Combining a popularity-productivity stochastic block model with a discriminative content model for detecting general structures

Bian-fang Chai,[1, 2, *] Jian Yu,[1] Cai-yan Jia,[1] Tian-bao Yang,[3] and Ya-wen Jiang[1]

[1]*School of Computer and Information Technology,*
Beijing Jiaotong University, Beijing 100044, China
[2]*Department of Information Engineering, Shijiazhuang University of Economics, Shijiazhuang, Hebei 050031,China*
[3]*GE Global Research, San Ramon, CA 94583, USA*

Latent community discovery that combines links and contents of a text-associated network, has drawn more attention with the advance of social medias. Most of the previous studies aim at detecting densely connected communities, and are not able to identify general structures, e.g., bipartite structure. Several variants based on stochastic block model are more flexible for exploring general structures by introducing link probabilities between communities. However, neither can these variants identify degree distributions of real networks due to lacking of modeling the differences among nodes, nor can they be suitable for discovering communities in text-associated networks due to ignoring the contents of nodes. In this paper, we propose a popularity-productivity stochastic block (PPSB) model by introducing two random variables, popularity and productivity, to model the differences among nodes in receiving links and producing links, respectively. The new model has the flexibility of existing stochastic block models in discovering general community structures, and inherits the richness of previous models that also exploit popularity and productivity in modeling the real scale-free networks with power law degree distributions. To incorporate contents in text-associated networks, we propose a PPSB-DC model which combines the PPSB model with a discriminative model that models the community memberships of nodes by their contents. We then develop EM algorithms for inferring the parameters in the two models. Experiments on synthetic and real networks have demonstrated that the proposed models can yield better performances than previous models, especially on networks with general structures.

I. INTRODUCTION

Recent years have seen emergence of a great volume of user generated data from online social media, e.g., Twitter, Facebook and other microblogs. It is an important task to analyze these networked data for helping people understand the structure and the function of these networks. It has been observed that networks usually exhibit a certain community structure. Therefore community detection is an important tool for analyzing the networked data [1, 2]. A common type of networked data in online social media consists of the links between nodes and the contents for describing the nodes. We refer to such networks as *text-associated networks*. A great challenge in detecting community structure in text-associated networks is how to model the community memberships, links and contents.

Many probabilistic models have been developed for community discovery by combining link and content information [3–12]. All of these models share a common framework that combines a link model and a content model. A link model defines how to generate the link probability between nodes. Stochastic block model is a popular probabilistic link model, which uses parameters to model the link probability between any two nodes that belong to the two modules ((or called community, block, group)), respectively. In order to generate a link between two nodes, it first generates a module for each node and then samples the link based on the probability associated with the two sampled modules. There are generally two categories of extensions based on the stochastic block model [13, 14]. One category models the joint link probability based on the idea of link community. They are roughly put into two kinds: and *models for general structure detection*, including mixed membership stochastic block (MMB) model [15], interaction dirichlet block model (IDBM) [16], and general stochastic block (GSB) model [17]; *models for traditional community detection*, including simple probabilistic algorithm for community detection employing Expectation-Maximization (SPAEM) [18], Bayesian link model [19], generative model for link community [20], popularity and productivity link (PPL) model [21] and etc [22–26]. The second category aims to model the conditional link probability that given a node how likely it will link to another node, and focuses on detecting traditional community in citation networks. The representative models include probabilistic HITS (PHITS) model [27], probabilistic conditional link (PCL) model [3]. The PCL model (and its generalization, the PPL model) introduces random variables of popularity (and productivity) to model the differences of nodes in receiving links (and in producing links), which are motivated by the observation that real networks usually exhibit power law degree distributions.

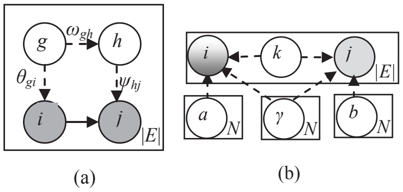
However, these link models suffer from certain shortcomings and are not suitable for modeling real networks. The conditional link models (e.g., PHITS, PCL) and the

[*] Electronic address: `chaibianfang@163.com`

joint link model (PPL) can not detect a wide variety of structures (e.g., bipartite structure), because they assume that nodes from the same community have a high chance to link to each other. Variants of the stochastic block model for general structure detection do not consider the differences among nodes in generating the links (e.g., the popularity and productivity of nodes), though they are flexible enough to detect general structures, such as assortative mixing (i.e., traditional community structure where nodes in each community are densely connected with each other), disassortative mixing (e.g., multipartite structure), and the structure of both types [28–30].

In this paper, we address these problems by proposing a popularity-productivity stochastic block (PPSB) model, which explicitly exploits the popularity and productivity of nodes to model the differences of nodes in receiving links and in producing links. The new model has the flexibility of existing stochastic block models in discovering general community structure and the richness of the PPL model in modeling the real scale-free networks with power law degree distributions. However, the proposed link model only considers links in real networks, and does not make full use of contents of nodes in text-associated networks.

A content model defines how to model the relationship between the community memberships and the contents. There exist two alternative methods for modeling the contents: generative models and discriminative models. A generative model is to model the contents from the community memberships. Probabilistic latent semantic analysis (PLSA) [31] and latent dirichlet allocation(LDA) [32] are two popular generative models. Most generative models often result in poor performance due to irrelevant features. A discriminative content model is to model the community memberships from the contents, which can alleviate the impact of irrelevant contents by weighting the content attributes Thanks to their discriminative power to the community memberships [3, 33]. The discriminative content (DC) model in a discriminative framework can be automatically to deal with unsupervised learning problems [3]. It has been observed by Yang et al. [3] that the DC model can yield substantial improvements over the generative models in terms of community detection. To incorporate the contents in the text-associated network, we also present a combined model named as PPSB-DC, which combines the PPSB model with the DC model that models the community memberships of nodes by their contents. We then develop EM algorithms for inferring the parameters in the two models. Our models can not only detect general structures, but also identify authority nodes and hub nodes. Experiments on synthetic and real networks have demonstrated that the proposed models can yield better performances than previous models, especially on networks with general structures.



(a)

(b)

FIG. 1: The plate models for network data: (a) the GSB model proposed in [17], (b) the PPL model proposed in [21]. Filled circles represent observed variables and unfilled ones correspond to latent variables. Solid lines with arrow indicate observed directed edges. Dashed lines indicate the relations between the two end variables, and arrows represent the directions of relation.

