Under consideration for publication in Knowledge and Information Systems

Efficient Nonparametric and Asymptotic Bayesian Model Selection Methods for Attributed Graph Clustering

Zhiqiang Xu[1], James Cheng[2], Xiaokui Xiao[3], Ryohei Fujimaki[4]

Yusuke Muraoka[5]

[1]Division of CEMSE, King Abdullah University of Science and Technology, Saudi Arabia
[2]Department of Computer Science and Engineering, The Chinese University of Hong Kong, China
[3]School of Computer Engineering, Nanyang Technological University, Singapore
[4]NEC Laboratories America, [5]NEC Laboratories Japan

Abstract. Attributed graph clustering, also known as community detection on attributed graphs, attracts much interests recently due to the ubiquity of attributed graphs in real life. Many existing algorithms have been proposed for this problem, which are either distance-based or model-based. However, model selection in attributed graph clustering has not been well addressed, that is, most existing algorithms assume the cluster number to be known a priori. In this paper, we propose two efficient approaches for attributed graph clustering with automatic model selection. The first approach is a popular Bayesian nonparametric method, while the second approach is an asymptotic method based on a recently proposed model selection criterion, factorized information criterion (FIC). Experimental results on both synthetic and real datasets demonstrate that our approaches for attributed graph clustering with automatic model selection, significantly outperform the state-of-the-art algorithm.

Keywords: Attributed graph clustering; Model selection; Dirichlet process; Factorized information criterion

1. Introduction

Given a graph with attributed nodes[1], *attributed graph clustering* (also known as community detection (Newman and Girvan, 2004) on attributed graphs) aims

Received Oct 08, 2015
Revised Aug 24, 2016
Accepted Jan 14, 2017

[1] i.e., we consider only node-attributed graphs throughout the paper.

to partition the attributed nodes into disjoint subsets (referred to as clusters or communities), such that intra-cluster nodes have both dense connections in structure and low diversity in attribute values, while inter-cluster nodes are sparsely connected and may have diverse attribute values (Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014). Different from clustering methods that are solely structure-based or attribute-based, attributed graph clustering tries to retain as much structure information as possible, while attaining as high attribute homogeneity as possible in the resultant clusters.

Attributed graph clustering has attracted increasing interests in recent years (Ester et al., 2006; Zhou, Cheng and Yu, 2009; Zanghi et al., 2010; Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014) for its capability to detect more informative clusters by combining both structure and attribute information. It has also become increasingly important due to the ubiquity of attributed graphs in real life and its many applications, such as online social network application, telecommunication application (Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014), web application and research collaboration study. There exists a number of algorithms for attributed graph clustering (Ester et al., 2006; Zhou, Cheng and Yu, 2009; Zanghi et al., 2010; Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014). However, most existing algorithms assume that the number of clusters is known a priori, which is not the case in most real-world applications (Banerjee et al., 2015). For example, the segment number in image segmentation (Semertzidis et al., 2015) or the face number in face clustering (Vretos et al., 2011) is hard to specify for complex images or applications to a large collection of images. To determine an appropriate cluster number, these existing algorithms need to be run from scratch many times for a group of candidate cluster numbers, and then choose the best one according to a certain criterion, e.g., AIC or BIC (Bishop, 2006). This process clearly aggravates the computational burden, especially for clustering large attributed graphs. Alternatively, one may use cross validation on the held-out data to address the problem, but it wastes a part of data for learning and also requires considerable extra running time (Yang et al., 2013).

The above discussion shows that the determination of cluster number, or formally the model selection problem, is an important practical problem for attributed graph clustering. However, the problem has not been thoroughly studied in the literature. In particular, we are only aware of two recent works on the problem, namely, PICS (Akoglu et al., 2012), and JointClust (Moser et al., 2007). PICS may produce clusters of the vertices that favor connections between dissimilar nodes (disassortative mixing (Newman, 2002)), and thus, it is not appropriate for community detection in many real applications since it requires sparse connections between dissimilar nodes. JointClust requires the input graph to be connected and takes the time quadratic in the number of vertices, which makes it impractical.

In this paper, we propose two model-based approaches (referred to as the nonparametric approach and the asymptotic approach, respectively) for attributed graph clustering with automatic determination of cluster number. Our approaches target simultaneous model selection and inference, eliminating the need for multiple runs of the clustering algorithm or cross validation. In addition, they are efficient and do not require the input graph to be connected.

For both approaches, we follow the probabilistic generative model for attributed graph clustering adopted in (Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014) and focus on assortative mixing in graph (Newman, 2002) (assor-

tative mixing is consistent with the definition of attributed graph clustering). Without loss of generality, we assume that the edges are un-weighted and the attributes are categorical. Accordingly, the generative model (on which our approaches are based) consists of stochastic block model with binary observations and latent class model with multivariate categorical observations. The stochastic block model is a finite mixture model for graph structure data (Nowicki and Snijders, 2001), while the latent class model is a special class of the finite mixture model for categorical attribute data, which assumes local independence of categorical variables conditioned on the class memberships (Lazarsfeld and Henry, 1968).

To solve the model selection problem, the nonparametric approach uses the popular Dirichlet process (Teh, 2010) from nonparametric Bayesian statistics. For our model described above, we only need to assign a stick-breaking prior (Teh, 2010) to the cluster proportion vector, which extends cluster number from finite to infinite. In practice, we work with the finite approximation by truncating the stick-breaking prior to facilitate variational inference, and shrink the parameter space gradually during inference to achieve the automatic model selection.

The asymptotic approach is based on a recently proposed model selection criterion, namely, factorized information criterion (FIC) (Fujimaki and Morinaga, 2012). FIC is an extension of BIC (Bishop, 2006) that can be used for non-regular models[2], e.g., mixture models. By Laplace approximating each component in the factorized representation of complete data likelihood, FIC naturally induces a regularization term pertaining to cluster indicators, which plays a crucial role in shrinking parameter space. It has achieved very promising results on vectorial data (Fujimaki and Morinaga, 2012) and temporal data (Fujimaki and Hayashi, 2012). To the best of our knowledge, FIC has not yet been applied to attributed graphs. Since it induces a simple and effective algorithm over mixture models and the mechanism of model selection is easy to understand, deriving an attributed graph clustering algorithm based on FIC will provide a good alternative to the nonparametric approach.

We transform the attributed graph clustering problem into a probabilistic inference problem for simultaneous parameter estimation and determination of the cluster number. To find the solution, we employ variational Bayesian inference for the nonparametric approach, and variational EM for the asymptotic approach. Since the shrinkage of parameter space is embedded into iterative variational inference process, our methods work significantly more efficiently than existing algorithms without automatic model selection for attributed graph clustering. As we will show later, both of our approaches are efficient since they both have time and space complexity linear in the number of edges, vertices and the number of attributes.

We evaluate our approaches using both synthetic and real datasets. Compared with the state-of-the-art algorithm for attributed graph clustering with automatic model selection, PICS (Akoglu et al., 2012), both of our algorithms achieve significantly higher clustering quality, in terms of both structural and attribute information. Our algorithms are consistently faster than PICS for all datasets. In addition, the number of clusters obtained by our algorithms seems more reasonable than that obtained by PICS.

[2] Non-regular models refer to the models that do not satisfy regularity conditions with BIC (Bishop, 2006)

The rest of the paper is organized as follows. Section 2 introduces related work. Section 3 defines the problem of attributed graph clustering with automatic model selection. Section 4 introduces the Bayesian model for attributed graph clustering on which our approaches are based. Section 5 presents our Bayesian nonparametric approach for model selection in attributed graph clustering. Section 6 presents our asymptotic approach. Section 7 reports the experimental results. Finally, Section 8 concludes the paper.

2. Related Work

Existing algorithms on attributed graph clustering can be naturally divided into two categories as follows in our context.

2.1. Algorithms without Model Selection

Most existing works on attributed graph clustering assume that the cluster number is known a priori, which is not a realistic assumption for many real-world applications and real datasets. These approaches are either distance-based or model-based, which we discuss as follows.

The main idea of distance-based approaches (Steinhaeuser and Chawla, 2008; Zhou, Cheng and Yu, 2009) is to design a distance/similarity measure for vertex pairs that combines edge connection and vertex attribute values to construct a new non-attributed graph. Then, clustering algorithms, such as spectral clustering (Ng et al., 2001) via the eigen-decomposition (Xu et al., 2016; Xu and Ke, 2016*b*), for non-attributed graphs (i.e., structure only) are applied to this new graph. For example, Zhou et al. proposed a random walk distance measure over the augmented graph which is constructed by linking the vertices with certain common attribute value to an artificial node specifically representing the value (Zhou, Cheng and Yu, 2009). Then K-medoids algorithm is applied to find the clustering.

Instead of artificially designing a distance/similarity measure, the model-based approaches (Zanghi et al., 2010; Henderson et al., 2010; Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014; Yang et al., 2013) take a joint modeling of the interplay between edge connection and vertex attributes, and make use of this model to infer the optimal clustering. For example, Zanghi et al. proposed a non-Bayesian probabilistic model consisting of SBM (Nowicki and Snijders, 2001) and GMM (Bishop, 2006), with the cluster indicator shared (Zanghi et al., 2010). Xu et al. proposed a similar model, BAGC, consisting of SBM and LCM (Lazarsfeld and Henry, 1968), but took a full Bayesian approach (Xu et al., 2012; Xu, Ke, Wang, Cheng and Cheng, 2014). Yang et al. proposed the CESNA model for overlapping community detection in networks with node attributes based on affiliation network models (Yang et al., 2013). It is a probabilistic generative model in which the community affiliations are used to model both network data and node attribute data, and solved by alternating convex optimization. CESNA aims at overlapping clustering and only handles binary attributes, while our problem considers non-overlapping clustering and works with categorical attributes. Another class of model-based approaches that are remotely related to attributed graph clustering are the works on community detection in World Wide Web and citation networks (Nallapati et al., 2008; Yang et al., 2009; Sun et al., 2012; Xu

and Ke, 2016*a*), which strive to take advantage of the unstructured text data to improve clustering quality. In contrast, our work focuses on structured attributes.

