Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Bayesian Approach to Modeling and Detecting Communities in Signed Network

Bo Yang, Xuehua Zhao, and **Xueyan Liu**
School of Computer Science and Technology, Jilin University, Changchun, China
Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, China
ybo@jlu.edu.cn

Abstract

There has been an increasing interest in exploring signed networks with positive and negative links in that they contain more information than unsigned networks. As fundamental problems of signed network analysis, community detection and sign (or attitude) prediction are still primary challenges. To address them, we propose a generative Bayesian approach, in which 1) a signed stochastic blockmodel is proposed to characterize the community structure in context of signed networks, by means of explicitly formulating the distributions of both density and frustration of signed links from a stochastic perspective, and 2) a model learning algorithm is proposed by theoretically deriving a variational Bayes EM for parameter estimation and a variation based approximate evidence for model selection. Through the comparisons with state-of-the-art methods on synthetic and real-world networks, the proposed approach shows its superiority in both community detection and sign prediction for exploratory networks.

Introduction

In recent years, the study of signed networks becomes a burgeoning research area. In contrast to the extensively studied unsigned networks only encoding whether relationships exist or not, signed networks contain more information by extending the single relationship to positive and negative relationships, wherein positive ones represent to like, trust, support or collaborate and negative ones represent to dislike, distrust, oppose or compete, among others. For signed networks, community detection is of considerable importance for understanding the basic patterns of structure and dynamics. This task is trying to identify K antagonistic communities, so that most links within communities are positive while most links between communities are negative. In this sense, communities are consistent with the clusters defined in balance theory in social science (Cartwright and Harary 1956; Davis 1967), where a strongly (or weakly) balanced network can be divided into two (or K) clusters, so that all links within clusters are positive and all links between clusters are negative. Note that, real-world signed networks are usually unbalanced due to the frustration in them, i.e., negative links within clusters and positive links between clusters.

Although many methods have been proposed to address community detection since Girvan and Newman's work (Girvan and Newman 2002), however, most of them are exclusively designed for unsigned networks, which focus on link density rather than link sign to define and detect communities.Therefore, the primary techniques adopted by them cannot be directly applied to signed networks, such as modularity optimization (Newman 2004), Markov random walk (Zhou 2003), clique percolation model (Palla et al. 2005), spectral analysis (Mitrović and Tadić 2009), evolutionary optimization (Pizzuti 2008), among many others.

In view of this, new methods have been proposed for signed community detection. On one hand, some of them are studied from the perspective of social science. For instance, based on the social balance theory, Doreiian and Mrvar proposed a frustration-optimization based method, referred to as DM, which partitions a signed network by minimizing the sum of negative link quantity within communities and positive link quantity between communities (Doreian and Mrvar 1996). Thereafter, Larusso et al improved the same idea to partition weighted signed networks (Larusso, Bogdanov, and Singh 2010). Very similarly, Bansal et al proposed a correlation clustering method to maximize the agreement (i.e. the number of positive intra-cluster links and negative inter-cluster links) or to minimize the disagreement (i.e. the number of negative intra-cluster links and positive inter-cluster links) among nodes (Bansal, Blum, and Chawla 2004).

On the other hand, some of them are proposed by means of generalizing the current techniques of partitioning unsigned networks as mentioned above. For examples, based on potts model Traag et al deduced an improved modularity function for signed networks and then proposed a modularity-optimization based partition algorithm PSA (Traag and Bruggeman 2009). Yang et al generalized the Markov stochastic process on unsigned network to signed network and then proposed an improved random-walk based method FEC (Yang, Cheung, and Liu 2007). Huang et al improved the clique percoltion model to detect overlapping signed communities (Huang and Qiu 2010). Anchuri et al proposed a generalized spectral method for signed network partition (Anchuri and Magdon-Ismail 2012). Very recently, multi-objective evolutionary methods have been applied to

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

signed network decomposition, by simultaneously optimizing two objectives defined in terms of not only link density but also link sign, e.g. internal similarity versus external similarity (Liu, Liu, and Jiang 2014) and kernel k-means versus ratio-cut (Gong et al. 2014).

All the aforementioned methods can be seemed as discriminative, which just focused on looking for a way to distinguish notes by clustering them into different groups, based on either predefined optimization objectives (such as modularity) or heuristics (such as random walk model). However, they are not concerned with how the real-world signed networks containing community structures are generated. Distinctly, in this work we plan to propose a generative approach. Compared with discriminative methods, generative methods are more expected because they can be applied to not only community detection but network modeling, generation, as well as link and attitude prediction.

Being one important statistical network model, stochastic blockmodel (SBM) is a good generative model. As it enables us to reasonably decompose and then properly analyze an exploratory network without a priori knowledge about its intrinsic structure, SBM has attracted more and more attention since it was originally proposed by Holland and Leinhardt (Holland and Leinhardt 1981). Although various extensions of SBM have been proposed to address different tasks of network analysis, such as multiple role SBM (Airoldi et al. 2008), overlapping SBM (Latouche et al. 2011), dynamical SBM (Yang et al. 2011) and hierarchical SBM (Yang, Liu, and Liu 2012), however, to the best of our knowledge, all the existing SBMs are designed for unsigned networks and thereby incompetent for handling signed networks. In view of this, we are motivated to propose a novel generative Bayesian approach. More specifically, our main contributions are two-fold:

(1) We proposed a signed stochastic blockmodel to characterize and generate the block structures of signed networks by means of explicitly formulating both link density and link sign from a stochastic perspective. (2) We proposed an effective algorithm for learning this model from exploratory networks based on variational Bayes techniques, which can automatically detect block numbers and assignments.