II. A LINK MODEL BASED ON STOCHASTIC BLOCK MODEL

In this section we first describe two models related to our research. Then a popularity-productivity stochastic block (PPSB) model is presented. Finally the parameters of the PPSB model are estimated by the maximum likelihood estimation method, which is implemented through the EM algorithm.

A. Related link models

In this section, we first review two related link models, i.e., the GSB model [17] and the PPL model [21], whose graphical models are shown in FIG. 1.

The GSB model is a variant of the stochastic block model based on the idea of link community, which can model the likelihood of producing links in directed or undirected networks. It assumes that each link $< i, j >$ is from a hidden community pair $< g, h >$ with probability ω_{gh}. Community g samples node i with probability θ_{gi}, and community h samples node j with probability ψ_{hj}.

The PPL model uses a joint link probability to generate the link structure of directed networks. The probability of producing a link $< i, j >$ is related to three factors: 1) the probability of community k that a link is from; 2) the probability for node i to be selected by community k, which is related to the membership γ_{ik} of node i in community k and the productivity a_i of node i; 3) the probability for node j to be selected by community k, which is related to the membership γ_{jk} of node j in community k and the popularity b_j of node j.

The generative process of each directed link in the PPL model is similar to that in the GSB model. And the parameters of the two models are learned by the EM algorithm. But there are two differences between them. First, the PPL model assumes nodes from the same community have the high probability to produce a link, while the GSB model assumes the probability of producing a link is relevant to the link probability between communities that two ends nodes of the link belong to. This difference enables the GSB model rather than the PPL model to have the capability of detecting a more general community. Second, in the PPL model the generative process of links introduces two variables, *node productivity* and *node popularity*, to explicitly capture the outgoing links and incoming links, while the GSB model does not consider these factors. This difference enables the PPL model rather than the GSB model to generate real scale-free networks with power law degree distributions. To overcome the shortcomings of the two models and make use of their advantages, we design a popularity-productivity stochastic block model (PPSB) in the following section.

B. A Popularity-productivity stochastic block model (PPSB)

In this section, some terminologies and assumptions used in our model are first introduced. Then we provide the model and its parameter estimation for a directed network.

- Let A denote the adjacency matrix of a directed network with N nodes $V = \{1, ..., N\}$ and $|E|$ directed links $E = \{< i, j > |A_{ij} \neq 0\}$.

- The *link-in* space and *link-out* space of each node i are termed as $I(i)$ and $O(i)$ respectively. $I(i) = \{j|A_{ji} \neq 0\}$, $O(i) = \{j|A_{ij} \neq 0\}$.

- Let K denote the number of communities. A block matrix is denoted as ω, normalized by the constraint $\sum_{gh} \omega_{gh} = 1$. Each element ω_{gh} is defined as the link probability of producing a directed link $< i, j >$ between any community pair $< g, h >$, where node i and node j belong to community g and community h respectively. A matrix with small off-diagonal elements and big diagonal elements can produce assortative mixing. A matrix with big off-diagonal elements and small diagonal elements can produce disassortative mixing. By changing the matrix we can generate other complex structures with assortative and disassortative mixing simultaneously.

- Let γ_{ik} denote the probability that node i belongs to community k, with a constraint $\sum_k \gamma_{ik} = 1$.

- Let a_i denote the productivity of node i, which measures how likely node i produces links. Let b_j denote the popularity of node j, which represents how likely node j receives links. They satisfy constraints $\sum_i a_i = 1$ and $\sum_j b_j = 1$.

Our PPSB model is a joint link probabilistic model for general community detection, whose graphical model is shown in (a) of FIG. 2. The PPSB model is on the idea of link community, which independently generates each link either within one community or between communities. The definition of 'community' in our model is the same as what the GSB model has defined, namely, nodes in a community have the similar connection pattern to nodes in the other community. The community here is a more general community, which contains more broad types of structures besides traditional community, e.g., bipartite structure.

Different from the PPL model, a link generated by our model can be from any two communities. The probability of generating a directed link $< i, j >$ from a community pair $< g, h >$ is quantified by element ω_{gh}, where community g and community h can be same or different. This assumption guarantees that the PPSB model takes advantage of the strength of the GSB model and overcomes the shortcoming of the PPL model in detecting general community structure.

Different from the GSB model, the probability of producing a directed link $< i, j >$ in our model considers two additional factors: the productivity of tail node i and the popularity of head node j simultaneously and explicitly. This guarantees our model possesses the advantage of the PPL model and overcomes the shortcomings of the GSB model in producing real scale-free networks with power law degree distributions.

Besides the versatility in modeling the link generative process, our model can discover both overlapping and non-overlapping communities due to its idea on link community. Link community assumes that a vertex belongs to more than one community if it has more than one type of edges. In our model the membership γ_{ik} represents the propensity of node i to have edges from community k, which provides a soft membership that node i belongs to community k and implies that node i can be overlapping. For one node, more links from one community corresponds to larger membership in that community. If we want to get the non-overlapping partition of the network, we can assign each node i to the community with the largest membership.

Followed by terminologies and assumptions, the generative process of each link in a directed network A is given as follows:

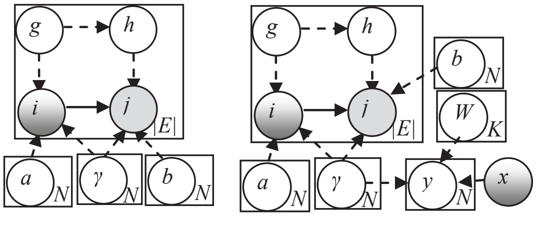
(1) Select two communities g and h for a directed link $< i, j >$ with probability ω_{gh}.

(2) Draw the tail node i from community g with probability $\frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}}$.

(3) Draw the head node j from community h with probability $\frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}}$.

According to the generative process, the marginal likelihood of the observed network A can be written as Eq. (1):

$$P(A) = \prod_{e \in E} \sum_{gh} \left(\frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}} \frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}} \omega_{gh} \right)^{A_{ij}} \tag{1}$$



(a)

(b)

FIG. 2: The plate models for network data: (a) the PPSB model, (b) the PPSB-DC model. The representation of the figure is the same as FIG. 1.