There are also heuristic-based approaches. For example, Ester et al. proposed the Connected k-Center (CkC) problem for graphs in which each vertex is associated with a coordinate vector, to group vertices into k clusters such that the maximum distance of any vertex to its corresponding cluster center is minimized and meanwhile the subgraph induced by each cluster is connected (Ester et al., 2006). Since CkC is NP-hard, a heuristic algorithm was proposed.

For more general knowledge on attributed graph clustering including the cases of node-attributed or edge-attributed graphs, readers can refer to one very recent paper by Bothorel et al. which provides a literature review on the models, metrics and methods of clustering attributed graphs (Bothorel et al., 2015). Besides, Papadopoulos et al. considered clustering attributed multi-graphs where graph nodes are associated with attributes and edges have different types (Papadopoulos et al., 2015). Zongqing Lu et al. proposed a community detection algorithm from the perspective of designing network protocols where node attributes were not considered (Lu et al., 2015).

However, none of aforementioned methods studied the model selection problem in attributed graph clustering.

2.2. Algorithms with Model Selection

We are only aware of two existing works that are most related to our case in this paper, JointClust proposed by Moser et al. (Moser et al., 2007), and PICS proposed by Akoglu et al. (Akoglu et al., 2012). Both of them are able to perform attributed graph clustering without specification of cluster number. Nevertheless, the three works, i.e., JointClust, PICS and ours (NBAGC and FABAGC), differ from each other as follows. JointClust is a heuristic-based algorithm consisting of two phases. The first phase determines cluster atoms, which are then merged in a bottom-up manner based on the Joint Silhouette Coefficient in the second phase. The clustering with the highest Joint Silhouette Coefficient is returned from the generated dendrogram at last. JointClust requires the input graph to be connected and takes a quadratic running time $O(N^2)$, which makes it impractical for clustering real or large datasets. NBAGC and FABAGC do not restrict the input graph on connectivity and take the running time linear in the number of edges $O(\text{nnz}(\mathbf{X}))$, and thus are more efficient and practical. PICS casts the co-clustering problem of vertices and attributes as a data compression task, where each cluster is treated as a compression of a cohesive subset of nodes that exhibit both similar connectivity patterns and high attribute homogeneity. They developed a matrix-based data compression model, and applied the minimum description length (MDL) principle to find the optimal number of clusters and co-clustering. PICS only handles binary attributes, and favors both assortative and disassortative mixings in graph structure. In this paper, we compared our algorithms with PICS and as evidenced by the experimental results, our approaches and PICS achieve different tradeoffs between structure and attribute quality in clustering, but our approaches are orders of magnitude faster than PICS.

In addition, Luo reviewed the automatic selection methods for machine learning algorithms and hyper-parameter values while our algorithm belongs to the case of selecting hyper-parameter values (i.e., cluster number) for a given ma-

chine learning algorithm (Luo, 2015). Henderson et al. proposed a model selection method for automatically finding the number of node roles in a network (Henderson et al., 2012). They applied the MDL principle as in PICS. But the greedy search was performed there to minimize the coding cost while PICS used the randomized search.

3. Problem Statement

An attributed graph G is defined as a 4-tuple (V, E, Λ, F), where $V = \{v_1, \cdots, v_N\}$ is the vertex set of size N, $E \subset V \times V$ is the edge set, $\Lambda = \{a_1, \ldots, a_T\}$ is a set of T categorical attributes with $dom(a_t) = \{a_{t1}, \cdots, a_{tM_t}\}$ for each t, and F is a matrix where the i-th row $F_i \in dom(a_1) \times \cdots \times dom(a_T)$ is the attribute vector associated with vertex v_i. We restrict our discussions on *undirected* attributed graphs, but our method can be easily extended to process directed attributed graphs.

Given an attributed graph G, the clustering problem studied in this paper is to partition the vertex set V of G into an appropriate number, say K, of disjoint subsets $V_1, V_2, \ldots, V_K$, where $V = \bigcup_{i=1}^{K} V_i$ and $V_i \cap V_j = \emptyset$ for any $i \neq j$, such that within-cluster vertices are densely connected and have low diversity in their attribute values, while between-cluster vertices are sparsely connected and may have diverse attribute values.

4. Preliminaries

In this section, we introduce a Bayesian model for attributed graph clustering, i.e., BAGC (Xu et al., 2012), upon which our model selection approaches are built in Section 5 and 6. We start with notions and notations, and then present the underlying model specification.

4.1. Notions and Notations

Given an attributed graph G described in Section 3, let $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ represent the *adjacency matrix*, *attribute matrix* and *clustering of vertices*, respectively. Specifically, $\mathbf{X} = [\mathbf{X}_{ij}]$ is an $N \times N$ zero diagonal symmetric random matrix[3] with $\mathbf{X}_{ij} \in \{0, 1\}$, $\mathbf{Y} = [\mathbf{Y}_{it}]$ an $N \times T$ random matrix with $Y_{it} \in dom(a_t)$, and $\mathbf{Z} = [\mathbf{Z}_i]$ an $N \times 1$ random vector with $\mathbf{Z}_i \in \{1, 2, \ldots, K\}$ denoting the cluster label of vertex v_i. In other words, the given graph (V, E, Λ, F) and the associated attributes F are realizations of $\mathbf{X}$ and $\mathbf{Y}$ dictated by the BAGC model, respectively.

4.2. Model Specification

For BAGC, the structure data $\mathbf{X}$ is assumed to be generated by the stochastic block model (SBM) where a graph can be treated as a realization of the random

[3] The zero diagonal of $\mathbf{X}$ means no self-loops in the corresponding graph while symmetry means that the graph is undirected, in accordance with our focus on undirected simple graphs.

graph characterized by the Erdos-Renyi Mixture Model (Nowicki and Snijders, 2001). Specifically, for each vertex $v_i \in V$ its cluster label $\mathbf{Z}_i$ can be generated from a multinomial distribution π, while for each pair of vertices v_i and v_j $(i < j)$ $\mathbf{X}_{ij}$ follows the Bernoulli distribution $\phi_{\mathbf{Z}_i\mathbf{Z}_j}$ conditioned on their cluster labels. That is,

$$p(\mathbf{Z}|\pi) = \prod_{i=1}^{N} p(\mathbf{Z}_i|\pi), \tag{1}$$

$$p(\mathbf{X}|\mathbf{Z},\phi) = \prod_{i<j} p(\mathbf{X}_{ij}|\phi_{\mathbf{Z}_i\mathbf{Z}_j}) \tag{2}$$

where

$$p(\mathbf{Z}_i|\pi) = \prod_k \pi_k^{\mathbf{Z}_{ik}}, \quad p(\mathbf{X}_{ij}|\phi_{kl}) = \phi_{kl}^{\mathbf{X}_{ij}}(1-\phi_{kl})^{1-\mathbf{X}_{ij}}.$$

Here π represents the size proportions of all the clusters and $\sum_k \pi_k = 1$, while ϕ_{kl} represents the proportion of edges among all the pairs of vertices from cluster k to cluster l $(k \neq l)$ or within one cluster k $(k = l)$. Note that the cluster label $\mathbf{Z}_i$ is represented by an integer (e.g., $\phi_{\mathbf{Z}_i\mathbf{Z}_j}$) or a one-of-K vector (e.g., $\mathbf{Z}_{ik} = \delta(\mathbf{Z}_i, k)$) interchangeably, which can be understood from the context easily.

On the other hand, the attribute data $\mathbf{Y}$ can be generated by the latent class model (LCM) (Lazarsfeld and Henry, 1968). LCM assumes that different attributes of a vertex conditioned on its cluster label, namely, $\mathbf{Y}_{it}|\mathbf{Z}_i$ for $t = 1, \cdots, T$, are independent (i.e., the so-called local independence), and for each attribute $\mathbf{Y}_{it}|\mathbf{Z}_i$ follows the multinomial distribution. That is,

$$p(\mathbf{Y}|\mathbf{Z},\theta) = \prod_{i=1}^{N} p(\mathbf{Y}_i|\theta_{\mathbf{Z}_i}), \tag{3}$$

where

$$p(\mathbf{Y}_i|\theta_{\mathbf{Z}_i}) = \prod_{t=1}^{T} p(\mathbf{Y}_{it}|\theta_{\mathbf{Z}_i}^{(t)}), \quad p(\mathbf{Y}_{it}|\theta_k^{(t)}) = \prod_{m_t=1}^{M_t} (\theta_{km_t}^{(t)})^{\delta(\mathbf{Y}_{it}, a_{tm_t})}.$$

Here $\delta(a,b) = 1$ if $a = b$ otherwise 0, and $\sum_{m_t} \theta_{km_t}^{(t)} = 1, 0 < \theta_{km_t} < 1$.

