Community detection within edge exchangeable models for interaction processes

Yuhua Zhang, Walter Dempsey

September 26, 2021

Abstract

Health scientists are increasingly interested in discovering community structure from peer-to-peer support networks. While many methods have been proposed for finding community structure, few account for the fact that these modern networks arise from processes of interactions in the population such as user post and comment exchanges. We introduce cluster-wise edge exchangeable models for the study of interaction networks with latent community structure. In particular, we introduce the cluster-wise vertex components model as a canonical example. Several theoretical and practical advantages over traditional vertex-centric approaches are highlighted. In particular, cluster-wise edge exchangeable models allow for sparse degree structure and power law degree distributions within communities. Our theoretical analysis bounds the misspecification rate of cluster assignments, while supporting simulations show the properties of the network can be recovered. A computationally tractable Gibbs algorithm is derived. We demonstrate the proposed model using post-comment interaction data from Talklife, a large-scale online peer-to-peer support network.

Keywords— Large-scale Sparse Network, Edge Exchangeable Model, Community Detection, Power-law Degree Distribution

1 Introduction

The WorldWideWeb, social media, and technological innovation allow people from around the world to easily interact across geographical, cultural and economic boundaries. Among its many benefits, increased connectivity enhances information flow, promotes community building, and facilitates the formation of support systems for individuals struggling with illness and addiction. In this paper, we focus on network data arising from sequences of interactions collected on social media platforms such as peer-to-peer support networks. Scientists are primarily interested in improving peer support by understanding these digital interactions [7]. In suicide research, for example, social support has been shown to be a preventative factor for future suicidal ideation [13]. To further these goals, this paper focuses on discovery of latent community structure.

An increasingly popular class of models for community detection are the stochastic block models (SBMs) [11]. The basic version of the SBM assumes vertices within the same block have the same probability of forming an edge (i.e., interact) with other vertices, and within-cluster interactions are more likely than between-cluster interactions. Many associated methods have been proposed. These include but are not limited to spectral clustering based methods ([17], [3]), Bayesian methods ([15], [14]), and pseudo-likelihood based methods ([2], [19]). The stochastic block model was later generalized to the degree-corrected stochastic block model (DCSBM) [12], which allows for vertex degree heterogeneity. Theoretical guarantees of community recovery have been well established [8, 2, 26].

It is noteworthy that the large-scale network data are always of complex structures [1]. It remains challenging to correctly identify the underlying community structure in the real-life network. On the other hand, the intrinsic properties of the network can provide some insights into the revelation of the networks structure. Though SBM and the DCSBM have been proven success in many cases, the underlying assumptions of these two models genuinely fit into dense networks. Therefore, the power-law structure and the sparsity of the network, which arise naturally in many of the practical networks, are not taken into account in SBM and DCSBM. While the previously proposed edge exchangeable model has well addressed this problem [5]. It incorporated the power-law structure and the sparsity of the network into the estimation of the network properties. Previous

work on edge exchangeable model focus mainly on the inference of the statistics of the network. In this paper, we propose a new model which extends the edge exchangeable model to the community detection scenario. Our goal is to have a better understanding of the network structure through the identification of the latent communities formed by nodes in the network, as well as provide the accurate estimates of the power-law parameters of the corresponding sub-networks.

1.1 Outline and main contributions

The paper is organized in this way. We begin by defining the network data and notations in Section 3, followed by the description of the network generative process. In Section 4, we briefly discuss the inference algorithm and the simulation results. In Section 5, we show the application of our method using the TalkLife data and the interpretation of the results. We close the paper with the summary of the proposed methods and its implication in the discussion section.

2 Cluster-wise Edge Exchangeable Model

2.1 Motivating Example

Our motivating example is TalkLife, a large-scale online peer support network for mental health. The users of TalkLife post short snippets of text to which other users react and comments. The platform has collected millions of interactions among its users. A primary purpose of TalkLife is to strengthen its peer support networks and improve mental health outcome for its users. Many of these interventions aim to improve visibility of certain users in order to increase their social support. A natural question that arise is the identification of certain groups of people from the interaction data.

2.2 Notation and Data

We start by defining block-structured interaction data. Interaction data has been well defined in the previous work [5]. Here we extend the definition to the setting where there exists an underlying block structure. To illustrate our general setting, we use a simple example based on Talklife.

Recall each Talklife post consists of a poster p and a set of commentators $\{f_1, \ldots, f_k\}$. Then a post on Talklife can be summarized by the ordered pair $\{\{p\}, \{f_1, \ldots, f_k\}\}$. See Figure 1 for a visualization taking in the raw Talklife data and constructing the ordered pairs for a set of posts. Here, posters and commentators are drawn from the same underlying population, which we denote $\mathcal{P}$. Definition 2.1 formalizes this example into a general definition of interaction data.

Definition 2.1 (Interaction data) For a set $\mathcal{P}$, let $fin(\mathcal{P})$ be the multisets of all finite subsets of $\mathcal{P}$. The interaction process of $\mathcal{P}$ is the correspondence $\mathcal{I} : \mathbb{N} \to fin(\mathcal{P})$ between the natural numbers and finite subsets of $\mathcal{P}$.

Additional structure can be incorporated into Definition 2.1. In Talklife, for example, there is a single poster and multiple commentators. Definition 2.1 can capture this by replacing $fin(\mathcal{P})$ with $fin_1(\mathcal{P}) \times fin(\mathcal{P})$ where $fin_1(\mathcal{P})$ are sets of size one from $\mathcal{P}$. See [6] for additional examples of these extensions.

Using Figure 1 as an example, each interaction is represented by the user who posts and the users who comment. The users, regardless of being posters or commentators, are drawn from the population $\mathcal{P}$ set equal to $\mathbb{N}$. The interaction process is a correspondence $\mathcal{I} : \mathbb{N} \to fin_1(\mathbb{N}) \times fin(\mathbb{N})$. For Post 1, $\mathcal{I}(1) = (\{1\},\{3, 4, 5\})$ representing the poster posted by User 1 being commented by User 3, 4, and 5. The second Post $\mathcal{I}(2) = (\{2\},\{6, 7\})$ represents User 2 posts and User 6 and 7 comment.

Individuals who more frequently comment on a user's post are likely to be from the same underlying community. In Figure 1, for example, User 3, 4, and 5 are in the same cluster, which leads to a higer likelihood of observing these individuals commentating on a post by User 1. User 8 and User 9 are in a separate cluster, which explains the second interaction. Note, however, that users in distinct clusters may comment on each other's posts, i.e., User 7 comments on a post by User 2. We formalize community structure in interaction data in Definition 2.2.

Definition 2.2 (Block structured interaction data) A *K-block structure* is defined as the mapping $B : \mathcal{P} \to [K]$, such that for all $x \in \mathcal{P}$, $B(x) \in [K]$ is the block that x is assigned. The

interaction process induced by B is called the *block interaction process*, which is the correspondence $\mathcal{I}_B : \mathbb{N} \to fin([K])$.