The logarithm of $P(A)$ is computed as follows:

$$L(A) = \sum_{e \in E} A_{ij} ln(\sum_{gh} (\frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}} \frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}} \omega_{gh})) \tag{2}$$

Our aim is to get the optimal parameters by the maximum likelihood estimation. In Eq. (2) a community pair $< g, h >$ of each link are hidden variables, which make parameters estimation intractable. An EM algorithm [34] is a convenient and general iterative approach to maximize the likelihood under hidden variables.

In E step, our algorithm needs to infer the community pair distribution of each link given the current parameters γ,a,b,ω. From Jensen's inequality we transform Eq. (2) into the lower bound of the log likelihood as Eq. (3), i.e., expected log likelihood.

$$\begin{aligned} \overline{L} &= \sum_e A_{ij} \sum_{gh} q_{ijgh} (ln(\frac{\frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}} \frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}} \omega_{gh}}{q_{ijgh}})) \\ &= \sum_e A_{ij} \sum_{gh} q_{ijgh} ((ln \frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}} + ln \frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}})) \\ &\quad + \sum_e A_{ij} \sum_{gh} q_{ijgh} ((ln\omega_{gh}) - ln(q_{ijgh})) \end{aligned} \tag{3}$$

By the inequality of $-logx \geq 1-x$, Eq. (3) is transformed to Eq. (4):

$$\begin{aligned} \overline{L} \geq & \sum_e A_{ij} \sum_{gh} q_{ijgh} (ln\gamma_{ig} a_i \gamma_{jh} b_j + 1 - \frac{\sum_{i'} \gamma_{i'g} a_{i'}}{\eta_g} - ln(\eta_g)) \\ & + \sum_e A_{ij} \sum_{gh} q_{ijgh} ((1 - \frac{\sum_{j'} \gamma_{j'h} b_{j'}}{\tau_h} - ln(\tau_h))) \\ & + \sum_e A_{ij} \sum_{gh} q_{ijgh} ((ln\omega_{gh}) - ln(q_{ijgh})) \end{aligned} \tag{4}$$

In Eq. (4), q_{ijgh} denotes the probability that one observes a link $< i, j >$ with i and j in communities g and h respectively, given the observed network and model parameters. η_g and τ_h can be computed as follows:

$$\eta_g = \sum_{i' \in I(j)} \gamma_{i'g} a_{i'} \qquad \tau_h = \sum_{j' \in O(i)} \gamma_{j'h} b_{j'} \tag{5}$$

Maximizing the right of Eq. (4) given the parameters, we compute the expected distribution q_{ijgh} as Eq. (6).

$$q_{ijgh} \propto \frac{\gamma_{ig} a_i}{\sum_{i' \in I(j)} \gamma_{i'g} a_{i'}} \frac{\gamma_{jh} b_j}{\sum_{j' \in O(i)} \gamma_{j'h} b_{j'}} \omega_{gh} \tag{6}$$

In M step, the algorithm optimizes the parameters given the current 'filled in' data after we get the community pair distribution of each link. We compute γ,a,b,ω as Eq. (7) through maximizing the right of Eq. (4) given the fixed community pair distribution of links.

$$\begin{aligned} \gamma_{ig} &= \frac{n(i,g)}{m_g^\eta a_i + m_g^\tau b_i} \\ a_i &= \frac{n_{out}(i)}{\sum_g m_g^\eta \gamma_{ig}} \\ b_i &= \frac{n_{in}(i)}{\sum_g m_g^\tau \gamma_{ig}} \\ \omega_{gh} &= \frac{\sum_{<i,j>} A_{ij} q_{ijgh}}{\sum_{<i,j,g,h>} A_{ij} q_{ijgh}} \end{aligned} \tag{7}$$

The rest variables in Eq. (7) are defined as:

$$\begin{aligned} n_{in}(i,g) &= \sum_{j \in I(i), h} A_{ji} q_{jihg} \\ n_{out}(i,g) &= \sum_{j \in O(i), h} A_{ij} q_{ijgh} \\ n_{in}(i) &= \sum_g n_{in}(i,g) \\ n_{out}(i) &= \sum_g n_{out}(i,g) \\ n(i,g) &= n_{in}(i,g) + n_{out}(i,g) \\ mo_g &= \sum_{<i,j> \in E, h} A_{ij} q_{ijgh} \\ mi_g &= \sum_{<i,j> \in E, h} A_{ij} q_{ijhg} \\ m_g^\eta &= \frac{mo_g}{\eta_g}, \quad m_g^\tau = \frac{mi_g}{\tau_g} \end{aligned} \tag{8}$$

We can easily extend the above model to the case of an undirected network by transferring each undirected edge (i, j) to two directed edges $< i, j >$ and $< j, i >$, and letting $a = b$.

III. A COMBINED MODEL BASED ON A DISCRIMINATIVE FRAMEWORK

In order to incorporate contents in text-associated networks, we combine the PPSB model with a discriminative content (DC) model [3] and provide a combined discriminative model, named as PPSB-DC, whose graphic model is shown in (b) of FIG. 2. The DC model is given by Eq.

(9):

$$P(z_i = r) = y_{ir} = \frac{exp(W_r^T x_i)}{\sum_l exp(W_l^T x_i)} \tag{9}$$

In Eq. (9), W_r denotes the weight vector of features for community r, whose dimension is the feature number of node i; and y_{ir} denotes the membership of node i in community r in terms of node contents.

By incorporating the DC model into the PPSB model, the log likelihood of the combined model is modified according to Eq. (2), replacing γ_{ig} with y_{ig}. A similar EM algorithm can be used to maximize the log likelihood of the combined model over the parameters W, a, b, ω. The algorithm is described as follows:

(1) Initialize W, a, b, ω;

(2) E-step: compute η, τ, q by Eq. (5) and (6), replacing γ_{ig} with y_{ig} ;

(3) M-step: compute γ, a, b, ω as in Eq. (7); and update W by multi-class logistic regression; and compute y by Eq. (9);

(4) repeat E-step and M-step until the iteration number is over threshold or the algorithm has converged..

The parameter estimation algorithm of the PPSB-DC model is derived from the EM algorithm. In E step the algorithm computes the distribution of hidden variables q, with a time complexity of $O(|E|K^2C_1)$. In M step it has a time complexity of $O(NKC_2+T)$. C_1 is a constant in computing q_{ijrs} by Eq. (6), and C_2 is a constant in computing γ_{ik}, a_i, b_i by Eq. (7). T is the time for computing y_{ir} in Eq. (9). We run M iterations to get the local optimal solution. In summary, the whole time complexity of the algorithm is $O(M(|E|K^2C_1+NKC_2+T))$. Because $|E|$ is often larger than N, the complexity can be written as $O(M(|E|K^2C_1))$. The memory cost is the space size of q, γ, a and b, i.e., $O(M(|E|K^2))$.