Signed Stochastic Blockmodel

Let A denote the adjacency matrix of a signed network N containing n nodes. a_{ij} is equal to 1 or -1 if node i is connected to node j by a positive or negative link. Otherwise, a_{ij} will be zero. The signed stochastic blockmodel (SSBM for short) of N is defined as a 4-tuple:

$$X = (K, \Pi, \Theta, \Omega) \tag{1}$$

K is the number of blocks. Ω is a K-dimension vector, wherein ω_q denotes the prior probability that a node is assigned to block q. $\Pi = (\pi_1, \pi_{-1}, \pi_0)$ is a 3-dimension vector, in which each component denotes the probability that there is a positive link, negative link, or no link between a pair of nodes within the same block, respective. Similarly, we define $\Theta = (\theta_1, \theta_{-1}, \theta_0)$, in which each component denotes the probability that there is a positive link, negative link, or no link between a pair of nodes belonging to different blocks.

Given a signed network, one can deduce a latent $n \times K$ matrix Z, indicating the relationship between node and block assignment. $z_{ik} = 1$ if node i is assigned to block k, otherwise $z_{ik} = 0$. Moreover, z_i follows the following multinomial distribution with a parameter Ω:

$$z_i \sim M(1, \Omega = \{\omega_1, \omega_2, ..., \omega_K\})$$

Given Z, a_{ij} follows the following multinomial distribution with parameters Π and Θ:

$$a_{ij} \sim M(1, \Pi = \{\pi_1, \pi_{-1}, \pi_0\}), z_{iq} z_{jl} = 1 \, and \, q = l$$
$$a_{ij} \sim M(1, \Theta = \{\theta_1, \theta_{-1}, \theta_0\}), z_{iq} z_{jl} = 1 \, and \, q \neq l$$

According to SSBM, one can generate a synthetic signed network with a block structure by following steps:

1) assign nodes to blocks according to Ω.

2) generate positive and negative links between nodes within the same blocks according to Π.

3) generate positive and negative links between nodes belonging to different blocks according to Θ.

Accordingly, we have proofed that the log-likelihood of complete data is as follows:

$$\begin{aligned} \log p(N, Z|K) = & \sum_{i=1}^{n} \sum_{q=1}^{K} z_{iq} \log \omega_q + \sum_{i<j} \sum_{q,l} (z_{iq} z_{jl} \times \\ & \log M(a_{ij}; \Pi) + (1 - z_{iq} z_{jl}) \log M(a_{ij}; \Theta)) \end{aligned} \tag{2}$$

We now describe the aforementioned SSBM in a full Bayesian framework. In the framework, we need specify the priors for the model parameters (Π, Θ, Ω). Since $p(z_i|\Omega), p(a_{ij}|Z, \Pi)$ and $p(a_{ij}|Z, \Theta)$ satisfy multinomial distribution, respectively, we select Dirichlet distribution as their conjugate prior distributions, as follows:

$$p(\Omega|\boldsymbol{\rho}^0 = \{\rho_1^0, ..., \rho_K^0\}) = \text{Dir}(\Omega; \boldsymbol{\rho}^0)$$
$$p(\Pi|\boldsymbol{\eta}^0 = \{\eta_1^0, \eta_{-1}^0, \eta_0^0\}) = \text{Dir}(\Pi; \boldsymbol{\eta}^0)$$
$$p(\Theta|\boldsymbol{\mu}^0 = \{\mu_1^0, \mu_{-1}^0, \mu_0^0\}) = \text{Dir}(\Theta; \boldsymbol{\mu}^0)$$

where $\forall q$:ρ_q^0, $\forall h$: η_h^0, $\forall h$:μ_h^0 are hyperparameters, which are interpreted as an effective pseudo-occupations of respective blocks in the prior, pseudo-observations of three types of links (positive, negative, no-link) within or between blocks in the prior, respectively. In other words, in the full Bayesian framework, parameters Π, Θ, and Ω are regarded as random variables, the distributions of which depend on their respective hyperparameters.

Being a generation of standard SBM (Snijders and Nowicki 1997), SSBM is much more flexible and it is able to depict more structural patterns of unsigned or signed networks, as defined in terms of either link density or link sign or both of them, from a stochastic perspective. For examples: (1) in the case of $\pi_{-1} = 0$ and $\theta_{-1} = 0$, SSBM is able to characterize either the community structure (when $\pi_1 > \theta_1$) or the multipartite structure (when $\pi_1 < \theta_1$) of unsigned networks in terms of link density; (2) in the case of $\pi_{-1} = 0$ while $\theta_1 = 0$, SSBM is able to characterize a balanced signed network in terms of link sign; (3) in the most general case of $\pi_{-1} \neq 0$ while $\theta_1 \neq 0$, SSBM is able

to characterize the frustration of an unbalanced signed network in terms of both density and sign, in which there are a small fraction of negative links within communities and a small fraction of positive links between communities.