The key of combining two types of information for one clustering is to couple the two submodels together by sharing the same cluster labels $\mathbf{Z}$. Finally, BAGC treats the model parameters as random variables as well and places a conjugate prior over them. Additionally, we suppress the disassortative mixing in graph structure that favors connections between different clusters (Newman, 2002), such that the resulting clustering is more consistent with our definition[4] in Section 3. This can be accomplished by fixing a small inter-cluster edge proportion ε (we use $\varepsilon = 10^{-6}$ throughout the paper). Specifically, we have that[5]

$$\pi \sim \text{Dirichlet}(\alpha),\ \theta_k^{(t)} \sim \text{Dirichlet}(\beta^{(t)}) \text{ and } \phi_{kk} \sim \text{Beta}(\gamma_1, \gamma_2).$$

[4] The definition of our clustering requires as less edges as possible between distinct clusters.
[5] Multinomial and Dirichlet distributions are conjugate. As a special case, Bernoulli and Beta distributions are conjugate as well.

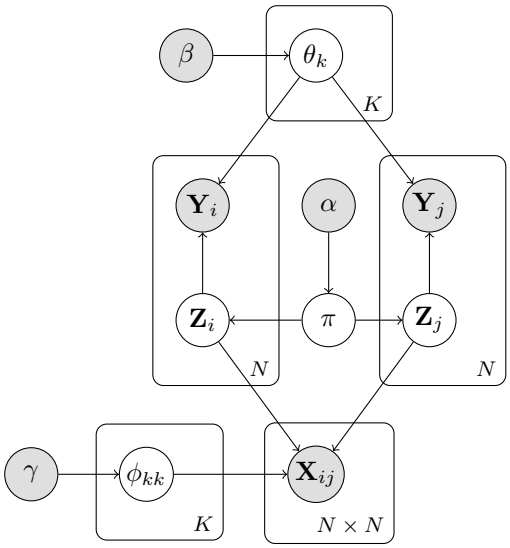


Fig. 1. Graphical representation of BAGC.

That is,

$$p(\pi|\alpha) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \pi_k^{\alpha_k - 1},$$

$$p(\theta|\beta) = \prod_t \prod_k p(\theta_k^{(t)}|\beta^{(t)}), \qquad p(\theta_k^{(t)}|\beta^{(t)}) = \frac{\Gamma(\sum_{m_t} \beta_{m_t}^{(t)})}{\prod_{m_t} \Gamma(\beta_{m_t}^{(t)})} \prod_{m_t} (\theta_{km_t}^{(t)})^{\beta_{m_t}^{(t)} - 1},$$

$$p(\phi|\gamma) = \prod_k p(\phi_{kk}|\gamma) \prod_{k \neq l} \delta(\phi_{kl}, \varepsilon), \qquad p(\phi_{kk}|\gamma) = \frac{\Gamma(\gamma_1 + \gamma_2)}{\Gamma(\gamma_1)\Gamma(\gamma_2)} \phi_{kk}^{\gamma_1 - 1} (1 - \phi_{kk})^{\gamma_2 - 1}.$$

Note that since ϕ_{kl} $(k \neq l)$ are fixed to ε throughout the paper, we can write $p(\phi|\gamma) = \prod_k p(\phi_{kk}|\gamma)$ for brevity. Figure 1 shows the graphical representation of BAGC which encodes the dependence relations between variables.

5. Nonparametric Approach

The first approach for model selection in attributed graph clustering is based on Bayesian nonparametrics and called nonparametric Bayesian attributed graph clsutering (NBAGC). For model selection in the mixture modeling, one popular approach from the nonparametric Bayesian statistics is to use the Dirichlet process, which provides a prior for mixture parameters such that the component number can grow with data and automatically get inferred from data as well. In particular, Dirichlet process is a generalization of Dirichlet distribution, e.g., $p(\pi|\alpha)$ in Section 4.2, and can be constructed by stick-breaking process (Teh, 2010).

Next we will plug the Dirichlet process prior into the BAGC model to obtain a new model termed NBAGC in Section 5.1, and then resolve the corresponding

inference problem in Section 5.2 and last analyze the computational complexity in Section 5.3.

5.1. Stick-Breaking Prior

For our problem, we can replace the Dirichlet distribution $\mathrm{Dir}(\alpha)$ in the BAGC model with a stick-breaking distribution[6] $\mathrm{GEM}(\alpha)$ (Teh, 2010), i.e.,

$$\pi_k = u_k \prod_{l=1}^{k-1}(1-u_l),\ u_k \sim \mathrm{Beta}(1,\alpha) \quad \text{where } k = 1, 2, \cdots, \infty,$$

as a prior over the cluster proportion π. To see that Dirichlet process is a generalization of Dirichlet distribution $\mathrm{Dir}(\alpha)$, we only need to set $u_K = 1$ and then have $\sum_{k=1}^{K} \pi_k = 1$ (Teh, 2010). However, different from directly sampling from $\mathrm{Dir}(\alpha)$, the K-dimensional π sampled this way results in a size-biasing one (Teh, 2010; Kurihara et al., 2007).

Since π now is dependent on u, we can directly write $p(\mathbf{Z}|\pi) = p(\mathbf{Z}|u)$. Then the joint log likelihood can be written as

$$\begin{aligned}&\log p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \phi, \theta, u|\alpha, \beta, \gamma)\\ =&\log[p(\mathbf{X}|\mathbf{Z}, \phi)p(\phi|\gamma)\big] + \log[p(\mathbf{Y}|\mathbf{Z}, \theta)p(\theta|\beta)] + \log[p(\mathbf{Z}|u)p(u|\alpha)].\end{aligned}$$

In practice, a mixture model fitting the given data has only a finite component number and thus we approximate the underlying Dirichlet process by setting a truncation level $\tilde{K}$ in the stick-breaking distribution, that is,

$$\begin{aligned}&u_k \sim \mathrm{Beta}(1,\alpha),\ k = 1, 2, \cdots, \tilde{K}-1 \text{ and } u_{\tilde{K}} = 1\\ &\pi_k = u_k \prod_{l=1}^{k-1}(1-u_l),\ k = 1, 2, \cdots, \tilde{K} \text{ and } \pi_k = 0, k > \tilde{K}\end{aligned}$$

such that $\tilde{K}$ is large enough to cover the finite and "true" cluster number in the attributed graph. On the other hand, since the truncated stick-breaking distribution is defined in the space over cluster labels instead of partitions, we can obtain the optimal labeling of the clusters by sorting the cluster proportion in decreasing order for maximizing the likelihood over different labelings (Kurihara et al., 2007; Zobay, 2009). We maintain this optimal labeling during inference.

We note that Miller et al. pointed out the Dirichlet process mixture is inconsistent in the component number but it can be mitigated by ignoring tiny clusters (Miller and Harrison, 2013). Thus, we follow the strategy in (Ghahramani and Beal, 1999) to progressively prune the tiny clusters during inference for both efficiency and consistency.

[6] The stick-breaking prior is a representation of the Dirichlet process and often used for variational inference. The Dirichlet process here is the distribution of a random probability measure over positive integers.

5.2. Variational Bayesian inference

We find the *maximum a posteriori* (MAP) estimate of the clustering $\mathbf{Z}$ conditioned on $(\mathbf{X}, \mathbf{Y})$ with fixed hyper-parameters (α, β, γ), i.e.,

$$\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \mathbf{Y}; \alpha, \beta, \gamma).$$

The exact inference of this estimation problem is intractable. Thus we use the mean field variational inference (Jordan et al., 1999; Xu, Ke and Wang, 2014) for tractability and efficiency. Specifically, it needs to define a family of variational distributions over latent variables $(\mathbf{Z}, \phi, \theta, u)$ in the factorized form

$$q(\mathbf{Z}, \phi, \theta, u|\tilde{\pi}, \tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}) = q(\mathbf{Z}|\tilde{\pi})q(\phi|\tilde{\gamma})q(\theta|\tilde{\beta})q(u|\tilde{\alpha}) \tag{4}$$

where

$$\begin{aligned}
q(\mathbf{Z}|\tilde{\pi}) &= \prod_{i=1}^{N} \text{Multinomial}(\mathbf{Z}_i|\tilde{\pi}_i), \\
q(\theta|\tilde{\beta}) &= \prod_{t=1}^{T}\prod_{k=1}^{\tilde{K}} \text{Multinomial}(\theta_k^{(t)}|\tilde{\beta}_k^{(t)}), \\
q(\phi|\tilde{\gamma}) &= \prod_{k=1}^{\tilde{K}} \text{Beta}(\phi_{kk}|\tilde{\gamma}_{kk}), \\
q(u|\tilde{\alpha}) &= \prod_{k=1}^{\tilde{K}-1} \text{Beta}(u_k|\tilde{\alpha}_k)
\end{aligned}$$

and $\tilde{\pi}$, $\tilde{\beta}$, $\tilde{\gamma}$ and $\tilde{\alpha}$ are variational parameters. Note that $\tilde{\pi}_i$, $\tilde{\beta}_k^{(t)}$, $\tilde{\gamma}_{kk}$ and $\tilde{\alpha}_k$ are vectors of size $\tilde{K}$, $M_t = |dom(a_t)|$, 2 and 2, respectively. We then find as its approximation the member of this family, $q^*(\mathbf{Z}, \phi, \theta, u)$ that is closest to the true posterior $p(\mathbf{Z}, \phi, \theta, u|\mathbf{X}, \mathbf{Y}; \alpha, \beta, \gamma)$ in terms of the KL divergence:

$$q^*(\mathbf{Z}, \phi, \theta, u) = \arg\min_{q \in \mathcal{Q}} \text{KL}(q(\mathbf{Z}, \phi, \theta, u)||p(\mathbf{Z}, \phi, \theta, u|\mathbf{X}, \mathbf{Y}))$$

where $\mathcal{Q} = \{q : q \text{ of the form } (4)\}$. Since variational distributions have fixed parametric forms, they are fully determined by the variational parameters. Thus, we can write

$$q^*(\mathbf{Z}, \phi, \theta, u) = q(\mathbf{Z}, \phi, \theta, u|\tilde{\pi}^*, \tilde{\alpha}^*, \tilde{\beta}^*, \tilde{\gamma}^*).$$