IV. EXPERIMENTS

In this section, we evaluate the performance of our models on several experimental studies. First we have analyzed the convergence of the PPSB model and its ability for overlapping community detection. Then the PPSB model and the PPSB-DC model are compared with several baseline models by three metrics NMI, PWF and ACC on non-overlapping community detection. In addition, the problems of model selection are discussed.

A. Data description

We use several real and synthetic networks in our experiments. Real networks in the experiments are:

- **Real undirected networks**[39]: Zachary's karate club, noted as Karate, which has 2 communities, 34 nodes and 78 edges; Dolphin network, noted as Dolphin, which has 2 communities, 62 nodes and 159 edges; American football team network, noted as Football, which has 12 communities, 112 nodes and 613 edges; Political blog network, noted as Polblogs, which has 2 communities, 1490 nodes and 19025 edges.

- **Bipartite network**: Word adjacency network [28], noted as Adjnoun, which has 112 adjectives and nouns and whose most edges connect an adjective to a noun. This network has two communities and 425 edges.

- **Text-associated directed networks**

  - **Cora Data Sets**[40] is a subset of the larger Cora citation data set. It includes 7 subcategories: Case-based reasoning, Genetic Algorithms, Neural Networks, Probabilistic Methods,Reinforcement Learning, Rule Learning and Theory, and contains 2708 nodes and 5429 links.
  - **Citeseer Data Sets**[41] is a subset of the larger Citeseer citation data set. It includes 6 subcategories: AI, Agents, DB, HCI, IR, ML, and contains 3312 nodes and 4732 links.
  - **WebKB Data Sets**[42] is a subset of the larger Webkb data set, which is classified into one of the following five classes: course, faculty, student, project, staff. It includes webpages' networks of four universities: Cornell, Texa, Washington and Wisconsin. Each school has 195, 187, 230, 265 nodes and 304, 328, 446, 530 links respectively.

According to the networks and their ground truths, we can derive the block matrix of each network, shown in FIG. 3, and Fig4. From the figures we can find that the structures of networks **Football, Cora, Citeseer** are assortative mixing, while the structure of network **Adjnoun** is disassortative mixing. The structure in each network of **WebKB Data Sets** is neither assortative mixing nor disassortative mixing singly, and we call it a mixture structure.

Our synthetic networks are generated by the growing model described by Arend Hintze and Christoph Adami [35]. It can produce a broad range of degree distribution using only a small set of parameters. Since there is only one disassortative network among real networks, we use the growing model to generate three disassortative networks to test our model better. The block matrices of the three synthetic networks are E1= $\begin{pmatrix} 0.4 & 0.6 \\ 0.7 & 0.3 \end{pmatrix}$, E2= $\begin{pmatrix} 0.2 & 0.8 \\ 0.8 & 0.2 \end{pmatrix}$, E3= $\begin{pmatrix} 0.1 & 0.9 \\ 1 & 0 \end{pmatrix}$ respectively. In the generative process of the networks, the related parameters are all set as $P_N = 0.1, p = 1, P_E = 1, q = 0.9, P_D = 0$. The network with block matrix E1 is named as **E1Net**, and the other two networks are **E2Net** and **E3Net**.

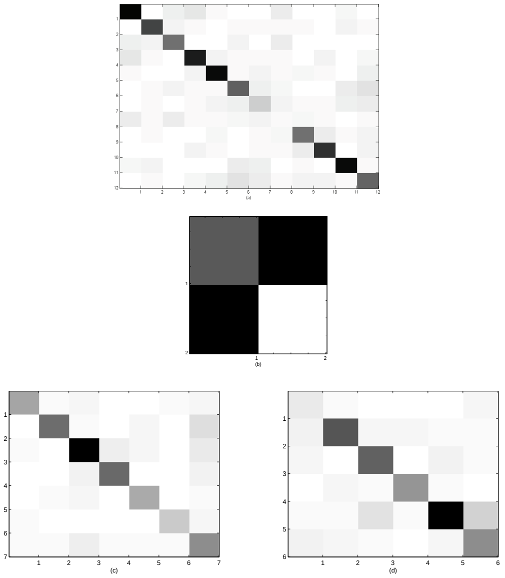


FIG. 3: Block matrixes of the networks: (a) Football, (b) Adjnoun , (c) Cora and (d) Citeseer. Each block represents the link probabilities between the corresponding community pair, and darker colors of the blocks correspond to larger link probability between the community pair.

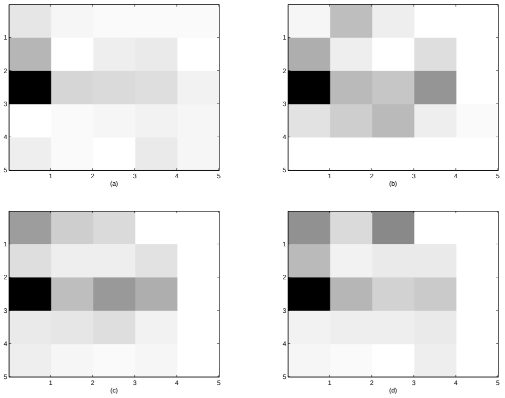


FIG. 4: Block matrixes of the networks in WebKB Data Sets: (a) Cornell, (b) Texa, (c) Washington and (d) Wisconsin. The representation of the figure is the same as Fig.3.

B. Metrics for non-overlapping community detection

In order to compare the PPSB model and the PPSB-DC model with other baseline models for community detection, the following metrics are given to measure the performance.

A community structure is $C = (C_1, C_2, ..., C_K)$, where C_k contains a set of nodes that are in the kth community. $C' = (C'_1, C'_2, ..., C'_K)$ represents a community structure given by an algorithm. Normalized mutual information is defined by:

$$NMI(C, C') = \frac{2MI(C, C')}{H(C) + H(C')} \tag{10}$$

where $H(C)$ and $H(C')$ are the entropies of the partitions C and C', and $MI(C, C')$ is the mutual information between the two partitions.