Variational Bayes SSBM Learning

Now we introduce SSBM learning algorithm (SSL for short). SSL adopts a variational Bayes EM algorithm to estimate parameters and an approximate Bayesian model evidence for model selection. We adopt such variational techniques due to two main reasons.

First of all, standard EM algorithm cannot be directly used for SSBM in that the components of latent variable Z are correlated and thus the posterior distribution of Z, under the condition of data and model parameters, cannot be explicitly derived as an input required by standard EM. More specifically, component z_i is correlated to others means that the computation of its posterior distribution $P(z_i|N,\Theta,\Pi,\Omega)$ is recursively dependent on the distribution of z_j for any $j \neq i$. Using variational Bayes EM, one can infer an approximate posterior distribution of Z in terms of estimated superparameters. Note that, in the literature, variational EM has ever been adopted for SBM learning (Daudin, Picard, and Robin 2008). Unlike variational EM based on point estimation, variational Bayes EM infers the distribution of Z based on the distributions of parameters instead of their point estimation values (or maximum likelihood estimation values). Therefore, compared with variational EM, variational Bayes EM is more robust and is expected to infer a better posterior distribution close to the truth from real-world networks usually containing much noise.

In addition, although the Bayesian model evidence of network N (i.e. $\log p(N|K)$) can be obtained by computing the marginal integration of $\log p(N,Z|K)$ (see Eq. 2) over Z, however, this computation involves a summation of K^n terms, which will quickly becomes prohibitively intractable. By taking the model parameters (Ω,Π,Θ) as random variables, bases on variational Bayes techniques one can readily compute a lower bound of the marginal likelihood in terms of their superparamters, as an approximation of true evidence, for model selection.

Superparameter estimation The log-likelihood of N (or the marginal log-likelihood of complete data) can be decomposed into two terms:

$$\log p(N) = \mathcal{L}(q(\cdot)) + \mathrm{KL}(q(\cdot)||p(\cdot|N)) \tag{3}$$

where

$$\begin{aligned}\mathcal{L}(q(\cdot)) = &\sum_{Z}\int\int\int q(Z,\Pi,\Theta,\Omega)\\ &\times\log\left\{\frac{p(N,Z,\Pi,\Theta,\Omega)}{q(Z,\Pi,\Theta,\Omega)}\right\}d_{\Pi}d_{\Theta}d_{\Omega}\end{aligned} \tag{4}$$

$$\begin{aligned}\mathrm{KL}(q(\cdot)||p(\cdot|N)) = -&\sum_{Z}\int\int\int q(Z,\Pi,\Theta,\Omega)\times\\ &\log\left\{\frac{p(Z,\Pi,\Theta,\Omega|N)}{q(Z,\Pi,\Theta,\Omega)}\right\}d_{\Pi}d_{\Theta}d_{\Omega}\end{aligned} \tag{5}$$

In Eqs.3 and 5, KL denotes the Kullback-Leibler divergence between $q(Z,\Pi,\Theta,\Omega)$ and $p(Z,\Pi,\Theta,\Omega|N)$. To minimizing Eq.5 with respect to $q(Z,\Pi,\Theta,\Omega)$ is equivalent to maximizing the lower bound Eq.4. To obtain a computationally tractable algorithm, we use mean field approximation, one of the most popular forms of variational inference, in which we assume the posterior $q(Z,\Pi,\Theta,\Omega)$ is a fully factorized approximation. Formally, we have:

$$q(Z,\Pi,\Theta,\Omega) = q(\Pi)q(\Theta)q(\Omega)\prod_{i=1}^{n}q(z_i) \tag{6}$$

where $q(\Pi)$, $q(\Theta)$, $q(\Omega)$ and $q(z_i)$ denote the posteriors of variables Π, Θ, Ω and Z, respectively, which will be inferred by a variational Bayes EM. Specifically, in its E-step, each distribution $q(z_i)$ is optimized; and in its M-step, $q(\Pi)$, $q(\Theta)$ and $q(\Omega)$ are optimized, respectively.

We first derive the optimal approximation at node i. According to variational Bayes, the optimal posterior $q(z_i)$ is:

$$\begin{aligned}\log q(z_i) =& \mathrm{E}_{Z^{\setminus i},\Pi,\Theta,\Omega}\big[\log p(N,Z,\Pi,\Theta,\Omega)\big] + \text{const}\\ =& \mathrm{E}_{Z^{\setminus i},\Pi,\Theta}\big[\log p(N|Z,\Pi,\Theta)\big]\\ &+ \mathrm{E}_{Z^{\setminus i},\Omega}\big[\log p(Z|\Omega)\big] + \text{const}\\ =&\sum_{q=1}^{K} z_{iq}\Big(\sum_{j\neq i}\big(\tau_{jq}\sum_{h}\delta(a_{ij},h)(\psi(\eta_h)-\psi(\sum_{h}\eta_h))\\ &+\sum_{l\neq q}\tau_{jl}\sum_{h}\delta(a_{ij},h)(\psi(\mu_h)-\psi(\sum_{h}\mu_h))\big)\\ &+(\psi(\rho_q)-\psi(\sum_{k}\rho_k))\Big) + \text{const}\end{aligned} \tag{7}$$

