". Community #8 gathers fans of "R&B" and "hip-pop". "Soundtrack" and "Folk" lovers gather in community #14 and #26, respectively.



Community #28 General topic #4 Community #14 Community #8 Community #26

Figure 6: The fourth group of communities related to general topic #4. The word cloud in the center shows key words of general topic #4. The surrounding four word clouds are specialized topic words of communities #28, #8, #14 and #26, respectively.

In summary, this case study validates that our model can find related communities with similar interests, describing their shared characteristics with general topic words and the distinct interest tendencies of each community with specialized topic. We can get more appropriate explanations for communities by differentiating these two-level topics clearly.

4.2 Quality Evaluation on Real-World Networks

Here we evaluate the performance of TLSC for detecting communities on eight real-world networks with ground truth of communities, as shown in Table 2. We consider three types of existing community detection methods for comparison. The first type uses network topology alone, including DCSBM [Karrer and Newman, 2011] and BigCLAM [Yang and Leskovec, 2013]. The second, including SMR [Hu *et al.*, 2014], exploits only content information, i.e. node attributes. The last type uses information of network topology and node contents together, including PCL-DC [Yang *et al.*, 2009], Block-LDA [Balasubramanyan and Cohen, 2011], CESNA [Yang *et al.*, 2013], DCM [Pool *et al.*, 2014] and SCI [Wang *et al.*, 2016]. All the above methods require the number of communities to be pre-specified, as well as our method. We set it to the same value that in the ground truth. In our algorithm, we set the number of specialized topics and general topics to 1 and 1/2 of the number of communities.

We used 4 well-known metrics in community detection for evaluation. Accuracy (AC) and normalized mutual information (NMI) [Liu *et al.*, 2012] are the first group of evaluation metrics. But some baselines aim at finding overlapping communities which cannot be evaluated using AC and NMI. We also adopt a new group of evaluation metrics, which includes F-score and Jaccard similarity [Yang *et al.*, 2013].

Datasets	n	e	m	c	Descriptions [Leskovec, 2016]
Texas	187	328	1,703	5	The WebKB network consists of four subnetworks from four
Cornell	195	304	1,703	5	American universities, which are Texas, Cornell, Washington
Washington	230	446	1,703	5	and Wisconsin, respectively.
Wisconsin	265	530	1,703	5	
Twitter	171	796	578	7	Largest subnetwork (id 629863) in Twitter data
Cite	3,312	4,732	3,703	6	A Citeseer citation network
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: Datasets used. n is the number of nodes, e the number of edges, m the number of attributes, and c the number of communities.

Metrics	Methods		Datasets							
(%)	Type	Name	Texas	Cornell	Washington	Wisconsin	Twitter	Cora	Cite	Pubmed
AC	Topo	DCSBM	48.09	37.95	31.80	32.82	60.49	26.57	38.48	53.64
	Cont	SMR	47.54	31.79	49.77	40.84	38.27	30.28	30.87	39.95
	Both	Block-LDA	54.10	46.15	39.17	49.62	35.80	24.35	25.52	49.01
	Both	PCL-DC	38.80	30.26	29.95	30.15	56.79	24.85	34.08	**63.55**
	Both	SCI	62.30	45.64	51.15	**50.38**	50.62	27.98	**40.62**	*N/A*
	Both	TLSC	**65.02**	**47.69**	**51.61**	49.23	**62.87**	**47.62**	35.74	61.38
NMI	Topo	DCSBM	16.65	9.69	9.87	3.14	**57.48**	4.13	17.07	12.28
	Cont	SMR	3.55	8.45	7.3	7.21	3.26	1.18	13.28	0.0367
	Both	Block-LDA	4.21	6.81	3.69	10.09	0	2.42	1.41	6.58
	Both	PCL-DC	10.37	7.23	5.66	5.01	52.64	2.99	17.54	**26.84**
	Both	SCI	17.84	11.44	12.37	**17.03**	43.00	4.87	19.26	*N/A*
	Both	TLSC	**23.92**	**13.61**	**17.63**	16.65	49.14	**33.20**	**23.16**	19.63

Table 3: Comparison of algorithms with disjoint community structures in terms of AC and NMI. "Topo" and "Cont" denote methods using topology and contents, separately; "Both" denotes methods using topology and contents together. Best results are in bold.

Metrics	Methods		Datasets							
(%)	Type	Name	Texas	Cornell	Washington	Wisconsin	Twitter	Cora	Cite	Pubmed
F-score	Topo	BigCLAM	20.64	13.23	13.35	12.84	39.79	9.30	18.89	7.72
	Both	CESNA	23.54	23.48	21.91	23.17	43.82	3.38	31.05	27.97
	Both	DCM	11.15	14.38	12.45	10.45	10.57	2.50	3.43	0.38
	Both	TLSC	**46.92**	**38.47**	**38.67**	**49.23**	**49.85**	**45.74**	**31.08**	**38.85**
Jaccard	Topo	BigCLAM	12.18	7.18	7.25	7.01	26.13	5.01	10.89	4.04
	Both	CESNA	13.57	13.47	12.40	13.14	29.63	1.73	19.10	16.26
	Both	DCM	6.03	7.95	6.72	5.54	5.75	1.27	1.76	0.19
	Both	TLSC	**34.38**	**25.14**	**26.10**	**28.75**	**36.65**	**31.14**	**18.80**	**29.15**

Table 4: Comparison of algorithms with overlapping community structure in terms of F-score and Jaccard.