Let T denote a set of node pairs that have the same label, S denote a set of node pairs assigned to the same community by an algorithm, $|T|$ denote the cardinality of T. Pairwise F-measure is computed as follows:

$$PWF = \frac{2 \times precision \times recall}{precision + recall} \tag{11}$$

where $precision = |S \cap T|/|S|$, and $recall = |S \cap T|/|T|$.

Given a node i, its true label s_i and the assigned label r_i obtained from an algorithm, accuracy is defined as follows:

$$ACC = \frac{\sum_{i \in Nodeset} \delta(s_i, map(r_i))}{|Nodeset|} \tag{12}$$

where $|Nodeset|$ is the number of all the nodes, and $\delta(x, y)$ is a delta function that is one if $x = y$ and is zero otherwise, and $map(r_i)$ is a permutation mapping function that maps the label r_i of node i to the corresponding label in the ground truth.

For all the three metrics, i.e., NMI, PWF, and ACC, the larger the values, the better the performances.

In order to analyze how initial parameters in an EM algorithm affect final results, we use a functional modularity measure Q_H provided in [35].

$$Q_H = \frac{1}{2m} \sum_{ij} A_{ij} \widetilde{S}_{ij} \tag{13}$$

where m denotes the number of edges; $\widetilde{S}_{ij} = 1$ if i has the same community as j, otherwise $\widetilde{S}_{ij} = -\frac{1}{K-1}$. The value of Q_H changes from -1 to 1. A larger Q_H of a network existing a certain structure implies that the network underly assortative structure.

C. Convergence analysis and overlapping community detection

With a group of initial parameters, our algorithm for the PPSB model can converge to a kind of results described by a group of posterior parameters including the block link matrix ω, the memberships γ, the popularity b and the productivity a. Our algorithm is derived from an EM algorithm, and always converges to different local maxima of the log likelihood with different initial parameters. In order to approximately get a global optimal point, it is well known that the algorithm has been often run for many times with different random initial values [3, 17, 21] and the result with the largest likelihood is selected as the final solution.

Initial parameters actually have an impact on the local optimum of the algorithms. But when initial parameters are considerably complex, it is hard and even impossible to get a distribution between initial parameters and final results. But we can analyze the qualitative relations between each initial parameter and final results, and then test whether the result with the largest likelihood corresponds to the best choice. Here, we run the algorithm on our link model 20 times in four networks with assortative and disassortative mixing, including Karate, Dolphin, $E2_1$, Adjnoun. We find out that for each network the results have two kinds of outputs: one kind corresponds to assortative mixing, the other corresponds to disassortative mixing.

In order to analyze the relation between the final results and each corresponding set of initial parameters, Q_H value of the network with the set of initial memberships γ is computed, which is used as the agency of γ. It is shown that there is no rule between the Q_H value of γ and the accuracy of final results, which implies the initial parameters γ have little effect on the final results.

The relation between each initial block matrix ω and the corresponding results are also analyzed. Let π denote the sum of the diagonal values in ω, and $1-\pi$ denote the sum of the non-diagonal values. The analysis demonstrates the algorithm can converge to good results in assortative networks so long as π is larger than $1-\pi$; and in disassortative networks, the algorithm can also converge to good results so long as π is smaller than $1-\pi$. Whether the structure of the network is assortative or disassortative, the algorithm of our model can converge to good results because the block matrix can characterize broad types of structures. We also find that the results with largest or almost largest likelihoods are most approximate to the ground truth. So we can run our algorithm many times and select the results with the largest likelihood as final solution.

The model can capture the memberships γ, the popularity b and the productivity a for nodes in directed networks ($a=b$ for undirected networks). We can find which node is overlapping in terms of γ, and identify the hub nodes and authority nodes in terms of a, b. The results from the PPSB model on three networks are shown in FIG. 5,6,7. All the nodes can be identified accurately by our algorithm. Nodes in the same group are denoted by the same color (white or black). Several gray nodes are overlapping, such as 3, 9, 14, 20, 31, 32 in Karate network. The darker the gray color of the node, the larger the membership of the node in the black group, vice versa. The sizes of each node indicates its popularity or productivity, and the two values are equal for undirected networks. There is no proper measure for overlapping community detection. Thus, we analyze the performance of our models in non-overlapping case as previous studies [3, 21] in the following section.

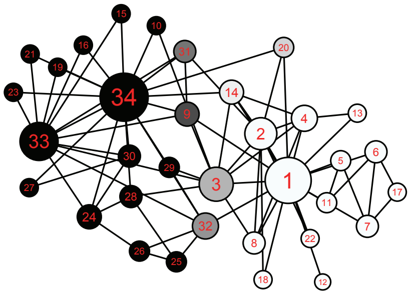


FIG. 5: (Color online)The results of PPSB model for overlapping community detection in the Karate network.

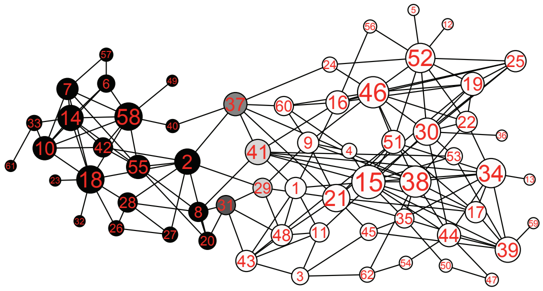


FIG. 6: (Color online)The results of PPSB model for overlapping community detection in the Dolphin network.

D. Performance of our models for non-overlapping community detection

The task of the experiments in this subsection is to detect communities based on only links and on both links and contents. To test the performance of the PPSB model, we compare it with four baseline models based on links: the GSB model [17], IDBM [16], the PCL model

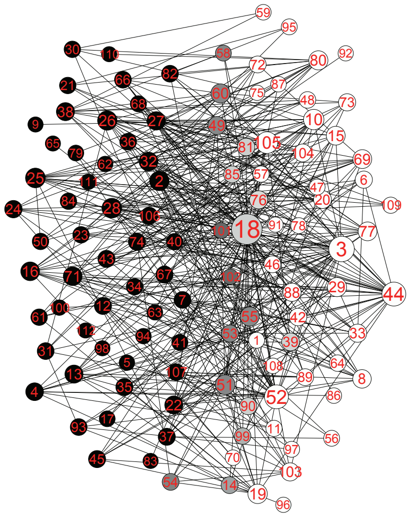


FIG. 7: (Color online)The results of PPSB model for overlapping community detection in the Adjnoun network.