where $Z^{\setminus i}$ denotes Z of all nodes except node i, $\delta(a,h) = 1\cdot I_{\{a=h\}} + 0\cdot I_{\{a\neq h\}}$, and $h \in \{1,-1,0\}$. When $y \sim \mathrm{Dir}(y;a_1,a_2,...,a_K)$, $E_y[\log(y)] = \psi(a_q) - \psi(\sum a_q)$ where $q \in \{1,2,...,K\}$ and $\psi(\cdot)$ is Digamma function. To simplify calculations, the terms that do not depend on Z_i have been absorbed into the constant. After taking the exponential of Eq.7 and normalization, the optimal approximation at node i is the following multinomial distribution:

$$q(z_i) = M(z_i;1,\tau_{i1},...,\tau_{iK}) \tag{8}$$

where τ_{iq} is the probability of node i belonging to block q, and satisfies:

$$\begin{aligned}\tau_{iq} \propto e^{\psi(\rho_q)-\psi(\sum_k\rho_k)}\prod_{j\neq i}^{n}\Big(&e^{\tau_{jq}\sum_h\delta(a_{ij},h)(\psi(\eta_h)-\psi(\sum_h\eta_h))}\\ &\times\prod_{l=1}^{K}e^{\tau_{jl}\sum_h\delta(a_{ij},h)(\psi(\mu_h)-\psi(\sum_h\mu_h))}\Big)\end{aligned} \tag{9}$$

Then, we derive the posteriors $q(\Omega)$, $q(\Pi)$, $q(\Theta)$ by optimizing the lower bound (see Eq.4), respectively. According

to variational Bayes, the optimal distribution $q(\Omega)$ is:

$$\begin{aligned}\log q(\Omega) &= \mathrm{E}_{Z,\Pi,\Theta}\big[\log p(N,Z,\Pi,\Theta,\Omega)\big] + \text{const} \\ &= \mathrm{E}_{Z}\big[\log p(Z|\Omega)\big] + \log p(\Omega) + \text{const} \\ &= \sum_{q=1}^{K}\Big(\rho_q^0 - 1 + \sum_{i=1}^{n}\tau_{iq}\Big)\log\omega_q + \text{const}\end{aligned} \tag{10}$$

After taking the exponential of Eq.10 and normalization, we obtain the optimal approximation of $q(\Omega)$, i.e., a Dirichlet distribution, which is the same form as its prior $p(\Omega)$.

$$q(\Omega) = \mathrm{Dir}(\Omega;\boldsymbol{\rho}), \qquad \rho_q = \rho_q^0 + \sum_{i=1}^{n}\tau_{iq} \tag{11}$$

In the same way, we obtain $q(\Pi)$ and $q(\Theta)$, two Dirichlet distributions, which are the same form as their priors.

$$q(\Pi) = \mathrm{Dir}(\Pi;\boldsymbol{\eta}),\ \eta_h = \eta_h^0 + \sum_{i<j}^{n}\sum_{q=1}^{K}\tau_{iq}\tau_{jq}\delta(a_{ij},h) \tag{12}$$

$$q(\Theta) = \mathrm{Dir}(\Theta;\boldsymbol{\mu}),\ \mu_h = \mu_h^0 + E_h - \sum_{i<j}^{n}\sum_{q=1}^{K}\tau_{iq}\tau_{jq}\delta(a_{ij},h) \tag{13}$$

where E_h denotes the number of the positive, negative and no link in the network, respectively.

Evidence approximation and model selection So far we have derived the approximated posteriors of model parameters and latent variables. However, the problem of automatically determining block number K has not been addressed, which is significant for exploring real-world networks, in that we usually have not a prior knowledge about K. According to Bayesian theory, an optimal model should be the one with the largest evidence. Formally, the evidence of SSBM is $\log P(N|K) = \log\{\sum_Z \int\int\int P(N,Z,\Pi,\Theta,\Omega|K)d_\Pi d_\Theta d_\Omega\}$. Unluckly, the computation of SSBM evidence is intractable in that for each value of K, it involves a multiple integration over all possible values of parameters and latent variables.

To tackle this issue, we plan to approximate the evidence by its lower bound, as suggested by (Hofman and Wiggins 2008). Recall Eq.3, an evidence is the sum of lower bound (Eq.4) with respect to $q(\cdot)$ and KL divergence (Eq.5). After the convergence of minimizing KL divergence by variational Bayes EM, $q(\cdot)$ is expected to be close to true posterior distribution, or in other words, the KL divergence is expected to be much smaller than the lower bound, thereby the evidence can be approximated by its lower bound with a small error, which can be seemed as the model selection criterion of SSBM. The formula of calculating the lower bound in terms of estimated posteriors of latent variables and parameters is derived as follows:

$$\begin{aligned}&\mathcal{L}(q(\cdot)) \\ &= \sum_Z \int\int\int q(Z,\Pi,\Theta,\Omega)\log\left\{\frac{p(N,Z,\Pi,\Theta,\Omega)}{q(Z,\Pi,\Theta,\Omega)}\right\}d_\Pi d_\Theta d_\Omega \\ &= \mathrm{E}_{Z,\Pi,\Theta}\big[\log p(N|Z,\Pi,\Theta)\big] + \mathrm{E}_{Z,\Omega}\big[\log p(Z|\Omega)\big] + \mathrm{E}_{\Pi}\big[\log p(\Pi)\big] \\ &\quad + \mathrm{E}_{\Theta}\big[\log p(\Theta)\big] + \mathrm{E}_{\Omega}\big[\log p(\Omega)\big] - \sum_i^n \mathrm{E}_{z_i}\big[\log q(z_i)\big] \\ &\quad - \mathrm{E}_{\Pi}\big[\log q(\Pi)\big] - \mathrm{E}_{\Theta}\big[\log q(\Theta)\big] - \mathrm{E}_{\Omega}\big[\log q(\Omega)\big] \\ &= \sum_h\Big(\eta_h^0 - \eta_h + \sum_{i<j}^{n}\sum_{q=1}^{K}\tau_{iq}\tau_{jq}\delta(a_{ij},h)\Big)\Big(\psi(\eta_h) - \psi(\sum_h \eta_h)\Big) \\ &\quad + \sum_h\Big(\mu_h^0 - \mu_h + \sum_{i<j}^{n}\sum_{q\neq l}^{K}\tau_{iq}\tau_{jl}\delta(a_{ij},h)\Big)\Big(\psi(\mu_h) - \psi(\sum_h \mu_h)\Big) \\ &\quad + \sum_{q=1}^{K}\Big(\big(\rho_q^0 - \rho_q + \sum_{i=1}^{n}\tau_{iq}\big)\big(\psi(\rho_q) - \psi(\sum_q \rho_q)\big)\Big) \\ &\quad - \sum_{i=1}^{n}\sum_{q=1}^{K}\tau_{iq}\log\tau_{iq} + \log\left\{\frac{\Gamma(\sum_{q=1}^K \rho_q^0)\prod_{q=1}^K\Gamma(\rho_q)}{\Gamma(\sum_{q=1}^K \rho_q)\prod_{q=1}^K\Gamma(\rho_q^0)}\right\} \\ &\quad + \log\left\{\frac{\Gamma(\sum_h \eta_h^0)\prod_h\Gamma(\eta_h)}{\Gamma(\sum_h \eta_h)\prod_h\Gamma(\eta_h^0)}\right\}\left\{\frac{\Gamma(\sum_h \mu_h^0)\prod_h\Gamma(\mu_h)}{\Gamma(\sum_h \mu_h)\prod_h\Gamma(\mu_h^0)}\right\}\end{aligned}$$

According to Eqs.11,12 and 13, the terms $\eta_h^0 - \eta_h + \sum_{i<j}^n\sum_{q=1}^K \tau_{iq}\tau_{iq}\delta(a_{ij},h)$, $\mu_h^0 - \mu_h + \sum_{i<j}^n\sum_{q\neq l}^K \tau_{iq}\tau_{jl}\delta(a_{ij},h)$, and $\rho_q^0 - \rho_q + \sum_{i=1}^n \tau_{iq}$ in the lower bound vanish. So, finally the low bound is:

$$\begin{aligned}\mathcal{L}(q(\cdot)) &= \log\left\{\frac{\Gamma(\sum_{q=1}^K \rho_q^0)\prod_{q=1}^K\Gamma(\rho_q)}{\Gamma(\sum_{q=1}^K \rho_q)\prod_{q=1}^K\Gamma(\rho_q^0)}\right\} \\ &+ \log\left\{\frac{\Gamma(\sum_h \eta_h^0)\prod_h\Gamma(\eta_h)}{\Gamma(\sum_h \eta_h)\prod_h\Gamma(\eta_h^0)}\right\} \\ &+ \log\left\{\frac{\Gamma(\sum_h \mu_h^0)\prod_h\Gamma(\mu_h)}{\Gamma(\sum_h \mu_h)\prod_h\Gamma(\mu_h^0)}\right\} - \sum_{i=1}^{n}\sum_{q=1}^{K}\tau_{iq}\log\tau_{iq}\end{aligned} \tag{14}$$

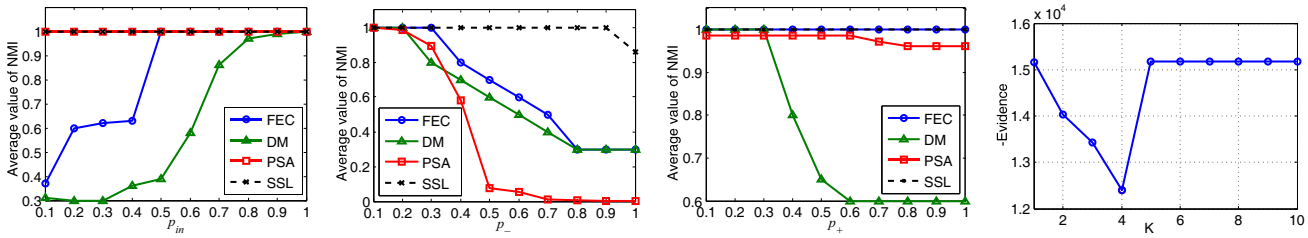
SSL Algorithm In summary, the algorithm of SSBM learning based on variational Bayes approach is given in Table 1, which can automatically detect the block structure of a given signed network. Next, we analyze its time complexity. Updating the posterior of Z by **for** loop in line 07-09 takes $O(Kn^2)$. Updating the posterior of Ω by **for** loop in line 10-11 takes $O(Kn)$. Updating the posteriors of Π and Θ by **for** loop in line 12-14 takes $O(Kn^2)$. Consequently, when K is given, the time of SSL is $O(IKn^2)$, where I denotes the iterations of **repeat** loop until convergence. Calculating the lower bound $\mathcal{L}_K$ in 16 takes $O(Kn)$. So, when K is unknown, the total time of SSL is $O(In^2(K_{max} - K_{min})^2)$.