In our running example shown in Figure 1, $K = 2$ and the interaction process induced by B is $\mathcal{I}_B(1) = (\{1\}, \{1, 1, 1\})$ for Post 1, and $\mathcal{I}_B(2) = (\{1\}, \{1, 2\})$ for Post 2, and $\mathcal{I}_B(3) = (\{2\}, \{2, 2\})$ for Post 3. Note that for the pre-image B^{-1}, $B^{-1}(i) \cap B^{-1}(j) = \emptyset$ if $i \neq j$, and $B^{-1}(i) = B^{-1}(j)$ if $i = j$ for $i, j \in [K]$. That is, B partitions $\mathcal{P}$ into K non-overlapping sets.

Above, interaction processes are distinct if they are defined on different populations $\mathcal{P}$ and $\mathcal{P}'$. Here, we formally define an equivalence of networks built from two interaction processes. Let $\rho : \mathcal{P} \to \mathcal{P}'$ be a bijection between the two populations. Then ρ induces an action on $(\mathcal{P}) \to (\mathcal{P}')$; that is, for $(f_1, \ldots, f_m) \in fin(\mathcal{P})$, we have

$$\rho \circ (f_1, \ldots, f_m) = (\rho(f_1), \ldots, \rho(f_m)) \in fin(\mathcal{P}').$$

With this, the bijection also acts on the associated block interaction process $\rho I_B : \mathbb{N} \to fin([K])$ by $\rho I_B(n) = B \circ (\rho \circ I)(n)$. That is:

$$B \circ (\rho \circ (f_1, \ldots, f_m)) = (B(\rho(f_1)), \ldots, B(\rho(f_m))) \in fin([K])$$

In other words, the bijection preserves the block structure. With this, the network data with finite observations $m \in \mathbb{N}$ can be induced by the equivalence class:

$$\mathbf{y}_{\mathcal{I},B} = \bigcup_{\#\mathcal{P}'=\#\mathcal{P}} \{\mathcal{I}' : \mathbb{N} \to fin(\mathcal{P}') : \rho\mathcal{I} = \mathcal{I}',\ \rho\mathcal{I}_B = \mathcal{I}'_B,\ \text{for } \rho : \mathcal{P} \to \mathcal{P}'\}$$

In the remainder of the paper, we write $\mathcal{Y}_m$ to indicate the observed network with edges labeled in $m \in \mathbb{N}$, and $E_1, E_2, \ldots, E_m$ as the interactions observed, where $E_j = (\{S_j, C_j^{(s)}\}, \{R_j, C_j^{(r)}\})$, $j \in [m]$, that is, the interaction j is initiated from the sender S_j of cluster $C_j^{(s)}$ and points to the receiver R_j of cluster $C_j^{(r)}$.

Let $v(\mathcal{Y}_m)$ be the number of vertices in the network, and $m(\mathcal{Y}_m)$ be the total degree of the network. Note that $m(\mathcal{Y}_m) = 2m$. We subscript $c \in [K]$ to indicate the cluster specific variables. For example, $m(\mathcal{Y}_c)$ is the total degree of cluster c. The interactions within the same cluster, denoted

as $\mathcal{Y}_c$, $c \in [K]$, are assumed to be exchangeable. That is, for the edges initiated from the same cluster $\mathcal{Y}_c \subset \mathcal{Y}_m$, we have $\mathcal{Y}_c^{\sigma} =_D \mathcal{Y}_c$, for all permutations $\sigma : \mathcal{P} \to \mathcal{P}$.

2.3 Sequential description of the C-VCM

We next provide a sequential description of a particular family of relatively exchangeable interaction processes, termed the C-VCM. For ease of comprehension, the process is described in the context of TalkLife posts assuming a single commentator. Suppose m^* posts $E_{[m^*]} := \{E_1, ..., E_{m^*}\}$ have been observed along with the cluster assignments for each observed individual. That is, the post $j \in [m^*]$ is given by $E_j = (\{S_j, C_j^{(s)}\}, \{R_j, C_j^{(r)}\})$ where S_j is the sender and R_j is the commentator, and the cluster assignments are given by $C_j^{(s)}$ and $C_j^{(r)}$. Let $\mathcal{H}_{m^*}^{(c)}$ denote the set of unique individuals in cluster $c \in [K]$ that have either posted or commented on a post in the first m^* interactions.

The next post, E_{m^*+1}, conditional on $E_{[m^*]}$ and the associated cluster assignments are constructed by first drawing a cluster, denoted $C_{m^*+1}^{(s)}$, according to

$$P(C_{m^*+1}^{(s)} = c | \pi_1, ..., \pi_K) \sim \text{Multinomial}(\pi_1, ..., \pi_K)$$

where π_c is the pre-specified propensity of an interaction initiated from the cluster c, such that $\sum_{c=1}^{K} \pi_c = 1$. Note that each individual per cluster can be involved in multiple interactions, either as a poster or commentator. Let $D_{m^*}(i, c)$ be the number of times individual $i \in \mathcal{H}_{m^*}^{(c)}$ in cluster c has been observed (either has poster or commentator) in the first m^* interactions. Let S_{m^*+1} denote the poster for the (m^*+1)st interaction. Then, similar to the Chinese Restaurant Process [16], given parameters $0 < \alpha_c < 1$ and $\theta_c > -\alpha_c$ for each $c \in [K]$, poster S_{m^*+1} can be either chosen from one of the previously observed nodes i in cluster c, or an unobserved node in cluster c with probability:

$$P(S_{m^*+1} = s \mid \mathcal{H}_{m^*}, C_{m^*+1}^{(s)} = c) \propto \begin{cases} D_{m^*}(i, c) - \alpha_c, & \text{if } s = i \in \mathcal{H}_{m^*}^{(c)} \\ \theta_c + \alpha_c \sum_{i \in \mathcal{H}_{m^*}^{(c)}} D_{m^*}(i, c), & \text{if } s \notin \mathcal{H}_{m^*}^{(c)} \end{cases} \tag{1}$$

If a new individual in cluster c is chosen, that person is given the next label in the order of appearance. For example, if they are the 10th person in cluster 1 to be observed, they are labelled as individual 10 in cluster 1.

Given the sender, a similar process is used to choose the $(m^*+1)st$ commentator. Given the block structure, the cluster for the receiver, denoted $C^{(r)}_{m^*+1} = c' \in [K]$, for the receiver node is chosen with the corresponding probability given by the propensity matrix $\mathcal{B}$,

$$P(C^{(r)}_{m^*+1} = c' | C^{(s)}_{m^*+1} = c, \mathcal{B}) \sim \text{Multinomial}(\mathcal{B}(c,1), ..., \mathcal{B}(c,K))$$

Let the set of previously observed nodes in cluster c' in the first m^* interactions be $\mathcal{H}^{(c')}_{m^*}$.