Then accordingly, our MAP estimate $\mathbf{Z}^*$ can be approximated as

$$\mathbf{Z}^* \approx \arg\max_{\mathbf{Z}} q(\mathbf{Z}|\tilde{\pi}^*) = (\arg\max_k \tilde{\pi}_{1k}^*, \cdots, \arg\max_k \tilde{\pi}_{Nk}^*)^T.$$

Minimizing KL divergence is equivalent to maximizing the evidence lower bound

$$L(q) = E[\log \frac{p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \phi, \theta, u)}{q(\mathbf{Z}, \phi, \theta, u)}]$$

on the marginal log likelihood $\log p(\mathbf{X}, \mathbf{Y})$ due to the identity

$$L(q) + KL(q(\mathbf{Z}, \phi, \theta, u)||p(\mathbf{Z}, \phi, \theta, u|\mathbf{X}, \mathbf{Y})) \equiv \log p(\mathbf{X}, \mathbf{Y}),$$

where the expectation is taken w.r.t the variational distribution. By optimizing

$L(q)$, i.e., $\max_{q\in\mathcal{Q}} L(q)$, we obtain the following update equations on the variational parameters which constitute the iterative steps of the coordinate ascent algorithm:

$$\tilde{\alpha}_k = (1 + \sum_i \tilde{\pi}_{ik}, \alpha + \sum_{l=k+1}^{\tilde{K}} \sum_i \tilde{\pi}_{il}) \tag{5}$$

$$\tilde{\beta}_k^{(t)} = \beta^{(t)} + \sum_i \tilde{\pi}_{ik}(\delta(\mathbf{Y}_{it}, a_{t1}), \cdots, \delta(\mathbf{Y}_{it}, a_{tM_t})) \tag{6}$$

$$\tilde{\gamma}_k = \gamma + \frac{1}{2}\sum_{i\neq j} \tilde{\pi}_{ik}\tilde{\pi}_{jk}(\mathbf{X}_{ij}, 1 - \mathbf{X}_{ij}) \tag{7}$$

$$\begin{aligned}\tilde{\pi}_{ik} \propto \exp\Big\{ & g(\tilde{\alpha}; k, 1) + \sum_{l=1}^{k-1} g(\tilde{\alpha}; l, 2) \\ & + (\log\varepsilon, \log(1-\varepsilon)) \sum_{l\neq k}\sum_{j\neq i} \tilde{\pi}_{jl}(\mathbf{X}_{ij}, 1 - \mathbf{X}_{ij})^T \\ & + (g(\tilde{\gamma}; k, 1), g(\tilde{\gamma}; k, 2)) \sum_{j\neq i} \tilde{\pi}_{jk}(\mathbf{X}_{ij}, 1 - \mathbf{X}_{ij})^T \\ & + \sum_t \sum_{m_t} \delta(\mathbf{Y}_{it}, a_{tm_t})[\psi(\tilde{\beta}_{km_t}^{(t)}) - \psi(\sum_{m_t'} \tilde{\beta}_{km_t'}^{(t)})]\Big\}\end{aligned} \tag{8}$$

where $g(a; k, r) = \psi(a_{kr}) - \psi(a_{k1} + a_{k2})$ ($r = 1$ or 2, and $a = \tilde{\alpha}$ or $\tilde{\gamma}$). Due to the update of $\tilde{\alpha}, \tilde{\beta}$, and $\tilde{\gamma}$, the evidence lower bound can be greatly simplified:

$$\begin{aligned}L(q) = & (\log\varepsilon, \log(1-\varepsilon)) \sum_{k\neq l}\sum_{i<j} \tilde{\pi}_{ik}\tilde{\pi}_{jl}(\mathbf{X}_{ij}, 1 - \mathbf{X}_{ij})^T - \sum_{i,k} \tilde{\pi}_{ik}\log\tilde{\pi}_{ik} \\ & + \sum_{k=1}^{\tilde{K}-1} \log\frac{\mathcal{B}(\tilde{\alpha}_{k1}, \tilde{\alpha}_{k2})}{\mathcal{B}(1, \alpha)} + \sum_{t,k} \log\frac{\mathcal{B}(\tilde{\beta}_k^{(t)})}{\mathcal{B}(\beta^{(t)})} + \sum_k \log\frac{\mathcal{B}(\tilde{\gamma})}{\mathcal{B}(\gamma)}\end{aligned} \tag{9}$$

where $\mathcal{B}(b) \triangleq \frac{\prod_i \Gamma(b_i)}{\Gamma(\sum_i b_i)}$ for any positive vector b.

We use the simplified evidence lower bound to monitor the convergence of our iterative procedure, which is summarized in Algorithm 1 and guaranteed to converge according to the theory of variational Bayes (Beal, 2003). For hyperparameters β and γ, we follow (Xu et al., 2012) to fix them to all-ones vectors throughout the paper, corresponding to non-informative priors[7]. For the hyperparameter α, according to the study in (Zobay, 2009; Yu et al., 2006), the result is insensitive to it and thus we set it to 1 as well. For pruning small clusters, we set a threshold ξ to define the small clusters by $\sum_i \tilde{\pi}_{ik}/N < \xi$. The threshold controls the granularity of clusters. Generally, the cluster size increases with this value. And the final cluster number $K^* < \tilde{K}$ due to the pruning can be automatically obtained and naturally viewed as the optimal one.

[7] That is, each prior is a uniform distribution over the components. This is reasonable given that we do not have any prior information on the proportion of different components and thus they are treated equally important.

Input: a truncation level $\tilde{K} = K^{(0)}$, an initial value $\tilde{\pi}^{(0)}$, tolerance η, a limit on the number of iterations $t_{\max}$
Output: a cluster number K^*, a vertex clustering $\mathbf{Z}^*$

1. $t \leftarrow 0$
2. **repeat**:
   (a) Given $\tilde{\pi}^{(t)}$, update $\tilde{\alpha}^{(t+1)}, \tilde{\beta}^{(t+1)}, \tilde{\gamma}^{(t+1)}$ according to Equations (5)–(7)
   (b) Check stopping criterion: $L(q^{(t)}) - L(q^{(t-1)}) < \eta$ or $t > t_{\max}$
   (c) Given $\tilde{\alpha}^{(t+1)}, \tilde{\beta}^{(t+1)}, \tilde{\gamma}^{(t+1)}, \tilde{\pi}^{(t)}$, update $\tilde{\pi}^{(t+1)}$ according to Equation (8)
   (d) Reorder clusters and prune small ones
   (e) $t \leftarrow t + 1$
3. **return** $K^* = K^{(t)}$, $\mathbf{Z}^* = (\arg\max_k \tilde{\pi}^{(t)}_{1k}, \cdots, \arg\max_k \tilde{\pi}^{(t)}_{Nk})$

Algorithm 1: NBAGC - Iterative Optimization of $L(q)$

5.3. Complexity Analysis

We analyze the complexity of our NBAGC algorithm as follows. For the large datasets with $N \gg K$ and $N \gg M_t$ (for all t), the time complexity of NBAGC is dominated by the quadratic time consumption for updating $\tilde{\gamma}$ and $\tilde{\pi}$. However, real graphs often exhibit sparse structures, and note that

$$\sum_{i \neq j} \tilde{\pi}_{ik}\tilde{\pi}_{jk} = (\sum_i \tilde{\pi}_{ik})(\sum_j \tilde{\pi}_{jk}) - \sum_i \tilde{\pi}_{ik}\tilde{\pi}_{ik},$$
$$\sum_{l \neq k} \sum_{j \neq i} \tilde{\pi}_{jl}\mathbf{X}_{ij} = \sum_{j \neq i} \mathbf{X}_{ij}(1 - \tilde{\pi}_{jk}).$$

Thus, each update of $\tilde{\gamma}$ and $\tilde{\pi}$ actually can be calculated in $O(\mathrm{nnz}(\mathbf{X})\tilde{K})$ time, where $\mathrm{nnz}(\mathbf{X})$ represents the undirected edge number. For the space complexity, the consumption, compared to those for parameters $(\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\pi})$, mainly comes from the storage of input data $(\mathbf{X}, \mathbf{Y})$, i.e., $O(\mathrm{nnz}(\mathbf{X}) + NT)$ under the sparse representation of adjacency matrix $\mathbf{X}$.

Thus, our NBAGC algorithm has both time and space complexity linear in the number of edges, vertices and attributes.

6. Asymptotic Approach

The asymptotic method we consider for model selection in attributed graph clustering is based on a recently proposed model selection criterion called factorized information criterion (FIC) (Fujimaki and Morinaga, 2012). FIC, which is a variant of Bayesian information criterion (BIC) (Bishop, 2006) without the regularity assumption of BIC, can be applied to non-regular models, including mixture models, neural networks, and hidden Markov models. The key difference from BIC is that FIC applies Laplace approximation to each of components in the factorized representation of joint likelihood, instead of the whole of joint likelihood directly as in BIC.