Validation

In this section, we test the proposed SSBM and SSL toward two main tasks: community detection and sign prediction.

Validation on community detection

In showing the superiority of SSBM and SSL, three representative algorithms for signed community detection are selected to compare. They are the frustration-optimization based DM (Doreian and Mrvar 1996), the random-walk based FEC (Yang, Cheung, and Liu 2007), and the modularity-optimization based PSA (Traag and Bruggeman 2009), respectively. We use both synthetic networks and



(a) (b) (c) (d)

Figure 1: Performance comparisons of community detection.

Table 1: SSL Algorithm

X=SSL(N,K_{min},K_{max})

Input: N, K_{min}, K_{max}
Output: Z
initialize $\mathcal{L}$
for $K = K_{min}$ to K_{max}
  initialize τ, μ, η, ρ;
  repeat
    for $i = 1$ to N // Update the posterior over each z_i
      for $q = 1$ to K
        update τ_{iq} according to Eq. 9;
    for $q = 1$ to K // Update the posterior over Ω
      update ρ_q according to Eq. 11;
    for $h \in \{1, -1, 0\}$ // Update the posterior over Π, Θ
      update η_h according to Eq. 12;
      update μ_h according to Eq. 13;
  until convergence
  update $\mathcal{L}_K$ according to Eq. 14; // Update the lower bound
  if $\mathcal{L}_K > \mathcal{L}$ **then** $\mathcal{L} = \mathcal{L}_K$; $Z_p = \boldsymbol{\tau}$;
calculate Z according to Z_p;

real-world networks to test the four algorithms. Since all test networks contain ground truth community structures, the NMI criterion (Kuncheva and Hadjitodorov 2004) is adopted to quantitatively measure the accuracy of community detections. Intuitively, the larger NMI, the closer to ground truth.

Synthetic signed networks We first use synthetic networks to test. Although the proposed SSBM is a generation model of signed networks, for the sake of fairness, here we choose a widely used model (Yang, Cheung, and Liu 2007) to produce synthetic signed networks, which is defined as:

$$SG(c, n, k, p_{in}, p_-, p_+)$$

where c is the number of communities, n is the number of nodes in each community, k is the average degree of node, p_{in} is the probability of each node connecting other nodes in the same community. p_- and p_+ regulate noise levels, denoting the probabilities of negative links within communities and positive links across communities, respectively.

First, we generate two types of synthetic signed networks: balanced networks and unbalanced networks. For balanced networks, the generation model is $SG(4, 300, 100, p_{in}, 0, 0)$, in which p_{in} increases from 0.1 to 1 stepping by 0.1. For unbalanced networks, two models are used, i.e. $SG(4, 300, 100, 0.8, p_-, 0.2)$ and $SG(4, 300, 100, 0.8, 0.2, p_+)$, in which p_- and p_+ gradually increase from 0 to 1 stepping by 0.1, respectively. The two models are used to test the influence of two types of noise on the performance of community detection. For each model mentioned above, we generate 100 random networks.

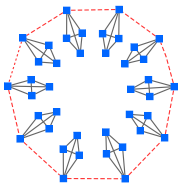
Fig. 1(a) shows the performance of four algorithms on balanced networks. As we can see, SSL and PSA perform the best. For all p_{in}, the detections provided by these two algorithms are exactly the same as ground truth (i.e. NMI=1). Compared with DM and FEC, this result implies a good feature of SSL and PSA. That is, when handling balanced networks, the performance of these two algorithms will be not affected by the link density within communities.

Figs. 1(b) and 1(c) show the performance of four algorithms on unbalanced networks. In Fig. 1(b), p_+ is fixed and the noisy level within communities augments as p_- increasing. As we see, the performance of SSL is significantly better than other three, and the detections provided by it are exactly the same as ground truth except for $p_- > 0.9$. In Fig. 1(c), p_- is fixed and the noisy level outside communities augments as p_+ increasing. In this case, SSL, FEC and PSA, particularly the first two, performs much better than DM. The main reason is, as the fraction of positive links across communities (i.e. p_+) increasing, the signed network being handled gradually turn into an unsigned network, in which community structure are dominated by link density. Compared with DM that focuses on optimizing the frustration of signs, the other three consider not only link sign but also link density when they are partitioning a network, thereby leading to a much better performance in this case.

From these results, one notes that SSL performs the best when handling unbalanced networks contain different types and different levels of noise. The rationale is two told: 1) SSBM explicitly models such noise with parameters such as π_{-1} and θ_1; and 2) SSL adopts variational Bayes to estimate the distributions rather than point values of such parameters.