Note that if $c' = c$, $\mathcal{H}^{(c')}_{m^*} = \mathcal{H}^{(c')}_{m^*} \cup \{s\}$, and $D_{m^*}(s,c') = D_{m^*}(s,c') + 1$. Let R_{m^*+1} be the commentator of the (m^*+1)st interaction. Similar to the process of selecting the sender S_{m^*+1}, conditional on $\mathcal{H}^{(c')}_{m^*}$, the receiver node R_{m^*+1} is selected with probability:

$$P(R_{m^*+1} = r | \mathcal{H}^{(c')}_{m^*}, C^{(r)}_{m^*+1} = c') \propto \begin{cases} D_{m^*}(i,c') - \alpha_{c'}, & \text{if } r = i \in \mathcal{H}^{(c')}_{m^*} \\ \theta_{c'} + \alpha_{c'} \sum_{i \in \mathcal{H}^{(c')}_{m^*}} D_{m^*}(i,c'), & \text{if } r \notin \mathcal{H}^{(c')}_{m^*} \end{cases} \tag{2}$$

Note that if either the poster or commentator is the first node selected from the cluster, then this node is labelled as individual 1 in that cluster. Let $\mathbf{C}_{m^*}$ denote the cluster assignments of all observed nodes in the first m^* interactions. Based on the above sequential description, given the total number of observed interactions being m ($m \geq m^*$), the proposed relatively exchangeable interaction process (C-VCM) takes the explicit form:

$$\begin{aligned} P(\mathcal{Y}_m = \mathbf{y}_m | \{\alpha_c, \theta_c\}, \mathbf{C}_m) = & \prod_{c=1}^{K} \pi_c^{L_c} \times \prod_{j=1}^{m} P(S_j = s | \alpha_c, \theta_c, \mathcal{H}^{(c)}_j, C^{(s)}_j = c) \\ & \times P(R_j = r | \alpha_{c'}, \theta_{c'}, \{S_j\}, \mathcal{H}^{(c')}_j, C^{(r)}_j = c') \times \prod_{c'=1}^{K} \mathcal{B}(c,c')^{W_m(c,c')} \end{aligned} \tag{3}$$

where L_c is the number of observed interactions that are initiated from cluster c; $W_m(c,c')$ is the number of observed interactions from cluster c to c'. The joint distribution of all the nodes within the same cluster $\mathcal{H}^{(c)}_m$ can be expressed as

$$P(\mathcal{H}_m^{(c)}|\alpha_c,\theta_c,\{C_m\}) = \frac{[\alpha_c+\theta_c]_{\alpha_c}^{v(\mathbf{y}_c)-1}}{[\theta_c+1]_1^{m(\mathbf{y}_c)-1}} \prod_{j\in\mathcal{H}_m^{(c)}} [1-\alpha_c]_1^{D_m(j,c)-1} \tag{4}$$

where for real number x and a, and non-negative integer N, the operation on x, $[x]_a^N = x(x+a)...(x+(N-1)a)$. The following equivalent form of (3) can be obtained by replacing the product with (4)

$$P(\mathcal{Y}_m = \mathbf{y}_m|\{\alpha_c,\theta_c\},\mathbf{C}_m) = \prod_{c=1}^{K} \pi_c^{L_c} \times \underbrace{\frac{[\alpha_c+\theta_c]_{\alpha_c}^{v(\mathbf{y}_c)-1}}{[\theta_c+1]_1^{m(\mathbf{y}_c)-1}} \prod_{j\in\mathcal{H}_m^{(c)}} [1-\alpha_c]_1^{D_m(j,c)-1}}_{(II)} \times \prod_{c'=1}^{K} \mathcal{B}(c,c')^{W_m(c,c')} \tag{5}$$

3 Simulation

In this section, we are going demonstrate the efficacy of our algorithm regarding the inference of network structures. We start with a brief introduction of the inference algorithm, and proceed into the accurate estimate of the power-law parameters and the recoverability of the cluster assignment. At the end of this section, we propose one potential model selection criteria based on the marginal log likelihood.

3.1 Inference algorithm

Unfortunately, the likelihood shown in Eq(5) does not have closed form solutions for the parameters $\{\alpha_c\}$, $\{\theta_c\}$, and $\{\mathbf{C}_m\}$. Here, we present a Bayesian approach to inference for these model parameters. An important problem is the selection of the priors. First, consider the cluster assignment $\mathbf{C}_m$. It is natural to assume that:

$$P(\mathbf{C}_m) \sim \text{Multinomial}(\pi_1,...,\pi_K),\ \pi \sim \text{Dirichlet}(\eta)$$

Without strong prior information, we'd suggest η being a symmetric Dirichlet.

As of the connection between and within the cluster, a node has to be connected to another

node either in the same cluster or in the other cluster. Hence, the entries in the same row of propensity matrix $\mathcal{B}$ should sum up to 1. Without forcing the propensity matrix to be symmetric, we proposed here a row-wise Dirichlet prior:

$$P(\mathcal{B}(c,\cdot)) \sim \text{Dirichlet}(\nu),\ c \in [K]$$

In terms of the priors of power-law parameters $\{\alpha_c\}$ and $\{\theta_c\}$, let $\alpha_c \sim P(\mu)$, $\theta_c \sim P(\delta)$, $c \in [K]$. A reasonable choice could be $P(\mu)$ takes the form of a Beta distribution, while $P(\sigma)$ takes the form of Gamma distribution such that the posterior distribution will have the close form to sample from.

Given the proper choice of priors, we introduce a Bayesian computation is performed via a Gibbs-style implement the algorithm built upon Gibbs sampling to infer the model parameters. Based on the description of our model, the parameters to be updated are the power-law parameters $\{\alpha_c\}$, $\{\theta_c\}$, and the latent clustering assignment indicator $\{\mathbf{C}_m\}$.

The first parameter to be updated is $\{\mathbf{C}_m\}$. The conditional updates of latent cluster indicator $\{\mathbf{C}_m\}$ is based on the following derivations. Suppose the algorithm is assigning a specific node $s \in \{\mathcal{H}_m\}$ to a specific cluster $c \in \{1, ..., K\}$. The corresponding probability p_c is proportional to:

$$p_c \propto P(\{C_m^{(s)}\} = c|\gamma)^{(1)} \times P(\mathcal{B}(C_m^{(s)} = c,)|\nu)^{(2)} \times P(\{\mathcal{H}_m^{(c)}\}|\alpha_c, \theta_c, \delta, \mu)^{(3)} \tag{6}$$

The conditional updates of α_c and θ_c is achieved through auxiliary variable sampling methods [21] that was also introduced in the hierarchical edge exchangeable model [6]. We introduce the auxiliary variables x, y, and z, such that the posterior of θ_c and α_c are given by

$$\theta_c|C_m \sim \text{Gamma}(\sum_{i=1}^{v(\mathbf{y}_c)-1} y_i + a, b - logx) \tag{7}$$

.

$$\alpha_c|C_m \sim \text{Beta}(c + \sum_{i=1}^{v(\mathbf{y}_c)-1} (1-y_i), d + \sum_{s_c} \sum_{j=1}^{D_m(c,s)-1} (1 - z_{s_c,j})) \tag{8}$$

Given the cluster allocation $c \in [K]$, we have the row-wise update of the propensity matrix

taking the following form:

$$\mathcal{B}(c,)|C_m \sim \text{Dirichlet}(\nu + \vec{*}) \tag{9}$$

where $\vec{*}$ is the count of interactions between cluster c and cluster $\{1, ..., K\}$.