Next we will derive the FIC of BAGC model in Section 6.1, and then present an algorithm for simultaneous parameter estimation and model selection through maximizing the FIC in Section 6.2.

6.1. BAGC FIC

Similar to BIC, FIC is an asymptotic approximation to the marginal data evidence

$$\begin{aligned}\log p(\mathbf{X},\mathbf{Y}|\alpha,\beta,\gamma) &= \log\sum_{\mathbf{Z}} p(\mathbf{X},\mathbf{Y},\mathbf{Z}|\alpha,\beta,\gamma)\\ &= \log\sum_{\mathbf{Z}}\iiint p(\mathbf{X},\mathbf{Y},\mathbf{Z},\phi,\theta,\pi|\alpha,\beta,\gamma)\mathrm{d}\phi\mathrm{d}\theta\mathrm{d}\pi.\end{aligned}$$

We start from the derivation with the joint likelihood of BAGC model:

$$\begin{aligned}\log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\phi,\theta,\pi) = &\sum_{k\neq l}\sum_{i<j}\mathbf{Z}_{ik}\mathbf{Z}_{jl}\log p(\mathbf{X}_{ij}|\varepsilon) + \sum_k \log p(\mathbf{X},\phi_{kk}|\mathbf{Z}_k;\gamma)\\ &+\sum_{t,k}\log p(\mathbf{Y}_t,\theta_k^{(t)}|\mathbf{Z}_k;\beta^{(t)}) + \log p(\mathbf{Z},\pi|\alpha)\end{aligned}$$

where

$$\begin{aligned}\log p(\mathbf{X},\phi_{kk}|\mathbf{Z}_k;\gamma) &= \log p(\phi_{kk}|\gamma) + \sum_{i<j}\mathbf{Z}_{ik}\mathbf{Z}_{jk}\log p(\mathbf{X}_{ij}|\phi_{kk}),\\ \log p(\mathbf{Y}_t,\theta_k^{(t)}|\mathbf{Z}_k;\beta^{(t)}) &= \log p(\theta_k^{(t)}|\beta^{(t)}) + \sum_i \mathbf{Z}_{ik}\log p(\mathbf{Y}_{it}|\theta_k^{(t)}),\\ \log p(\mathbf{Z},\pi;\alpha) &= \log p(\pi|\alpha) + \sum_i \log p(\mathbf{Z}_i|\pi).\end{aligned}$$

Taylor approximating each of the components $\log p(\mathbf{X},\phi_{kk}|\mathbf{Z}_k;\gamma)$, $\log p(\mathbf{Y}_t,\theta_k^{(t)}|\mathbf{Z}_k;\beta^{(t)})$ and $\log p(\mathbf{Z},\pi;\alpha)$ around the MAP estimate of parameters (ϕ,θ,π), denoted as $(\bar{\phi},\bar{\theta},\bar{\pi})$, to the second order gives us

$$\begin{aligned}&\log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\phi,\theta,\pi)\\ &\approx \log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\bar{\phi},\bar{\theta},\bar{\pi}) - \sum_k \frac{\gamma_0+\sum_{i<j}\mathbf{Z}_{ik}\mathbf{Z}_{jk}}{2}\tilde{I}(\bar{\phi}_{kk})(\phi_{kk}-\bar{\phi}_{kk})^2\\ &\quad -\sum_{t,k}\frac{\beta_0^{(t)}+\sum_i \mathbf{Z}_{ik}}{2}(\vartheta_k^{(t)}-\bar{\vartheta}_k^{(t)})^T\tilde{I}(\bar{\vartheta}_k^{(t)})(\vartheta_k^{(t)}-\bar{\vartheta}_k^{(t)})\\ &\quad -\frac{\alpha_0+N}{2}(\varpi-\bar{\varpi})^T\tilde{I}(\bar{\varpi})(\varpi-\bar{\varpi})\end{aligned}$$

where $\vartheta_k^{(t)} = (\theta_{k1}^{(t)},\cdots,\theta_{k,M_t-1}^{(t)})^T$, $\varpi = (\pi_1,\cdots,\pi_{\tilde{K}-1})^T$ (similarly for $\bar{\vartheta}_k$ and $\bar{\varpi}$), $\alpha_0 = \sum_k(\alpha_k-1)$, $\beta_0^{(t)} = \sum_{m_t}(\beta_{m_t}^{(t)}-1)$, and $\gamma_0 = \sum_{i=1}^2(\gamma_i-1)$. Especially,

$$\begin{aligned}\tilde{I}(\phi_{kk}) &= -\frac{1}{\gamma_0+\sum_{i<j}\mathbf{Z}_{ik}\mathbf{Z}_{jk}}\frac{\partial^2\log p(\mathbf{X},\phi_{kk}|\mathbf{Z}_k;\gamma)}{\partial\phi_{kk}^2},\\ \tilde{I}(\vartheta_k^{(t)}) &= -\frac{1}{\beta_0^{(t)}+\sum_i\mathbf{Z}_{ik}}\frac{\partial^2\log p(\mathbf{Y}_t,\theta_k^{(t)}|\mathbf{Z}_k;\beta^{(t)})}{\partial\vartheta_k^{(t)}\partial(\vartheta_k^{(t)})^T},\\ \tilde{I}(\varpi) &= -\frac{1}{\alpha_0+N}\frac{\partial^2\log p(\mathbf{Z},\pi;\alpha)}{\partial\varpi\partial\varpi^T}\end{aligned}$$

are observed Fisher information and asymptotically consistent with their expected counterparts below

$$I(\phi_{kk}) = -E_{\mathbf{X}}[\frac{\partial^2 \log p(\mathbf{X}_{ij}|\phi_{kk})}{\partial \phi_{kk}^2}],$$
$$I(\vartheta_k^{(t)}) = -E_{\mathbf{Y}}[\frac{\partial^2 \log p(\mathbf{Y}_{it}|\theta_k^{(t)})}{\partial \vartheta_k^{(t)} \partial (\vartheta_k^{(t)})^T}],$$
$$I(\varpi) = -E_{\mathbf{Z}}[\frac{\partial^2 \log p(\mathbf{Z}_i|\pi)}{\partial \varpi \partial \varpi^T}].$$

Then by applying Laplace approximation (Bishop, 2006) and ignoring asymptotically small terms, we arrive at

$$\begin{aligned}
&\log p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}|\alpha, \beta, \gamma) \\
&\approx \log p(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \bar{\phi}, \bar{\theta}, \bar{\pi}) - \frac{2 + \sum_t (M_t - 1)}{2} \sum_k \log(\sum_i \mathbf{Z}_{ik}) - \frac{\tilde{K} - 1}{2} \log N \\
&\triangleq \log \tilde{p}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}|\alpha, \beta, \gamma)
\end{aligned}$$

where we have used the following facts: as $N \to \infty$,

$$\frac{2\gamma_0 + \sum_{i \neq j} \mathbf{Z}_{ik} \mathbf{Z}_{jk}}{(\sum_i \mathbf{Z}_{ik})^2} \to 1, \frac{\beta_0^{(t)} + \sum_i \mathbf{Z}_{ik}}{\sum_i \mathbf{Z}_{ik}} \to 1, \text{ and } \frac{\alpha_0 + N}{N} \to 1.$$

Now we can define the FIC for BAGC model as follows

$$\mathrm{FIC}_{\mathrm{BAGC}}(\mathbf{X}, \mathbf{Y}; \alpha, \beta, \gamma) = \max_{q(\mathbf{Z})} E[\log \frac{\tilde{p}(\mathbf{X}, \mathbf{Y}, \mathbf{Z}|\alpha, \beta, \gamma)}{q(\mathbf{Z})}]$$

which is asymptotically consistent with the marginal data evidence, that is,

$$\log p(\mathbf{X}, \mathbf{Y}|\alpha, \beta, \gamma) = \lim_{N \to \infty} \mathrm{FIC}_{\mathrm{BAGC}}(\mathbf{X}, \mathbf{Y}; \alpha, \beta, \gamma).$$

The proof of the above claim, as well as several nice properties of $\mathrm{FIC}_{\mathrm{BAGC}}$ due to the regularization term $\frac{2+\sum_t (M_t - 1)}{2} E[\log(\sum_i \mathbf{Z}_{ik})]$, are similar to that in (Fujimaki and Morinaga, 2012).

6.2. Factorized Asymptotic Bayesian Inference

The goal now is to maximize $\mathrm{FIC}_{\mathrm{BAGC}}$. However, the MAP estimate $(\bar{\phi}, \bar{\theta}, \bar{\pi})$ are not available. Thus similarly, we try to maximize a variational lower bound on $\mathrm{FIC}_{\mathrm{BAGC}}$, which is the so-called factorized asymptotic Bayesian inference (FAB) (Fujimaki and Morinaga, 2012).