Fig. 1(d) shows the model selection process of SSL, in which y-axis denotes the minus evidence corresponding to different K. As an example, we just show the interval of K from 1 to 10. As we see, the evidence reaches its biggest value when $K = 4$, exactly the same as the truth.

Modularity-optimization based methods such as PSA will suffer the problem of resolution limitation. That is, such methods tend to detect a small number of bigger commu-



K_{True}	15	16	17	18	19	20	21	22	23	24	25
K_{SSL}	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
K_{PSA}	**15**	**16**	**17**	**18**	**19**	17	14	10	9	10	27

Figure 2: Resolution limitation test.

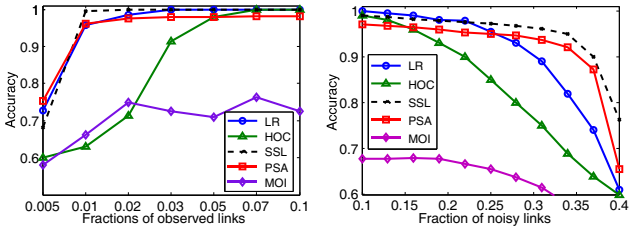
nities. Next, we will test whether the proposed Bayesian approach is able to fix this issue. In this experiment, the network to be tested is similar to the one suggested by (Hofman and Wiggins 2008), which consists of a ring of complete cliques, as shown in Fig. 2. Each clique stands for a community. The links within cliques are positive (see solid lines), and those between cliques are negative (see dotted lines). As the number of cliques in the ring (denoted by K_{true}) increasing, it gets more and more challenge to precisely detect them all. Fig.2 show the performance of SSL and PSA. As we see, SSL performs perfectly in all cases, much better than PSA in the cases of larger K_{true}, although PSA already takes effort to weaken the effect of resolution limitation.

Real-world signed networks We use Slovene parliamentary party network (Kropivnik and Mrvar 1996), Gahuku-Gama subtribes network (Read 1954) and monastery network (Doreian and Mrvar 1996) to further validate SSL. The three real-world networks are chosen because they all have ground truth community structures and thereby have been the benchmarks for testing the performance of signed community detection (Yang, Cheung, and Liu 2007; Doreian and Mrvar 1996). In all cases, the detections of SSL are exactly the same as the ground truth. Note that, before applying SSL to Slovene parliamentary party network, we first turn it into a binary network by setting zero as the threshold of positive and negative links.

Validation on sign (or attitude) prediction

In showing the superiority of SSL for sign prediction, three representative algorithms are selected to compare. They are the balance theory based MOI (Chiang et al. 2011), the supervised learning based HOC (Leskovec, Huttenlocher, and Kleinberg 2010), and the matrix factorization based LR (Chiang et al. 2013). Distinctly, SSL predicts link signs based on community detection. Provided that we have a community structure of a network, SSL predicts the sign of an incoming link based on the following rule: the link is positive if both end nodes fall into the same community, otherwise it is negative. Based on the same idea, the aforementioned PSA is also selected to join the comparison. The fraction of correct prediction is used to measure the performance of sign prediction, which is defined as $R_p = E_p/E_t$, where E_p is the number of links being correctly predicted and E_t is the total number of links to be predicted.

In this experiment, we follow the same way as suggested by (Chiang et al. 2013) to test the sign prediction performance of five algorithms, in which the learning data and testing data are generated as follows. Let N' be a fully-connected and balanced signed network, where there are five communities that contain 100, 200, 300, 400 and 500 nodes, respectively. A subnetwork N is constructed by sampling links from N' with a sampling rate s. N is regarded as an observed network for training and the rest links in $N' - N$ as incoming links for prediction. s takes the values 0.005, 0.01,0.02,0.03,0.05,0.07 and 0.1, alternatively. For each value, 100 subnetworks are sampled to calculate the prediction accuracy on average. Fig. 3(a) shows the performance of five algorithms. As we see, SSL, PSA and LR perform very good and stable; SSL provides the best prediction when $s > 0.01$.



(a) (b)

Figure 3: Performance comparisons of sign prediction.

Next, we test the five algorithms in a more challenge way by injecting different levels of noise into the above-generated balanced networks. In this validation, we first construct a subnetwork N with a sampling rate $s = 0.1$, and thereafter we change the signs of randomly selected links within or between communities with a noise rate ϵ, varying from 0.1 to 0.4 with an increment 0.03. Similarly, for each configuration we generate 100 subnetworks to calculate prediction accuracy on average. Fig. 3(b) shows the performance of five algorithms on such unbalanced networks. As we see, SSL works still better, particularly for the unbalanced networks containing more noise (i.e. $\epsilon > 0.25$). Note that, both SSL and PSA, the two community detection based methods, perform quite good among the five competitors for both balanced and unbalanced networks. This is probably because these methods implicitly take more information, provided by the global community structure in terms of both link density and sign, into account for prediction making.

Conclusion

Community detection and sign prediction are important for signed network analysis. Most of the existing methods are discriminative, which are depend on either predefined optimization objectives or heuristics. Distinctly, we propose a generative Bayesian approach to addressing these tasks, in which a signed stochastic blockmodel is proposed to characterize the block structures of signed networks in terms of both link density and sign and a variational Bayes method is proposed for model learning. To the best of our knowledge, this is the first effort in the literature to generalize the current SBM to address signed networks.