[3], the PPL model [21]. Among these models, the GSB model, IDBM and our PPSB model are able to detect general community structure. The PCL model and the PPL model are both provided for traditional community detection. To test the performance of the PPSB-DC model based on both links and contents, we compare it with four combined baselines: GSB+DC, IDBM+DC, PCL-DC, PPL-DC. Since the GSB model (and IDBM) can not be unified into an EM algorithm like PCL-DC, we combine the GSB model (and IDBM) with the DC model in order, and use the partition results of the GSB model (and IDBM) as supervised assignments of the DC model. PPL-DC combines the PPL model with the DC model like PCL-DC. All the algorithms are run on both real networks and synthetic networks for community detection, and are measured by the three metrics (NMI,PWF,ACC). In order to demonstrate the combined models are better than the corresponding link models, we compare each link model and the corresponding combined model.

Since some algorithms depend on their parameters (such as α and β in IDBM, the combination coefficient in each combined models), we run on a wide range of values and choose the best one in terms of the metrics. All the algorithms on these models are run until the relative difference of the objective is within 10^{-10}. In order to test the robustness of the algorithms, we run all the algorithms for 30 times. From the convergence analysis mentioned above, we know that the results always converge to two kinds of results for the PPSB model, the GSB model, PPSB-DC and GSB+DC for assortative and disassortative networks. We can get the best results with the largest likelihood on the two kinds of networks. For networks with a mixture structure, we can also get the best results by selecting the one with the largest likelihood. Thus, we select ten results with top 10 largest objective likelihoods, which are nicely nearest to the ground truth. We then compute the average metrics for these results.

We test the non-overlapping community detection performances for all the link models on six networks. We also report the average metrics for all the models, shown in Tables I,II,III. For some assortative networks (such as Football and Polblogs), the PCL model and the PPL model outperform the models for detecting general structures. The main reason is that the PCL model and the PPL model are specifically modeled for assortative networks, while our PPSB model, the GSB model and IDBM are modeled for networks with more broad structure. Our model can get better performance than the GSB model. For disassortative networks, including Adjnoun, E1Net, E2Net, E3Net, our model achieves better results than all the other baselines based on links. This implies that it is beneficial to model the link probability by productivity and popularity explicitly in a directed network or an undirected network. Sometimes IDBM shows better results than our model, but it is unstable for all the cases. In addition, it is not easy to tune to proper parameters for its sampling algorithm. All in all, the PPSB model is a better model based on links.

TABLE I: The average NMI for link models

NMI	Football	Polblogs	Adjnoun	E1Net	E2Net	E3Net
GSB	0.8002	0.4485	0.4901	0.1834	0.3713	0.7009
PCL	0.8803	0.4517	0.0075	0.0071	0.0007	0.0006
PPL	**0.8882**	**0.5441**	0.0225	0.0402	0.0106	0.0024
IDBM	0.8743	0.0011	0.4423	0.1544	0.4121	0.6978
PPSB	0.8167	0.4538	**0.5832**	**0.1942**	**0.5019**	**0.7141**

TABLE II: The average PWF for link models

PWF	Football	Polblogs	Adjnoun	E1Net	E2Net	E3Net
GSB	0.6625	0.7388	0.7822	0.6701	0.7236	0.8724
PCL	0.8541	0.7877	0.4834	0.5152	0.5499	0.5054
PPL	**0.8652**	**0.8071**	0.5232	0.5537	0.5463	0.5443
IDBM	0.8266	0.5267	0.7670	0.5987	0.7121	0.7577
PPSB	0.7722	0.7699	**0.8498**	**0.6781**	**0.8047**	**0.8858**

We get average metrics of all combined models and link models on several text-associated networks, shown in Tables IV,V,VI. The PPSB model performs better than the other link models on networks with a mixture structure, including Cornell, Texas, Washington, Wisconsin in WebKB data sets. For Cora and Citeseer networks, the PPL model gets better results than the PPSB model. The reason is that the structures in the two data sets are assortative, which matches what the PPL model assumes. IDBM also can get better results on Cora and Citeseer networks, but it can not get the best results on the rest networks. PPSB-DC outperforms the PPSB model in all the cases, and we have equal conclusions for the other combined models and their corresponding link models. This implies the combined models considering links and contents achieve more accurate results than the models based on links. For WebKB data sets the PPSB-DC model gets better results than the other combined models. Specially, for Wisconsin network the metric value PWF and ACC of GSB+DC are better than those of PPSB-DC, but GSB+DC can not get the best result on the other data sets. So PPSB-DC is a more efficient model, especially on networks with bipartite structure and mixture structure.

E. Selecting the number of communities

When using the PPSB model and PPSB-DC, we must specify the number of communities, which is a perennial problem for all the clustering methods and controls the model complexity. Our models are probabilistic parameter models, and there are some Bayesian approaches to choose the proper number, such as Bayesian informa-

TABLE III: The average ACC(mean ± std-err) for link models

ACC	Football	Polblogs	Adjnoun	E1Net	E2Net	E3Net
GSB	0.7422 ± 0.007	0.8266 ± 0.004	0.8728 ± 0.037	0.6964 ± 0.027	0.8340 ± 0.021	0.9271 ± 0.0097
PCL	0.8881 ± 0.013	0.8377 ± 0.006	0.5478 ± 0.025	0.6189 ± 0.087	0.5202 ± 0.006	0.5117 ± 0.007
PPL	**0.8967 ± 0.006**	**0.8847 ± 0.031**	0.5784 ± 0.007	0.6206 ± 0.032	0.5465 ± 0.025	0.55257 ± 0.007
IDBM	0.8673 ± 0.035	0.4682 ± 0.008	0.8561 ± 0.009	0.7112 ± 0.006	0.8197 ± 0.021	0.8919 ± 0.005
PPSB	0.7887 ± 0.015	0.8588 ± 0.013	**0.9187 ± 0.022**	**0.7500± 0.087**	**0.8887 ± 0.016**	**0.9390 ± 0.007**