The experimental results are shown in Tables 3 and 4 using different types of metrics. TLSC basically almost always surpasses state-of-the-art. We can conclude that the hierarchical use of contents indeed improves the accuracy of community detection. This may be because contents intrinsically have more than one level (we used two levels here) and we use more information in this new model.

5 Conclusion

In this paper, we have proposed a probabilistic generative model on attributed networks, which aims at finding community structures as well as the two-level semantics description for communities. This novel Bayesian model is trained through the variational expectation-maximization framework. Our model can 1) automatically recognize that words describing the attributes are either from general or specialized topics; 2) find the general topic and specialized topic for each community, and find which communities related to the same area share a common general topic; and 3) if some topology information is missing or noisy, the full use of semantics can further help finding more accurate community structures. A case study shows the good performance of TLSC for semantic interpretation of communities; comparison experiments on community detection demonstrate the superiority of TLSC in finding community structures.

Acknowledgments

This work was supported by National Key R&D Program of China (No. 2017YFC0820106), Natural Science Foundation of China (No. 61772361, 61502334, 61402070, 61702435) and Hong Kong General Research Fund (GRF) Project No. HKBU 1220091.

References

[Balasubramanyan and Cohen, 2011] Ramnath Balasubramanyan and Willim W.Cohen. Block-LDA: Jointly modeling entity annotated text and entity-entity links. In *Proceedings of 11th SIAM International Conference on Data Mining,* pages 450-461, Philly, USA, Feb 2011.

[Cantador, 2016] Cantador, I. *The 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, 2016. http://ir.ii.uam.es/hetrec2011.

[Girvan and Newman, 2002] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. In *Proceeding of the National Academy of Science,* 99(12):7821-7826, April 2002.

[Jin *et al*., 2018] Di Jin, Xiaobao Wang, Ruifang He, Dongxiao He, Jianwu Dang and Weixiong Zhang. Robust Detection of Link Communities in Large Social Networks by Exploiting Link Semantics. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 314-321, Louisiana, USA, February 2018.

[He *et al*., 2017] Dongxiao He, Zhiyong Feng, Di Jin, Xiaobao Wang and Weixiong Zhang. Joint Identification of Network Communities and Semantics via Integrative Modeling of Network Topologies and Node Content. In *Proceedings of the 31th AAAI conference on Artificial Intelligence*, pages 116-124, California, February 2017.

[Hu *et al*., 2014] Han Hu, Zhouchen Lin, Jianjiang Feng and Jie Zhou. Smooth representation clustering. In *Proceedings of the 27th Conference on CVPR,* pages 3834-3841, Piscataway, New Jersey, USA, June 2014.

[Karrer and Newman, 2011] Brian Karrer and Mark EJ Newman. Stochastic block models and community structure in networks. *Physical Review E,* 83(1): 016107.

[Leskovec, 2016] Leskovec Jure. Stanford Network Analysis Project. http://snap.standford.edu.

[Liu *et al*., 2012] Haifeng Liu, Zhaohui Wu, Xuelong Li, Deng Cai and Thomas S. Huang. Constrained nonnegative matrix factorization for image representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7): 1299-1311, July 2012.

[Pei *et al*., 2015] Yulong Pei, Nilanjan Chakraborty and Katia 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*, pages 2083-2089, San Francisco, California, USA, July 2015.

[Pool *et al*., 2014] Simon Pool, Francesco Bonchi and Matthijs Leeuwen. Description-driven community detection. *Intelligent Systems and Technology,* 5(2):28, 2014.

[Wang *et al*., 2011] Fei Wang, Tao Li, Xin Wang, Shenghuo Zhu and Chris Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery,* 22(3): 493-521, 2011.

[Wang *et al*., 2016] Xiao Wang, Di Jin, Xiaochun Cao, Liang Yang and Weixiong Zhang. Semantic community identification in large attribute networks. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence,* pages 172-178, Palo Alto, California, USA, February 2016.

[Xie and Xing, 2013] Pengtao Xie and Eric P. Xing. Integrating document clustering and topic modeling. ArXiv:1309.687, (2013):694-703, 2013.

[Yang and Leskovec, 2013] Jaewon Yang and Jure 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*, pages 587-596, Rome, Italy, February 2013.

[Yang and Leskovec, 2015] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems,* 42(1):181-213, 2015.

[Yang *et al*., 2009] Tianbao Yang, Rong Jin, Yun Chi and Shengguo Zhu. Combining link and content for community detection: A discriminative approach. In *Proceedings of the 15th ACM SIGKDD International Conference,* pages 927-936, Paris, June 2009.

[Yang *et al*., 2013] Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *Proceedings of the 13th IEEE International Conference on Data Mining,* pages 1151-1156, Piscataway, New Jersey, USA, December 2013.

[Zhang *et al*., 2018] Ziwei Zhang, Peng Cui, Jian Pei, Xiao Wang and Wenwu Zhu. TIMERS: Error- Bounded SVD Restart on Dynamic Networks. In *Proceedings of the 32th AAAI conference on Artificial Intelligence,* pages 224-231, Louisiana, USA, February 2018.