Acknowledgements

This work was funded by the Program for New Century Excellent Talents in University under Grant NCET-11-0204, and the National Science Foundation of China under Grants 61133011, 61373053, and 61300146.

References

Airoldi, E. M.; Blei, D. M.; Fienberg, S. E.; and Xing, E. P. 2008. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research* 9:1981–2014.

Anchuri, P., and Magdon-Ismail, M. 2012. Communities and balance in signed networks: A spectral approach. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, 235–242. IEEE Computer Society.

Bansal, N.; Blum, A.; and Chawla, S. 2004. Correlation clustering. *Machine Learning* 56(1-3):89–113.

Cartwright, D., and Harary, F. 1956. Structural balance: a generalization of heider's theory. *Psychological review* 63(5):277–293.

Chiang, K.-Y.; Natarajan, N.; Tewari, A.; and Dhillon, I. S. 2011. Exploiting longer cycles for link prediction in signed networks. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, 1157–1162. ACM.

Chiang, K.-Y.; Hsieh, C.-J.; Natarajan, N.; Tewari, A.; and Dhillon, I. S. 2013. Prediction and clustering in signed networks: A local to global perspective. *arXiv preprint arXiv:1302.5145*.

Daudin, J.-J.; Picard, F.; and Robin, S. 2008. A mixture model for random graphs. *Statistics and computing* 18(2):173–183.

Davis, J. A. 1967. Clustering and structural balance in graphs. *Human relations* 20(2):181–187.

Doreian, P., and Mrvar, A. 1996. A partitioning approach to structural balance. *Social networks* 18(2):149–168.

Girvan, M., and Newman, M. E. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99(12):7821–7826.

Gong, M.; Cai, Q.; Chen, X.; and Ma, L. 2014. Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *Evolutionary Computation, IEEE Transactions on* 18(1):82–97.

Hofman, J. M., and Wiggins, C. H. 2008. Bayesian approach to network modularity. *Physical review letters* 100(25):258701.

Holland, P. W., and Leinhardt, S. 1981. An exponential family of probability distributions for directed graphs. *Journal of the american Statistical association* 76(373):33–50.

Huang, Z., and Qiu, Y. 2010. A multiple-perspective approach to constructing and aggregating citation semantic link network. *Future Generation Computer Systems* 26(3):400–407.

Kropivnik, S., and Mrvar, A. 1996. An analysis of the slovene parliamentary parties network. *Developments in Statistics and Methodology* 209–216.

Kuncheva, L. I., and Hadjitodorov, S. T. 2004. Using diversity in cluster ensembles. In *Systems, man and cybernetics, 2004 IEEE international conference on*, volume 2, 1214–1219. IEEE.

Larusso, N.; Bogdanov, P.; and Singh, A. 2010. Identifying communities with coherent and opposing views. In *Proc. of the 15th Annual Graduate Student Workshop in Computing. Santa Barbara: UCSB*, 31–32.

Latouche, P.; Birmelé, E.; Ambroise, C.; et al. 2011. Overlapping stochastic block models with application to the french political blogosphere. *The Annals of Applied Statistics* 5(1):309–336.

Leskovec, J.; Huttenlocher, D.; and Kleinberg, J. 2010. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*, 641–650. ACM.

Liu, C.; Liu, J.; and Jiang, Z. 2014. A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. *Cybernetics, IEEE Transactions on* PP(99):1–1.

Mitrović, M., and Tadić, B. 2009. Spectral and dynamical properties in classes of sparse networks with mesoscopic inhomogeneities. *Physical Review E* 80(2):026123.

Newman, M. E. 2004. Fast algorithm for detecting community structure in networks. *Physical review E* 69(6):066133.

Palla, G.; Derényi, I.; Farkas, I.; and Vicsek, T. 2005. Uncovering the overlapping community structure of complex networks in nature and society. *Nature* 435(7043):814–818.

Pizzuti, C. 2008. Ga-net: A genetic algorithm for community detection in social networks. In *Parallel Problem Solving from Nature–PPSN X*. Springer. 1081–1090.

Read, K. E. 1954. Cultures of the central highlands, new guinea. *Southwestern Journal of Anthropology* 1–43.

Snijders, T. A., and Nowicki, K. 1997. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of classification* 14(1):75–100.

Traag, V., and Bruggeman, J. 2009. Community detection in networks with positive and negative links. *Physical Review E* 80(3):036115.

Yang, T.; Chi, Y.; Zhu, S.; Gong, Y.; and Jin, R. 2011. Detecting communities and their evolutions in dynamic social networksła bayesian approach. *Machine learning* 82(2):157–189.

Yang, B.; Cheung, W. K.; and Liu, J. 2007. Community mining from signed social networks. *Knowledge and Data Engineering, IEEE Transactions on* 19(10):1333–1348.

Yang, B.; Liu, J.; and Liu, D. 2012. Characterizing and extracting multiplex patterns in complex networks. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* 42(2):469–481.

Zhou, H. 2003. Distance, dissimilarity index, and network community structure. *Physical review e* 67(6):061901.