TABLE IV: The average NMI on text-associated networks

NMI	Cora	Citeseer	Cornell	Texas	Washington	Wisconsin
GSB	0.0511	0.0091	0.07173	0.0998	0.0547	0.0634
PCL	0.0893	0.0287	0.0266	0.0658	0.0519	0.0601
PPL	0.0917	**0.0349**	0.0317	0.0569	0.0524	0.0669
IDBM	**0.2610**	0.0201	**0.1685**	0.0711	0.0269	0.0727
PPSB	0.0679	0.0328	0.0825	**0.1107**	**0.1122**	**0.0788**
GSB+DC	0.2073	0.0855	0.1224	0.2658	0.0689	0.1575
PCL-DC	0.4672	0.2608	0.0834	0.0428	0.1166	0.0701
PPL-DC	**0.5164**	**0.4263**	0.0877	0.0724	0.1201	0.0871
IDBM+DC	0.2712	0.2221	0.2301	0.0369	0.0501	0.1604
PPSB-DC	0.4659	0.3870	**0.1211**	**0.3056**	**0.2391**	**0.2319**

TABLE V: The average PWF on text-associated networks

PWF	Cora	Citeseer	Cornell	Texas	Washington	Wisconsin
GSB	0.1966	0.1787	0.2601	0.3199	0.3187	0.2981
PCL	0.2087	0.1877	0.2701	0.3588	0.3062	0.2555
PPL	0.2127	**0.2088**	0.2601	0.3532	0.3089	0.2677
IDBM	**0.2987**	0.1878	0.2879	0.3301	0.2497	0.2877
PPSB	0.1904	0.1942	**0.3087**	**0.4667**	**0.3587**	**0.3287**
GSB+DC	0.2588	0.2322	0.4022	0.5701	0.2968	**0.5077**
PCL-DC	0.4898	0.3769	0.2800	0.3289	0.3299	0.2610
PPL-DC	**0.5397**	**0.5089**	0.2755	0.3488	0.3465	0.2877
IDBM+DC	0.3577	0.3335	0.3987	0.2645	0.2789	0.3432
PPSB-DC	0.5122	0.4782	**0.4768**	**0.6055**	**0.4978**	0.4212

TABLE VI: The average ACC(mean ± std-err) on text-associated networks

ACC	Cora	Citeseer	Cornell	Texas	Washington	Wisconsin
GSB	0.2422 ± 0.003	0.2219 ± 0.006	0.3144 ± 0.015	0.3788 ± 0.041	0.3728 ± 0.024	0.3588 ± 0.060
PCL	0.2855 ± 0.006	0.2428 ± 0.031	0.2553 ± 0.021	0.4001 ± 0.033	0.3417 ± 0.012	0.32593 ± 0.040
PPL	0.3011 ± 0.015	**0.2719 ± 0.021**	0.2861 ± 0.035	0.4101 ± 0.036	0.3418 ± 0.023	0.3014 ± 0.017
IDBM	**0.3979 ± 0.022**	0.2082 ± 0.013	0.3066 ± 0.016	0.3588 ± 0.014	0.2099 ± 0.044	0.3233 ± 0.044
PPSB	0.2631 ± 0.006	0.2437 ± 0.015	**0.3622 ± 0.026**	**0.5063 ± 0.012**	**0.4022 ± 0.021**	**0.3851 ± 0.032**
GSB+DC	0.3355 ± 0.035	0.2844 ± 0.014	0.4565 ± 0.021	0.5977 ± 0.030	0.3661 ± 0.021	**0.5366 ± 0.012**
PCL-DC	0.6475 ± 0.011	0.4965 ± 0.015	0.3278 ± 0.014	0.3755 ± 0.022	0.4112 ± 0.023	0.2998 ± 0.015
PPL-DC	**0.6623 ± 0.021**	**0.6188 ± 0.013**	0.3299 ± 0.008	0.4001 ± 0.017	0.4241 ± 0.003	0.3464 ± 0.036
IDBM+DC	0.4376 ± 0.023	0.4454 ± 0.015	0.4965 ± 0.020	0.3078 ± 0.044	0.3199 ± 0.028	0.4100 ± 0.018
PPSB-DC	0.6196 ± 0.030	0.5878 ± 0.025	**0.5365 ± 0.010**	**0.6295 ± 0.015**	**0.5710 ± 0.012**	0.4953 ± 0.016

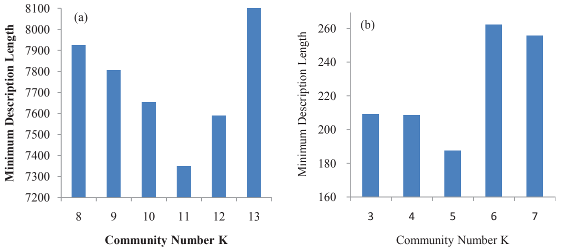


FIG. 8: (Color online)The results of model selection for the networks: (a) Football and (b) Cornell.

tion criterion and Akaike information criterion, which are unsuitable because of too many zero parameters in our models. Minimum description length principle(MDL) is based on the simple idea that the best way to capture regular features in data is to construct a model in a certain class which permits the shortest description of the data and the model itself [17, 18, 36]. MDL is a powerful and easy method for model selection, and is used here. We just provide the results of the PPSB model on the two networks: Football and Cornell. For the PPSB-DC and the other networks the method is similar.

The log likelihood of L in Eq. (2) increases as K increases, but at the same time the number of the parameters also increases. Based on the idea of MDL principle, the code length to describe the network data includes two parts: the coding length of the network in the PPSB model and the coding length of the model parameters. The former is $-L$ for directed networks and $-L/2$ for undirected networks. The latter is $-\sum_{gh}\omega_{gh}ln\omega_{gh} - \sum_{ig}\gamma_{ig}ln\gamma_{ig} - \sum_{i}(a_i lna_i + b_i lnb_i)$ for directed networks, and $-\sum_{gh}\omega_{gh}ln\omega_{gh} - \sum_{ig}r_{ig}lnr_{ig} - \sum_{i}(a_i lna_i)$ for undirected networks. We select one popular fast method for community detection that need not offer the K value, and run it so that we can get the prior knowledge of the number [37]. According to the prior of the number, we select the optimal K^* that minimizes the total description length over a potentially large number of models.

Two real networks are detected over a set of values of K. One is the American football team network, an undirected network, which is identified 11 communities by many methods. The other one is the Cornell network in WebKB Data Set, a directed network with 5 communities. We aim at selecting K^* which minimizes the MDL value. As shown in FIG. 8, this criterion is valid for selecting a proper number of communities from a wide range of K.