3.2 Power-law parameter inference

An important question is whether the posterior distributions adequately concentrate around the true cluster-specific parameters $\{\alpha_c\}_{c\in[K]}$. Recall that in the proposed model α_c controls the within-cluster sparsity and power-law structure. Therefore, to capture important network characteristics, our inferential algorithm needs to produce accurate parameter estimates. In this section, we use the generative model as described in Section 3.3 to simulate different network datasets to assess our ability to accurately estimate α_c in various settings.

Here we focus on simulations where the number of clusters K is equal to 2. In this setting, we choose $\{\alpha_1, \alpha_2\}$ from $\{\{0.1, 0.9\}, \{0.2, 0.8\}, \{0.3, 0.7\}, \{0.4, 0.6\}\}$ and set $\theta_1 = \theta_2 = 5$. The probability of observing an interaction within the same group is set to be 0.9 (i.e., $\mathcal{B}(1,1) = \mathcal{B}(2,2) = 0.9$), and between groups is set to be 0.1 (i.e., $\mathcal{B}(1,2) = \mathcal{B}(2,1) = 0.1$). Denote the within group connectivity as a, and the between group connectivity as b. We also assumed the interaction is equally likely to be initiated from both clusters (i.e., $\{\pi_1, \pi_2\} = \{0.5, 0.5\}$). A total number of 1,000, 10,000 and 100,000 interactions were simulated separately. The same simulation scheme was repeated 20 times. The estimates of the power-law parameters and the entries in propensity matrix are shown in Table 1. To assess how much uncertainty is the result of latent communities, we also infer the power-law parameters given the true cluster assignments and denote this estimate by $\tilde{\alpha}$.

We comment on three important findings based on Table 1. First, estimation accuracy improves when α is larger. For example, in the 1000 interactions setting where $\{a, b\} = \{0.9, 0.1\}$ and $\{\alpha_1, \alpha_2\} = \{0.3, 0.7\}$ (highlighted in Table 1 in red), the power-law estimates when the truth is equal to 0.7 are more accurate than the estimates when the truth is equal to 0.3. This conclusion holds regardless of whether the community labels are known or unknown. Second, power-law estimation accuracy degrades as the inter-community connection rate increases. For example, in the 1000

interaction setting where $\{\alpha_1, \alpha_2\} = \{0.2, 0.8\}$, the power-law estimate decreases in accuracy as the likelihood of inter-community connections goes up from 0.1 to 0.5 (highlighted in Table 1 in yellow). Third, estimation accuracy increases as the number of interactions increase. For example, in the case where $\alpha_1 = 0.1$, and $\{a, b\} = \{0.9, 0.1\}$, the power-law estimate becomes more accurate as the sample size grows from $N = 1,000$ to $100,000$. Also note that the gap in accuracy between power-law estimates when the true block structure is known and unknown decreases as the number of interactions increase.

3.3 Recovering Cluster Assignment Labels

Another important question is whether the proposed inferential algorithm for the C-VCM is able to recover the correct cluster assignment labels. The same simulation settings are inherited from the previous section, and are used to empirically confirm the ability of our algorithm to recover the underlying network structure when only a finite number of interactions are observed. The cross entropy loss is used as the metric to measure the difference between the inferred and true cluster assignments. We start by defining the cross entropy loss for a specific cluster $c \in [K]$:

$$\mathscr{L}_c = \sum_{j \in X_c} -p(C_j = c) \log q(\hat{C}_j = c)$$

where the $p(\cdot)$ is the probability of the true cluster assignment, and $q(\cdot)$ is the probability of the inferred cluster assignment. For example, in the two clusters scenario, a node s has cluster labeling $C_s = 1$ in the generative model, while in the inferred cluster assignment, the probability node s being assigned to cluster s is 0.6, the cross entropy loss for node s of cluster 1 is $\mathscr{L}_s = -1 \times \log(0.6)$. In the best case, all the nodes are assigned correctly with probability one and the resulting entropy loss is 0; if cluster assignments are incorrectly assigned with probability one, the entropy loss is infinite. A uniform distribution over cluster assignments (i.e., random guess) has expected entropy loss of a specific node is $-\log(0.5)$. Further define the overall cross entropy loss $\mathscr{L}$ and the average per node entropy loss $\mathscr{L}_s$ as:

$$\mathscr{L} = \sum_{c=1}^{K} \mathscr{L}_c; \ \mathscr{L}_s = \frac{1}{v(y_m)} \mathscr{L}$$

The results are shown in Table 2. Note that for fixed α, the higher the intra-community connection probability, the better the recovery of the cluster assignments. Second, as the number of observations increases, the cluster assignments become more accurate in almost all settings. Third, fixing the number of interactions and inter-community connection rate, as the difference in power-law parameters decreases between the two clusters, the cluster assignment accuracy decreases.

3.4 The selection of K

So far, the simulation focuses on the scenario where there are only two clusters. It remains to be addressed the selection of K. Previous work on SBM and DCSBM has shed some light on this problem. The well-acknowledged solutions include but are not limited to the likelihood ratio test based methods ([22], [23]), the cross validation based methods ([10],[9]), and the Bayesian selection criteria ([24],[20]). Though proven to be a big success in many cases, these methods cannot fit into our settings well. For example, the likelihood ratio test based method requires a good distributional approximation, which is hard to find in our setting; and cross validation based methods bring the even more complex sampling issues and the lack of the interpretation; and the implementation of Bayesian selection criteria can be very computationally expensive.

Therefore, we propose a simple but powerful way to select the number of clusters based on the marginal log-likelihood given in Eq 5, where the number of clusters is determined by choosing the K that gives the largest likelihood or the K right before the likelihood begin to drop. It remains to be explored a more sophisticated method to determine the number of clusters. But given our main goal in this paper is not about model selection, the idea we proposed is enough for this work, which is straight forward and easy to implement without the need to tune any parameters.

4 Case Study

4.1 Data overview

In this section, we apply the C-VCM to the TalkLife dataset which consists of millions of user posts and comments. Recall that TalkLife is a large-scale peer support network for mental health.