6.2.1. FIC Lower Bound

For tractability, we employ the fully factorized mean field variational distribution $q(\mathbf{Z}|\tilde{\pi}) = \prod_i q(\mathbf{Z}_i|\tilde{\pi}_i)$ where $q(\mathbf{Z}_i|\tilde{\pi}_i)$ follows the multinomial distribution, and

meanwhile apply the following two inequalities in $\mathrm{FIC}_{\mathrm{BAGC}}$.

$$\log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\bar{\phi},\bar{\theta},\bar{\pi}|\alpha,\beta,\gamma) \geq \log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\phi,\theta,\pi|\alpha,\beta,\gamma)$$
$$-\log a \geq -\log b - \frac{a}{b} + 1.$$

We get

$$\begin{aligned}
&\mathrm{FIC}_{\mathrm{BAGC}}(\mathbf{X},\mathbf{Y};\alpha,\beta,\gamma)\\
&\geq E[\log p(\mathbf{X},\mathbf{Y},\mathbf{Z},\phi,\theta,\pi|\alpha,\beta,\gamma)] - \frac{\tilde{K}-1}{2}\log N - E[\log q(\mathbf{Z})]\\
&\quad - \frac{2+\sum_t (M_t-1)}{2}\sum_k (\log h_k + \frac{\sum_i E[\mathbf{Z}_{ik}]}{h_k} - 1)\\
&\triangleq L(\phi,\theta,\pi,\tilde{\pi},h;\alpha,\beta,\gamma) \qquad (10)
\end{aligned}$$

where h is an additional variational parameter.

6.2.2. FAB Optimization

We present the FAB algorithm for simultaneous parameter estimation and model selection for BAGC, called FABAGC. It is similar to the variational EM algorithm and consisting of variational E-step and M-step.

E-step E-step maximizes the FIC lower bound L w.r.t variational parameters h and $\tilde{\pi}$, and has the update equations $h_k = \sum_i \tilde{\pi}_{ik}$ and

$$\tilde{\pi}_{ik} \propto \pi_k p(\mathbf{Y}_i|\theta_k)\exp\{\sum_{j\neq i}\sum_{l\neq k}\tilde{\pi}_{jl}\log p(\mathbf{X}_{ij}|\phi_{kl}) \underbrace{-\frac{2+\sum_t (M_t-1)}{2h_k}}_{\text{shrinking term}}\}. \qquad (11)$$

Compared to the conventional variational EM algorithm (Daudin et al., 2008), the difference of $\tilde{\pi}_{ic}$ update (11) lies only at the attribute part $p(\mathbf{Y}_i|\theta_k)$ and the extra factor $\exp\{-[2+\sum_t (M_t-1)]/[2h_k]\}$ where $h_k \approx N\pi_k$. This factor plays the role of regularization such that small clusters without sufficient data support will be shrunk gradually, achieving the automatic model selection. In practice, we can prune the clusters that have been shrunk to the size less than a threshold ξ, for accelerating the optimization process. Theoretically, ξ could be arbitrarily small. For real application, it can be set based on our prior knowledge on data as well as our requirements for the clustering results. As with NBAGC, we also need to initially choose a cluster number $\tilde{K}$ that is large enough to cover the "correct" one.

Input: an initial cluster number $\tilde{K} = K^{(0)}$, an initial value $\tilde{\pi}^{(0)}$, thresholds ε, η, ξ, a limit on the number of iterations $t_{\max}$
Output: a cluster number K^*, a vertex clustering $\mathbf{Z}^*$

1. $t \leftarrow 0$
2. **repeat**:
   (a) Given $\tilde{\pi}^{(t)}$, update $\phi^{(t+1)}, \theta^{(t+1)}, \pi^{(t+1)}$ according to Equations (12)–(14)
   (b) Check stopping criterion: $L^{(t)} - L^{(t-1)} < \eta$ or $t > t_{\max}$
   (c) Given $\phi^{(t+1)}, \theta^{(t+1)}, \pi^{(t+1)}, \tilde{\pi}^{(t)}$, update $\tilde{\pi}^{(t+1)}$ according to Equation (11)
   (d) Prune tiny clusters
   (e) $t \leftarrow t + 1$
3. **return** $K^* = K^{(t)}$, $\mathbf{Z}^* = (\arg\max_k \tilde{\pi}^{(t)}_{1k}, \cdots, \arg\max_k \tilde{\pi}^{(t)}_{Nk})$

Algorithm 2: FABAGC - Iterative Optimization of L

M-step M-step maximizes the FIC lower bound w.r.t. the model parameters (ϕ, θ, π), and has the following update equations

$$\phi_{kk} = \frac{\gamma_1 - 1 + \sum_{i<j} \tilde{\pi}_{ik}\tilde{\pi}_{jk}\mathbf{X}_{ij}}{\gamma_0 + \sum_{i<j} \tilde{\pi}_{ik}\tilde{\pi}_{jk}}, \tag{12}$$

$$\theta^{(t)}_{km_t} = \frac{\beta^{(t)}_{m_t} - 1 + \sum_i \tilde{\pi}_{ik}\delta(\mathbf{Y}_{it}, a_{tm_t})}{\beta^{(t)}_0 + \sum_i \tilde{\pi}_{ik}}, \tag{13}$$

$$\pi_k = \frac{\alpha_k - 1 + \sum_i \tilde{\pi}_{ik}}{\alpha_0 + N}. \tag{14}$$

It can be seen that the M-step is exactly the same as that with the variational EM algorithm for the MAP inference of BAGC. As $N \to \infty$, the prior information over (ϕ, θ, π) can be ignored. However, they can be used for parameter smoothing. For this reason, it suffices to set them (α, β, γ) to the constant vectors with entry value 2 in our experiments.

The iterative procedure with FABAGC is given in Algorithm 2. We can see that although FABAGC is derived from the Bayesian perspective, it ends up with a frequentist solution. And the shrinkage mechanism in FABAGC is explicit and intuitive, simply due to the regularization term. By inspecting the update equations for both approaches, we can see that FABAGC has the same time and space complexities as NBAGC.

7. Experimental Evaluation

We evaluate NBAGC and FABAGC on synthetic datasets and three real datasets, comparing with PICS (Akoglu et al., 2012). All the three algorithms were implemented in Matlab and tested on machines with Linux OS, Intel Xeon 2.67GHz CPUs, and 12GB of RAM. Following the studies in (Xu et al., 2012), our experiments are conducted on attributed graphs with undirected and un-weighted structures.

7.1. Experimental Setting

In this subsection, we detail the experimental settings including the baseline algorithm PICS, clustering quality measures, as well as the initialization of parameter $\tilde{\pi}$.

7.1.1. The PICS Algorithm

PICS (Akoglu et al., 2012) represents the state-of-the-art model selection algorithm for attributed graph clustering. It aims at co-clustering of vertices and binary attributes. Since a cluster of vertices that exhibit similar connectivity patterns and homogeneous attribute values incur a low encoding cost, PICS applies the minimum description length (MDL) principle to find the optimal number of clusters and co-clustering. Due to the NP-hardness of finding the optimum, PICS resorts to a greedy iterative heuristic solution.

For PICS to be applicable to categorical attributes, we decompose each of them, say a_t, into M_t binary attributes by using the 1-of-M_t vector. For example, if $\mathbf{Y}_{it} = a_{t1}$, then the corresponding binary attributes for $\mathbf{Y}_{it}$ is the vector $(1, 0, \cdots, 0)$ of length M_t.

7.1.2. Clustering Quality Measures

For datasets with ground truth, e.g., synthetic datasets, we use Normalized Mutual Information (NMI) (Strehl and Ghosh, 2002), which is commonly used to determine the clustering quality. Suppose that the given true clustering is $\tilde{\mathcal{C}} \triangleq \{\tilde{V}_1, \cdots, \tilde{V}_{K^\star}\}$ and the clustering obtained by an algorithm is $\mathcal{C} \triangleq \{V_1, \cdots, V_{K^*}\}$. Then NMI is estimated by

$$\mathrm{NMI}(\mathcal{C}, \tilde{\mathcal{C}}) = \frac{\sum_{k=1}^{K^*} \sum_{l=1}^{K^\star} N_{kl} \log \frac{N_{kl}}{N_k \tilde{N}_l}}{\sqrt{(\sum_{k=1}^{K^*} N_k \log \frac{N_k}{N})(\sum_{l=1}^{K^\star} \tilde{N}_l \log \frac{\tilde{N}_l}{N})}}$$

where N_k represents the number of vertices contained in the cluster V_k, $\tilde{N}_l$ represents the number of vertices contained in the cluster $\tilde{V}_l$, and N_{kl} represents the number of vertices contained in the intersection of two clusters V_k and $\tilde{V}_l$. The range of NMI value is $[0, 1]$ and a larger NMI value $\mathrm{NMI}(\mathcal{C}, \hat{\mathcal{C}})$ indicates a better clustering $\mathcal{C}$. When $\mathrm{NMI}(\mathcal{C}, \tilde{\mathcal{C}}) = 1$, the two clusterings are identical, i.e., $\mathcal{C} = \tilde{\mathcal{C}}$.

For real datasets without ground truth, since our objective is to cluster attributed graph, we assess the quality of the clustering in two aspects, *structure* and *attribute*.

We use *modularity* as a quality measure for structure. Modularity (Newman and Girvan, 2004) is popularly used in graph clustering to measure the strength of division of a graph into vertex clusters. Given a clustering $\mathcal{C} \triangleq \{V_1, \cdots, V_K\}$, let E_{kl} $(k \neq l)$ be the set of inter-cluster edges between V_k and V_l, and E_{kk} the set of intra-cluster edges in V_k. Then the fraction of inter-cluster edges is defined by $f_{kk} = \frac{|E_{kk}|}{|E|}$, and the fraction of inter-cluster edges between V_k and V_l $(k \neq l)$ is defined by $f_{kl} = f_{lk} = \frac{|E_{kl}|}{2|E|}$. By counting both intra-cluster and inter-cluster edges, the fraction of edge incident to cluster V_k is defined by $g_k = \sum_{l=1}^{K} f_{kl}$.