V. CONCLUSIONS

In this paper, a variant of stochastic block model (named as PPSB) is designed to detect more general structures in real and synthetic networks. Compared with the existing variants based on stochastic block model, the PPSB model considers more sufficient factors in the generative process of the network making it generate more practical networks with power law degree distributions. Compared with the existing probabilistic models that also model popularity and productivity, our PPSB model can detect more general structures. The memberships of nodes captured from the PPSB model makes it easily be combined with a discriminative content model which generates the community memberships from contents in text-associated networks. Tests for overlapping and non-overlapping structure detection on synthetic and real networks have demonstrated our models can get better results, especially on networks with other types of structures such as bipartite structure and mixture structure.

We also give a solution for selecting a plausible number of groups based on minimum description length principle. But it also requests a prior knowledge of the group number by running existing fast community detection methods. In the future we plan to use some Bayesian nonparametric models to provide a better way for the problem of model selection [38], and utilize our model on more real-world scenarios, e.g., automatic recommendation for online users.

Acknowledgments

This work was supported by the National Science Foundation (Grant No. 61033013), Beijing Natural Science Foundation (Grant No. 4112046), the Fundamental Research Funds for the Central Universities (Grant No. 2012YJS027), the Hebei province Science and Technology Planed Projects (Grant No. 13210702D). The authors appreciate Huaiyu Wan for their help in revising this work carefully. The author would also acknowledge Arend Hintze for code sharing, and thank the anonymous reviewers and the editorial staff of this journal for their constructive comments .

References

[1] S. Fortunato, Phys. Rep. **486**, 75 (2010).

[2] S. E. Schaeffer, Computer Science Review **1**, 27 (2007).

[3] T. Yang, R. Jin, Y. Chi, and S. Zhu, in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, 2009) pp. 927–936.

[4] T. Yang, R. Jin, Y. Chi, and S. Zhu, in *Proceedings of the Conference on Uncertainty in Artificial Intelligence* (AUAI Press, 2009) pp. 615–622.

[5] D. Li, Y. Ding, X. Shuai, J. Bollen, J. Tang, S. Chen, J. Zhu, and G. Rocha, Journal of Informetrics **6**, 237 (2012).

[6] R. Balasubramanyan and W. W. Cohen, in *Proceeding of the 7th SIAM International Conference on Data Mining* (Phoenix, AZ, 2011) pp. 450–461.

[7] Z. YIN, L. CAO, Q. GU, and J. HAN, ACM Transactions on Intelligent Systems and Technology **3**, 63 (2012).

[8] D. Hofmann, Advances in Neural Information Processing Systems(NIPS) **13**, 430 (2000).

[9] E. Erosheva, S. Fienberg, and J. Lafferty, Proceedings of the National Academy of Sciences of the United States of America **101**, 5220 (2004).

[10] R. Nallapati, A. Ahmed, E. Xing, and W. Cohen, in *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, 2008) pp. 542–550.

[11] M. Kim and J. Leskovec, in *International Conference on Machine Learning* (2012).

[12] D. Duan, Y. Li, R. Li, Z. Lu, and A. Wen, The Computer Journal **56**, 336 (2013).

[13] K. Nowicki and T. Snijders, Journal of the American Statistical Association **96**, 1077 (2001).

[14] T. Snijders and K. Nowicki, Journal of Classification **14**, 75 (1997).

[15] E. Airoldi, D. Blei, S. Fienberg, and E. Xing, J. Mach. Learn. Res. **9**, 1981 (2008).

[16] Á. Gyenge, J. Sinkkonen, and A. Benczúr, in *Proceedings of the Eighth Workshop on Mining and Learning with Graphs* (ACM, Washington, D.C., 2010) pp. 62–69.

[17] H. Shen, X. Cheng, and J. Guo, Phys. Rev. E **84**, 056111 (2011).

[18] W. Ren, G. Yan, X. Liao, and L. Xiao, Phys. Rev. E **79**, 036111 (2009).

[19] J. Parkkinen, J. Sinkkonen, A. Gyenge, and S. Kaski, in *Proceedings of the 7th International Workshop on Mining and Learning with Graphs* (2009).

[20] B. Ball, B. Karrer, and M. Newman, Phys. Rev. E **84**, 036103 (2011).

[21] T. Yang, Y. Chi, S. Zhu, Y. Gong, and R. Jin, in *SIAM International Conference on Data Mining* (2010) pp. 742–753.

[22] J. Daudin, F. Picard, and S. Robin, Statistics and computing **18**, 173 (2008).

[23] H. Zanghi, C. Ambroise, and V. Miele, Pattern Recognition **41**, 3592 (2008).

[24] M. Hastings, Phys. Rev. E **74**, 035102 (2006).

[25] J. Hofman and C. Wiggins, Phys. Rev. Lett. **100**, 258701 (2008).

[26] B. Karrer and M. Newman, Phys. Rev. E **83**, 016107 (2011).

[27] D. Cohn and H. Chang, in *Proceedings of the 17th International Conference on Machine Learning* (Citeseer, 2000) pp. 167–174.

[28] M. E. Newman, Phys. Rev. E **74**, 036104 (2006).

[29] M. E. J. Newman, Phys. Rev. E **67**, 026126 (2003).

[30] M. Newman and E. Leicht, Proceedings of the National Academy of Sciences **104**, 9564 (2007).

[31] T. Hofmann, in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (Morgan Kaufmann Publishers Inc., 1999) pp. 289–296.

[32] D. M. Blei, A. Y. Ng, and M. I. Jordan, J. Mach. Learn. Res. **3**, 993 (2003).

[33] S. Lacoste-Julien, F. Sha, and M. I. Jordan, Advances in Neural Information Processing Systems **21** (2008).

[34] A. P. Dempster, N. M. Laird, and D. B. Rubin, Joural of the Royal Statistical Society, Series B **39**, 1 (1977).

[35] A. Hintze and C. Adami, Biology Direct **5**, 32 (2010).

[36] J. Rissanen, Automatica **14**, 465 (1978).

[37] M. Rosvall and C. T. Bergstrom, Proceedings of the National Academy of Sciences **105**, 1118 (2008).

[38] S. J. Gershman and D. M. Blei, Journal of Mathematical Psychology **56**, 1 (2012).

[39] http://www-personal.umich.edu/~mejn/netdata/

[40] http://vmwxs.umd.edu/projects/linqs/projects/lbc/index.html

[41] http://citeseer.ist.psu.edu/

[42] http://www-2.cs.cmu.edu/ webkb/