Individuals on the network receive support for a variety of mental health issues including anxiety, depression, eating disorders, and self-harm. Users can post to the platform and respond to other users' posts. The social network has collected millions of interactions among users to date. To better understand how users organize on their platform, TalkLife is interested in identification of any underlying communities and each community's network properties.

Here we consider all posts on TalkLife during the year 2019. Since users can delete their own posts and comments, we limit our analysis to those posts that were not later deleted by the user, that is a total of 1,508,895 posts with an average of 2.82 comments per post, leading to 4,258,792 post-comment pairs. TalkLife deploys a series of classification algorithms to determine whether a post can be flagged as including language related to one of several mental health topics. For example, a post could be flagged as *Anxiety Panic Fear Suspected* if the corresponding classifier was triggered by the text of the post. There are a total of 33 classifiers applied to the majority of posts, leading to a final dataset consisting 1,481,296 posts with an average of 2.83 comments, summing up to 4,194,609 post-comment pairs.

Figure 2 shows the global degree distribution for posts, and the distribution for subsets of posts flagged by 4 out of 33 different classifiers, including Emptiness Suspected, Nausea With Eating Disorder Suspected, SelfHarm Remission Or Relapse Suspected, and Emotional Exhaustion Suspected. The power-law degree distribution is apparent in the overall network of 2019, as well as the classifier-specific networks. The empirical phenomenon exactly fits the power-law assumption of the proposed model. We thus applied our model to the data in the following sections.

4.2 Inferring Cluster labels

In this section, we aim to identify the community structure of TalkLife's users. In Section 2.3, the C-VCM only considers pairs $(S, \{R_1, ..., R_k\})$. That is, a specific user S only compose a single poster. In TalkLife, however, the data consists of multiple posters posted by a singe user. That is, the jth post by user S can be summarized by $(S_j, \{R_{j,1}, \ldots, R_{j,k_j}\})$ where $k_j \geq 1$ is the number of commentators on the jth post. Here, we make a conditional independence assumption such that the link between poster and commentator depends not only on their propensities and cluster

assignments, but the poster as well, that is, for a specific poster S:

$$P(\cup_j\{S_j,\{R_{j,1},\ldots,R_{j,k_j}\}\}|C_s=c,\pi,f,B)=\pi_c\times\prod_j f_{s_j}^{(c)}\times\prod_{l=1}^{k_j}\left[f_{r_l}^{(c_l)}\times B(c,c_l)\right]. \quad (10)$$

where f is the degree-related parameter, the value of which is determined by the power-law parameters and the degrees of the nodes in the network.

Recall that each poster comes up with a presumed topic tag. In the following analysis, we focus on users that pay attention to posts of similar themes. We subset the data based on these intrinsic machine learning generated tags. That is, we construct the network and run our algorithm on the posters belonging to a specific topic, such as Alcohol and Substance Abuse suspected, which gives a total number of 33 sets of network data. We consider a varying number of clusters and infer the cluster assignments of each user over a range of K values.

We further investigate the within and the between cluster connectivity of the inferred communities. Using Alcohol and Substance Abuse network as an example, the connectivity of the detected communities when K is set to be 2, 6, and 10 is shown in Figure 3. It is consistently witnessed that the within community connectivity is greater than the between community connectivity based on the results of our method, which indicates the existence of the clustering structures of the users. As aforementioned, to boost the online social supports for certain users, it is of great importance to identify the group of people that have frequent interactions with them and strengthen the social supports among them.

4.3 Comparison with Spectral Clustering

In the section, we further compare our method with the spectral clustering method. The connectivity of the communities detected by the spectral clustering method is shown in Figure 3. Note that the spectral clustering method gives very different community structures as compared to our method in the sense that it tends to cluster nodes into one or two large chunks. This makes the communities detected by the spectral clustering method less informative than our method.

To benchmark the results, we check the consistency of the clusters inferred by our algorithm over time. We split the data into two halves based on whether the post was posted before Jun.30th,

and fit our model into the two halves of data separately. To quantify the difference of the cluster assignments in the two halves of data, we use the Hellinger distance defined as:

$$HL = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sqrt{2}}\sqrt{\sum_{k=1}^{K}(\sqrt{p_{ki}}-\sqrt{q_{ki}})^2}$$

where K is the number of presumed clusters, and n is the number of nodes; p and q correspond to the probability of a node belonging to one of the Kth clusters given by the Gibbs iterations and the underlying truth. Note that when K is large, there will be the matching of labeling issue. To overcome this problem, we randomly sample 20 possible combinations and take the minimal Hellinger distance as the final one.

For the illustrative purpose, we use Alcohol and Substance Abuse network and Behavoiral Symptoms network as two examples. The results are shown in Figure 4. In the Behavorial symptoms suspected network, for example, the overall likelihood hits the largest value when $K = 4$, which indicates setting the underlying communities to be 4 is the most reasonable grouping of users. In the both networks, the Hellinger Distance of clustering assignments of our model is consistent with the overall marginal likelihoods. It serves to prove the efficacy of the heuristic strategy mentioned in Section 3.4 in real data.

We also compare our method with the spectral clustering method. The minimal Hellinger Distance of the cluster assignment over different K is smaller in our model than in the spectral clustering, which indicates the communities detected by our method are more consistent over time than the communities detected by the spectral clustering method. That is, the trend holds true over time that users tend to have more connections with people from the same inferred communities than people outside the inferred communities.

5 Discussion

In this paper, we have proposed a new method that can be applied to cluster the nodes and identify the underlying community structures of real world networks. Specifically, our method features incorporating the power-law property which is observed in the large sparse networks. We

demonstrate the efficacy of our algorithm through simulation and finalize this paper by applying our method to the TalkLife data to identify the potential online user communities.

The contribution of our work is to help people understand the network structure better. Using TalkLife data as an example, our method can correctly identify the users' groups so that the platform will know to whom they can boost the visibility of the posts from certain users. Besides, our method provides the estimates of the power-law parameters for each of the group. The consistency of the communities detected by our method guarantees the effect of the interventions provided by the platform over time. The revelation of the network properties can also help the platform make proper decisions on what kind of interventions to make. For example, a smaller power-law parameter indicates a looser connection between users. The platform will need extra efforts to ensure certain users within the community getting enough attention and social supports. Note that though we demonstrate our method using the TalkLife data, it can be well applied to any sparse interaction data sets, such as Copenhagen study [18].

There are a numerous directions for future work. For example, our method only focuses on the classification of the nodes in the network. But, it can also be further extended to the clustering of different types of the connection. For example, the classification can be based on the type of interactions other than the nodes. Besides, when we adjust our model to fit into TalkLife data, we assume independence between different interactions. In future, it is possible for us to take into consideration the dependence between different interactions [25]. Another potential direction is increasing the speed of the algorithm given the size of the interaction data can easily hit millions[4]. The scalability of the algorithm need to meet the increasing scale of the data.