Then the modularity is defined by

$$\mathit{modularity}(\mathcal{C}) = \sum_{k=1}^{K} (f_{kk} - g_k^2).$$

The value of modularity falls within the range of $[-1, 1]$. A clustering result with high modularity has dense vertex connections within the same cluster and sparse vertex connections across different clusters.

For attributes, we use *entropy* as a quality measure to measure the degree of heterogeneity of attribute values in each cluster. Given a clustering $\mathcal{C} \triangleq \{V_1, \cdots, V_K\}$, for each attribute a_t, the entropy of a_t in cluster V_k is defined by

$$\mathit{entropy}(a_t, V_k) = -\sum_{m_t=1}^{M_t} p_{ktm_t} \log p_{ktm_t}$$

where p_{ktm_t} is the fraction of vertices in clusters V_k that take the value a_{tm_t}. Then, the entropy of an attribute a_t w.r.t. the clustering $\mathcal{C}$ is defined by

$$\mathit{entropy}(a_t) = \sum_{k=1}^{K} \frac{N_k}{N} \mathit{entropy}(a_t, V_k).$$

The range of entropy is $[0, \infty)$. A lower entropy indicates a higher degree of homogeneity in the attribute values associated with the vertices in the same cluster and thus a higher intra-cluster attribute similarity.

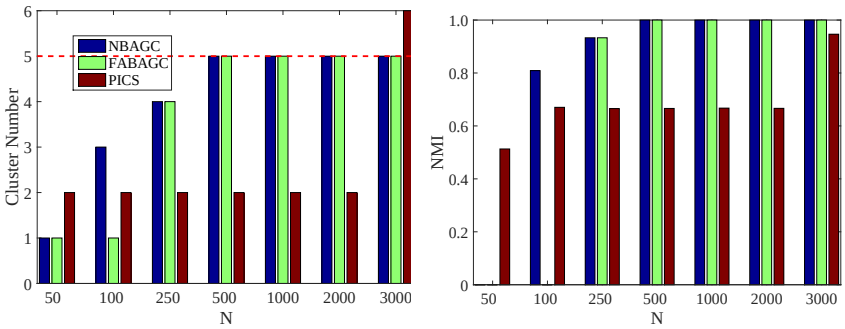
7.1.3. Initialization of Variational Parameter

We use METIS (Karypis and Kumar, 1998) to initialize our NBAGC and FABAGC algorithms. METIS is a fast structure-based graph partitioning algorithm. It partitions the graph vertices into K equally-sized clusters with minimum number or weighted sum of inter-cluster edges. To initialize the two algorithms, we set $\tilde{\pi}_{ik} = 1$ if the i-th vertex is assigned to the k-th cluster by METIS and $\tilde{\pi}_{il} = 0$ for $l \neq k$. When the initial cluster number $K^{(0)}$ is set to be large, our approaches will start with many small clusters imported from METIS, and most of them will rapidly vanish under a large size threshold for pruning tiny clusters. Then, we only prune the tiniest cluster each time once it satisfies the pruning condition, which saves more time for sufficient inference of optimal clustering by our approaches. For PICS, it is parameter-free and we use the implementation provided by the authors.

The threshold η for the lower bound L is set to 10^{-8}, the probability of inter-cluster connection ε is set to 10^{-6}, and the limit on the number of iterations $n_{\max}$ is set to 200 for the largest dataset and 100 for all the other datasets.

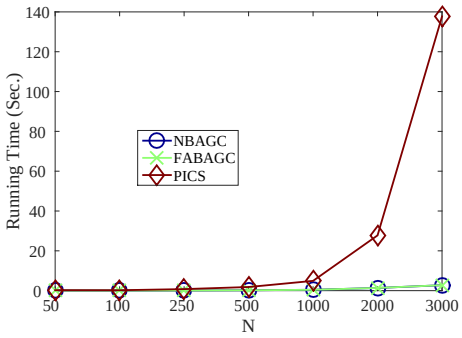
7.2. Performance on Synthetic Datasets

In order to test the capability and limitation of our algorithms, we generated 7 datasets with one attribute ($T = 1$) by Equations (1)-(3) in Section 4.2 (which constitutes the non-Bayesian version of the BAGC model). Following the setting in (Fujimaki and Morinaga, 2012), we set the sizes of these synthetic datasets



(a) Cluster number

(b) NMI



(c) Time

Fig. 2. Performance on Synthetic Datasets

to $N = \{50, 100, 250, 500, 1000, 2000\ , 3000\}$, the true cluster number $K^{\bigstar} = 5$, the initial cluster number $K^{(0)} = 20$, and the size threshold for pruning tiny clusters $\xi = 0.01$. And we use random parameter values sampled as follows before normalization, $\pi = (0.1, 0.15, 0.2, 0.25, 0.3) + \tau_1$, $\phi_{kk} = 0.8 + \tau_2$ for all k, and $\phi_{kl} = 0.2 + \tau_3$ for $l \neq k$, as well as

$$\theta = \begin{pmatrix} 0.25 & 0.15 & 0.15 & 0.15 & 0.15 \\ 0.15 & 0.25 & 0.15 & 0.15 & 0.15 \\ 0.15 & 0.15 & 0.25 & 0.15 & 0.15 \\ 0.15 & 0.15 & 0.15 & 0.25 & 0.15 \\ 0.15 & 0.15 & 0.15 & 0.15 & 0.25 \end{pmatrix} + \tau_4$$

where each τ entry follows the normal distribution $\mathcal{N}(0, 0.01)$.

Figure 2 shows the performance results of the three algorithms on the 7 synthetic datasets. Specifically, NBAGC and FABAGC find the correct cluster number $K^* = 5$ (shown as the red dotted horizontal line) as the dataset size becomes larger than or equal to 500, while PICS fails to find the correct cluster number for all the 7 datasets.

For the smaller datasets of size less than 500, both NBAGC and FABAGC underestimate the cluster number. We first observe that for clustering the smaller datasets, since there is no sufficient data to support the survival of tiny clusters, they vanish due to the shrinking power of NBAGC and FABAGC. As some of these tiny clusters may be true clusters, consequently our algorithms cannot

recover the ground truth for clustering small datasets. Further, we notice that FABAGC obtains fewer clusters than NBAGC at $N = 100$, which indicates that FABAGC is stronger than NBAGC in its shrinking strength. Since FABAGC is an asymptotic method and needs more data than non-asymptotic ones, like NBAGC, to work with, NBAGC computes a result closer to the truth than FABAGC when the dataset size is small. When the amount of data increases to be sufficient, both NBAGC and FABAGC are able to recover the true cluster number by shrinking superfluous clusters. However, whatever the dataset size is, PICS either underestimates or overestimates the cluster number, which is likely due to the fact that PICS is a heuristic-based algorithm.

It is worth noting that even if we delete the shrinking procedure in Step (b) of Algorithms 1 and 2, NBAGC and FABAGC can still find the true cluster number when $N \geq 500$. This implies that shrinking power is an inherent property of NBAGC and FABAGC, instead of coming from the shrinking step we introduce. We introduce the shrinking step only for the acceleration of clustering computation.

Now we report the result of the quality of the clustering obtained by the three algorithms. When NBAGC and FABAGC find the ground truth of cluster number for $N \geq 500$, they attain the maximum NMI value of 1, which indicates that NBAGC and FABAGC not only obtain the true cluster number, but also simultaneously recover the true clustering (i.e., the true cluster membership of each vertex). PICS does not find the true clustering for any dataset, but PICS performs better than NBAGC at $N = 50$ and FABAGC at $N = 50, 100$ since NBAGC and FABAGC return a single cluster clustering for these datasets, and we note that NMI is not suitable for measuring the quality of clustering with only a single cluster. Except for the single cluster case, both of our approaches find clusterings of higher quality than PICS.

In terms of clustering efficiency, both NBAGC and FABAGC use much less time than PICS. Especially for clustering the larger datasets, the running time of PICS grows rapidly, while the running time of both NBAGC and FABAGC increases slowly, showing that our algorithms are scalable. Although PICS have a linear complexity similar to our case (given in Section 5.3), it employs a greedy and heuristic algorithm to indirectly optimize the objective function (i.e., total encoding cost) in the combinatorial space and thus needs more iterations to reach the optimum. On the contrary, our approaches optimize the objective function (i.e., likelihood) directly using variational inference and thus are more scalable in practice.

In summary, given sufficient data, our approaches are consistently better than PICS in terms of cluster numbers, clustering quality, and clustering efficiency and scalability.

7.3. Performance on Real Datasets

We now evaluate our algorithms on three real datasets, which are described as follows.

– **Political Blogs**. The dataset has $1,490$ vertices and $19,090$ edges. Each vertex represents a webblog on US politics and each directed edge represents a hyperlink from one webblog to another. Each vertex is associated with an attribute, indicating the political leaning of the webblog, *liberal* or *conservative*.

Since we only consider undirected graphs in this work, we ignore the edge directions in this dataset, which results in 16,715 undirected edges.