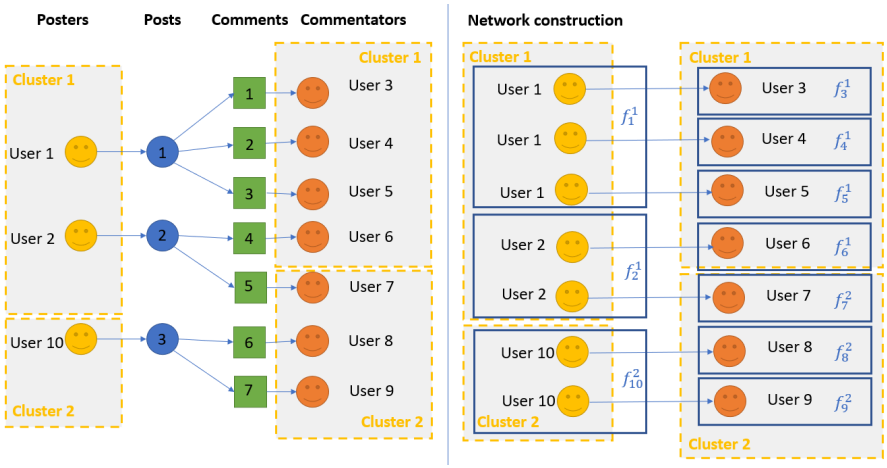
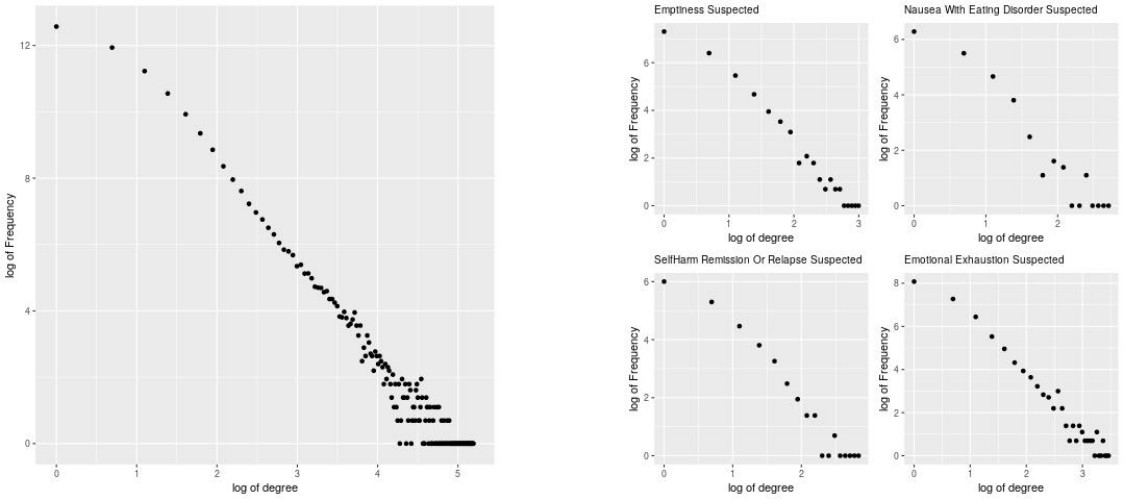
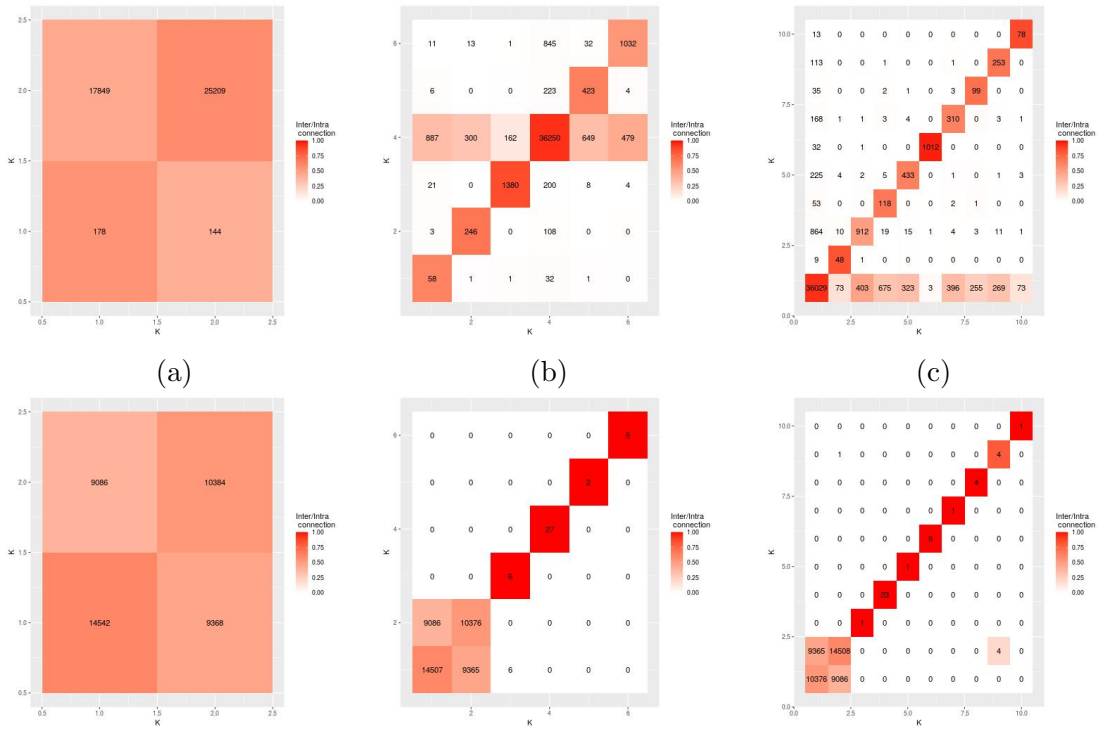


Figure 1: Interaction data collected on TalkLife (Left), and the corresponding network (Right). In this example, there are 3 posts and 10 users involved. Post 1 posted by User 1, is commented by User 3, 4, and 5; Post 2 posted by User 2 is commented by User 6 and 7; Post 3 posted by User 10 is commented by User 8 and 9. The network is constructed by connecting User 1 to User 3, 4, and 5; User 2 to User 6 and 7; User 10 to User 8 and 9. Note that User 1, 2, 3, 4, 5, 6 are from cluster 1, and the rest users are from cluster 2. Therefore, 4 of the interactions are from cluster 1 to cluster 1, 1 of them is from cluster 1 to cluster 2, the rest are from cluster 2 to cluster 2.



(a) Overall degree plot of 2019 senders (b) Degree plot of 2019 senders by different tags

Figure 2: (a) Overview of the degree distribution of 2019 senders; (b) Degree distribution of the 2019 senders in sub-networks



(a) (b) (c)

(d) (e) (f)

Figure 3: The inter/intra connectivity of the communities detected by our algorithm (the first line) spectral clustering (the second line), ranging from 0 to 1, indicating the proportion of the nodes that have within/between cluster connectivity. The number within each cell is the number of nodes.

1,000 interactions	Parameters	$\alpha = \{0.1, 0.9\}$	$\alpha = \{0.2, 0.8\}$	$\alpha = \{0.3, 0.7\}$	$\alpha = \{0.4, 0.6\}$
$\mathcal{B} = \{0.1, 0.9\}$	α_1	**0.301 (0.182)**	**0.296 (0.116)**	**0.374 (0.08)**	**0.449 (0.083)**
	$\tilde{\alpha}_1$	0.200 (0.100)	0.274 (0.084)	0.354 (0.068)	0.460 (0.086)
	α_2	**0.886 (0.079)**	**0.804 (0.022)**	**0.712 (0.037)**	**0.612 (0.056)**
	$\tilde{\alpha}_2$	0.904 (0.015)	0.805 (0.021)	0.714 (0.037)	0.604 (0.057)
	Diagonal	0.907 (0.020)	0.905 (0.019)	0.903 (0.019)	0.898 (0.018)
$\mathcal{B} = \{0.3, 0.7\}$	α_1	**0.335 (0.196)**	**0.335 (0.139)**	**0.412 (0.143)**	**0.501 (0.099)**
	$\tilde{\alpha}_1$	0.226 (0.094)	0.256 (0.095)	0.384 (0.113)	0.470 (0.100)
	α_2	**0.886 (0.069)**	**0.801 (0.029)**	**0.704 (0.061)**	**0.604 (0.064)**
	$\tilde{\alpha}_2$	0.901 (0.015)	0.806 (0.025)	0.712 (0.061)	0.617 (0.058)
	Diagonal	0.709 (0.031)	0.710 (0.028)	0.708 (0.030)	0.708 (0.034)
$\mathcal{B} = \{0.5, 0.5\}$	α_1	**0.627 (0.162)**	**0.624 (0.185)**	**0.613 (0.115)**	**0.523 (0.124)**
	$\tilde{\alpha}_1$	0.212 (0.096)	0.290 (0.079)	0.368 (0.078)	0.479 (0.105)
	α_2	**0.800 (0.267)**	**0.726 (0.125)**	**0.604 (0.112)**	**0.555 (0.106)**
	$\tilde{\alpha}_2$	0.898 (0.017)	0.804 (0.025)	0.710 (0.030)	0.601 (0.064)
	Diagonal	0.521 (0.042)	0.532 (0.051)	0.540 (0.053)	0.526 (0.050)
10,000 interactions	Parameters	$\alpha = \{0.1, 0.9\}$	$\alpha = \{0.2, 0.8\}$	$\alpha = \{0.3, 0.7\}$	$\alpha = \{0.4, 0.6\}$
$\mathcal{B} = \{0.1, 0.9\}$	α_1	**0.201 (0.089)**	**0.273 (0.083)**	**0.336 (0.051)**	**0.426 (0.042)**
	$\tilde{\alpha}_1$	0.175 (0.063)	0.244 (0.063)	0.336 (0.043)	0.420 (0.039)
	α_2	**0.899 (0.006)**	**0.799 (0.011)**	**0.700 (0.017)**	**0.605 (0.020)**
	$\tilde{\alpha}_2$	0.900 (0.006)	0.800 (0.011)	0.701 (0.017)	0.607 (0.019)
	Diagonal	0.900 (0.006)	0.900 (0.006)	0.901 (0.006)	0.900 (0.006)
100,000 interactions	Parameters	$\alpha = \{0.1, 0.9\}$	$\alpha = \{0.2, 0.8\}$	$\alpha = \{0.3, 0.7\}$	$\alpha = \{0.4, 0.6\}$
$\mathcal{B} = \{0.1, 0.9\}$	α_1	**0.157 (0.063)**	**0.242 (0.054)**	**0.322 (0.034)**	**0.413 (0.025)**
	$\tilde{\alpha}_1$	0.136 (0.045)	0.220 (0.040)	0.319 (0.027)	0.412 (0.022)
	α_2	**0.900 (0.002)**	**0.799 (0.004)**	**0.700 (0.006)**	**0.604 (0.012)**
	$\tilde{\alpha}_2$	0.900 (0.002)	0.799 (0.004)	0.700 (0.006)	0.605 (0.011)
	Diagonal	0.900 (0.002)	0.900 (0.002)	0.900 (0.002)	0.900 (0.002)

Table 1: Estimates of network parameters over different settings

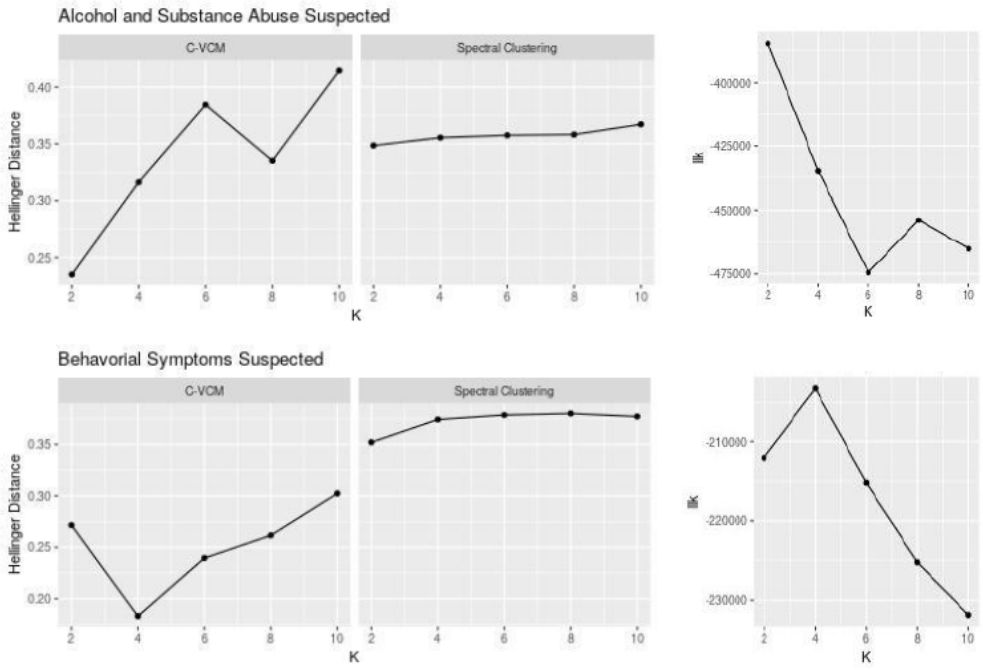


Figure 4: The Hellinger distances of the cluster assignment between the first half and the second half of the 2019 data of Alcohol and Substance Abuse suspected network and Behavorial Symptoms Suspected network.

Table 2

Cross Entropy Loss over different settings

	Interactions	$\alpha = \{0.1, 0.9\}$	$\alpha = \{0.2, 0.8\}$	$\alpha = \{0.3, 0.7\}$	$\alpha = \{0.4, 0.6\}$
$\mathcal{B} = \{0.1, 0.9\}$	1,000	0.056 (0.014)	0.095 (0.028)	0.152 (0.028)	0.212 (0.040)
	10,000	0.014 (0.004)	0.039 (0.013)	0.094 (0.012)	0.170 (0.032)
	100,000	0.0027 (0.0006)	0.013 (0.004)	0.054 (0.012)	0.134 (0.018)
$\mathcal{B} = \{0.2, 0.8\}$	1,000	0.084 (0.020)	0.151 (0.032)	0.239 (0.065)	0.343 (0.050)
	10,000	0.019 (0.005)	0.055 (0.015)	0.157 (0.084)	0.298 (0.074)
	100,000	0.013 (0.033)	0.018 (0.004)	0.097 (0.040)	0.223 (0.029)
$\mathcal{B} = \{0.3, 0.7\}$	1,000	0.102 (0.152)	0.182 (0.050)	0.346 (0.100)	0.445 (0.061)
	10,000	0.522 (1.178)	0.078 (0.020)	0.210 (0.089)	0.411 (0.123)
	100,000	0.039 (0.086)	0.048 (0.060)	0.099 (0.016)	0.342 (0.124)
$\mathcal{B} = \{0.4, 0.6\}$	1,000	0.161 (0.082)	0.345 (0.117)	0.489 (0.097)	0.625 (0.057)
	10,000	0.120 (0.129)	0.279 (0.186)	0.311 (0.120)	0.546 (0.098)
	100,000	0.291 (0.202)	0.120 (0.125)	0.216 (0.118)	0.508 (0.103)
$\mathcal{B} = \{0.5, 0.5\}$	1,000	0.311 (0.202)	0.517 (0.110)	0.621 (0.052)	0.658 (0.067)
	10,000	0.254 (0.158)	0.291 (0.140)	0.434 (0.115)	0.711 (0.136)
	100,000	0.422 (0.107)	0.283 (0.146)	0.332 (0.140)	0.604 (0.049)

References

[1] Lada A Adamic et al. "Search in power-law networks". In: *Physical review E* 64.4 (2001), p. 046135.

[2] Arash A Amini et al. "Pseudo-likelihood methods for community detection in large sparse networks". In: *The Annals of Statistics* 41.4 (2013), pp. 2097–2122.

[3] Peter Chin, Anup Rao, and Van Vu. "Stochastic block model and community detection in sparse graphs: A spectral algorithm with optimal rate of recovery". In: *Conference on Learning Theory*. PMLR. 2015, pp. 391–423.

[4] Julien Chiquet, Stephane Robin, and Mahendra Mariadassou. "Variational inference for sparse network reconstruction from count data". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 1162–1171.

[5] Harry Crane and Walter Dempsey. "Edge exchangeable models for network data". In: *arXiv preprint arXiv:1603.04571* (2016).

[6] Walter Dempsey, Brandon Oselio, and Alfred Hero. "Hierarchical network models for exchangeable structured interaction processes". In: *Journal of the American Statistical Association* (2021), pp. 1–18.

[7] Karen L Fortuna et al. "Digital peer support mental health interventions for people with a lived experience of a serious mental illness: systematic review". In: *JMIR mental health* 7.4 (2020), e16460.

[8] Chao Gao et al. "Community detection in degree-corrected block models". In: *The Annals of Statistics* 46.5 (2018), pp. 2153–2185.

[9] Justin Grimmer. "A Bayesian hierarchical topic model for political texts: Measuring expressed agendas in Senate press releases". In: *Political Analysis* 18.1 (2010), pp. 1–35.

[10] Bettina Grün and Kurt Hornik. "topicmodels: An R package for fitting topic models". In: *Journal of statistical software* 40.1 (2011), pp. 1–30.

[11] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. "Stochastic blockmodels: First steps". In: *Social networks* 5.2 (1983), pp. 109–137.

[12] Brian Karrer and Mark EJ Newman. "Stochastic blockmodels and community structure in networks". In: *Physical review E* 83.1 (2011), p. 016107.

[13] Evan M Kleiman and Richard T Liu. "Social support as a protective factor in suicide: Findings from two nationally representative samples". In: *Journal of affective disorders* 150.2 (2013), pp. 540–545.

[14] Morten Mørup and Mikkel N Schmidt. "Bayesian community detection". In: *Neural computation* 24.9 (2012), pp. 2434–2456.

[15] SL van der Pas and AW38078661407 van der Vaart. "Bayesian community detection". In: *Bayesian Analysis* 13.3 (2018), pp. 767–796.

[16] Jim Pitman et al. *Combinatorial stochastic processes*. Tech. rep. Technical Report 621, Dept. Statistics, UC Berkeley, 2002. Lecture notes for ..., 2002.

[17] Karl Rohe, Sourav Chatterjee, and Bin Yu. "Spectral clustering and the high-dimensional stochastic blockmodel". In: *The Annals of Statistics* 39.4 (2011), pp. 1878–1915.

[18] Piotr Sapiezynski et al. "Interaction data from the copenhagen networks study". In: *Scientific Data* 6.1 (2019), pp. 1–10.

[19] David Strauss and Michael Ikeda. "Pseudolikelihood estimation for social networks". In: *Journal of the American statistical association* 85.409 (1990), pp. 204–212.

[20] Yee W Teh, David Newman, and Max Welling. *A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation*. Tech. rep. CALIFORNIA UNIV IRVINE SCHOOL OF INFORMATION and COMPUTER SCIENCE, 2007.

[21] Yee Whye Teh. "A Bayesian interpretation of interpolated Kneser-Ney". In: (2006).

[22] Quang H Vuong. "Likelihood ratio tests for model selection and non-nested hypotheses". In: *Econometrica: Journal of the Econometric Society* (1989), pp. 307–333.

[23] YX Rachel Wang and Peter J Bickel. "Likelihood-based model selection for stochastic block models". In: *The Annals of Statistics* 45.2 (2017), pp. 500–528.

[24] Xiaoran Yan. "Bayesian model selection of stochastic block models". In: *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE. 2016, pp. 323–328.

[25] Yubai Yuan and Annie Qu. "Community Detection with Dependent Connectivity". In: *arXiv preprint arXiv:1812.06406* (2018).

[26] Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. "Consistency of community detection in networks under degree-corrected stochastic block models". In: *The Annals of Statistics* 40.4 (2012), pp. 2266–2292.