- **DBLP10K**. The dataset is a co-author network extracted from the DBLP Bibliography data. Each vertex represents a scholar and each edge represents a co-author relationship between two scholars. The dataset contains 10,000 scholars who have published in major conferences in four research fields: database, data mining, information retrieval, and artificial intelligence. Each scholar is associated with two attributes, *prolific* and *primary topic*. The attribute "*prolific*" has three values: "highly prolific" for the scholars with ≥ 20 publications, "prolific" for the scholars with ≥ 10 and < 20 publications, and "low prolific" for the scholars with < 10 publications. The domain of the attribute "*primary topic*" consists of 100 research topics extracted by a topic model (Hofmann, 1999) from a collection of paper titles from the scholars. Each scholar is then assigned a primary topic out of the 100 topics.
- **DBLP84K**. This dataset is a larger DBLP co-author network. It contains 84,170 scholars in 15 research fields. In addition to the four research fields used in *DBLP10K*, 11 additional fields are further included: machine learning, computer vision, networking, multimedia, computer systems, simulation, theory, architecture, natural language processing, human-computer interaction, and programming language. This dataset also has two vertex attributes, which are defined in a similar way as in *DBLP10K*.

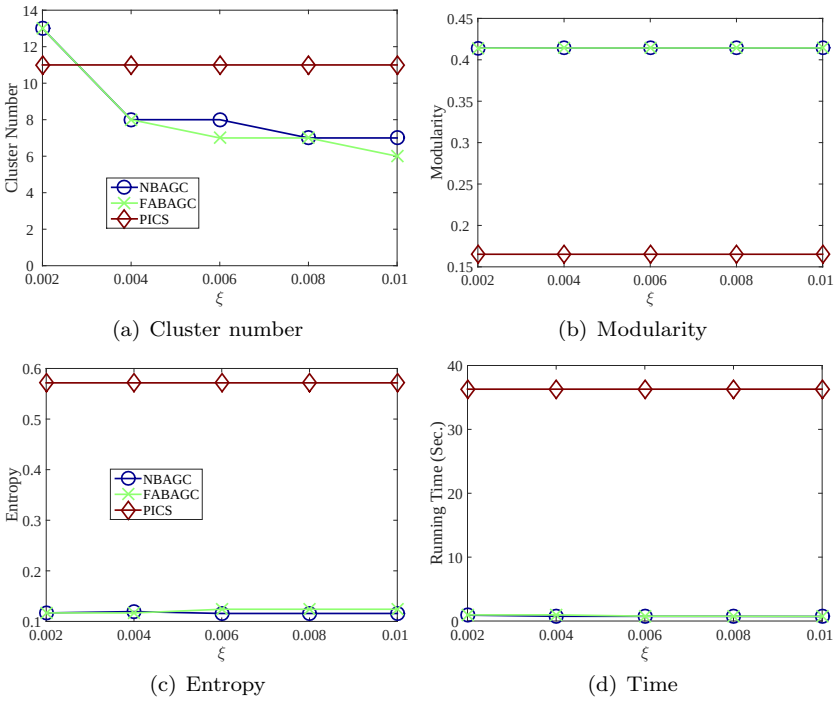
7.3.1. Performance on Political Blogs

We first report the results on clustering Political Blogs. To investigate the influence of two parameters, initial cluster number $K^{(0)}$ and the size threshold ξ for pruning tiny clusters, on the clustering results, we fix one and vary the other.

We first fix the initial cluster number to $K^{(0)} = 40$, and let the size threshold ξ vary in $\{0.002, 0.004, 0.006, 0.008, 0.01\}$. Figure 3 reports the clustering results of the three algorithms on the Political Blogs dataset. As the size threshold ξ increases, the cluster number obtained by our approaches decreases since more tiny clusters are pruned. In most cases, NBAGC and FABAGC obtain fewer clusters than PICS, while FABAGC obtains fewer clusters than NBAGC. For the clustering quality, Figures 3(b) and 3(c) show that NBAGC and FABAGC achieve a much higher modularity value and a much smaller entropy value than PICS, meaning that the clusterings computed by NBAGC and FABAGC are of much higher quality than that by PICS, in terms of both community structure and homogeneity of attribute values. The high modularity value and small entropy value of NBAGC and FABAGC also indicate that the corresponding clusterings contain a significant community structure with highly homogenous attribute values. The big difference in the clustering quality can be explained as follows. The Political Blogs dataset belongs to disassortative networks[8] (Zhou, Lü and Zhang, 2009) and PICS also aims at the disassortative mixing (Newman, 2002) that contradicts the fundamental assumptions of a good clustering. On the contrary, our approaches are based on a model that conforms to the fundamental assumptions of a good clustering.

In terms of the clustering efficiency, Figure 3(d) clearly shows that our approaches are significantly more efficient than PICS in all cases.

[8] The corresponding assortativity coefficient is negative, $r = -0.079$.



(a) Cluster number (b) Modularity

(c) Entropy (d) Time

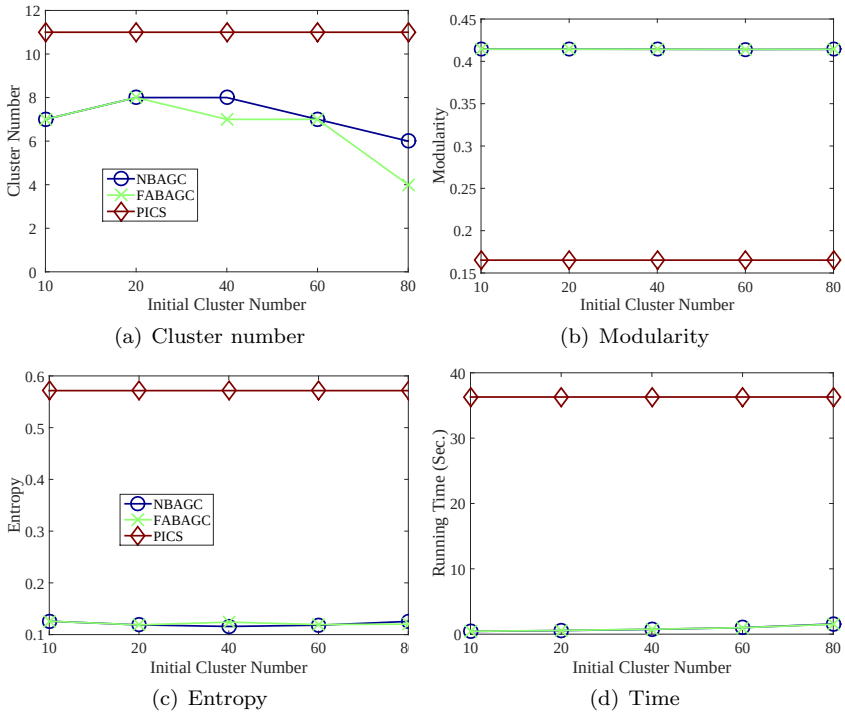
Fig. 3. Influence of ξ on clustering Political Blogs ($K^{(0)} = 40$)

Next, we fix $\xi = 0.006$ and vary $K^{(0)}$ in $\{10, 20, 40, 60, 80\}$. Figure 4 reports the corresponding results, which are similar to those in the previous setting. The final cluster number also decreases as the initial cluster number $K^{(0)}$ is set larger, which is caused by our initialization method METIS. METIS tries to partition the vertex set into equally-sized clusters. Thus, as the initial cluster number becomes large, NBAGC and FABAGC are initially fed with many small clusters that will be shrunk more easily.

In addition, we also observe that both NBAGC and FABAGC achieve rather stable clustering quality, in terms of both modularity and entropy, as well as stable running time, as ξ or $K^{(0)}$ varies. Thus, ξ and $K^{(0)}$ mainly impact the cluster number, that is, the compactness of the cluster structure, and their choices are relatively insensitive to the clustering quality.

Combining the clustering quality results, in terms of modularity and entropy, cluster number and clustering efficiency together, we can conclude that NBAGC and FABAGC find much more compact cluster structure, and are meanwhile orders of magnitude faster than PICS.

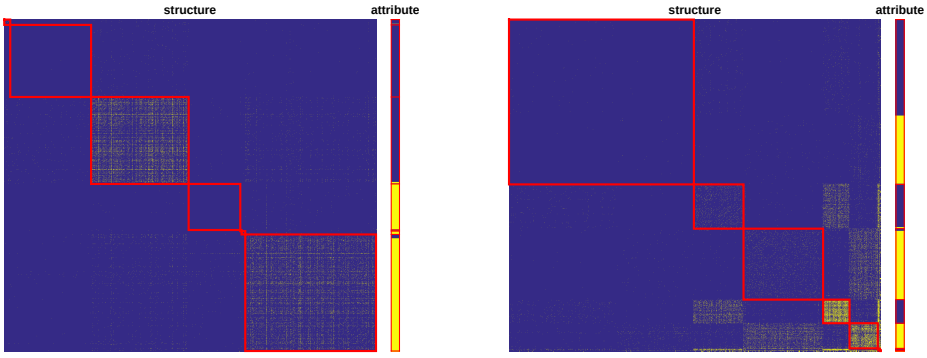
We also plot the cluster structures that are detected by our algorithm FABAGC and PICS in Figure 5, where we omit the cluster structure detected by NBAGC since it is very similar to that found by FABAGC. As we can see, these cluster structures detected by two algorithms are consistent with their performance measures, i.e., modularity and entropy. For example, FABAGC achieves a much higher modularity value than PICS. We thus see more edges crossing clusters from PICS. Meanwhile, FABAGC attains a lower entropy value and hence more



(a) Cluster number (b) Modularity

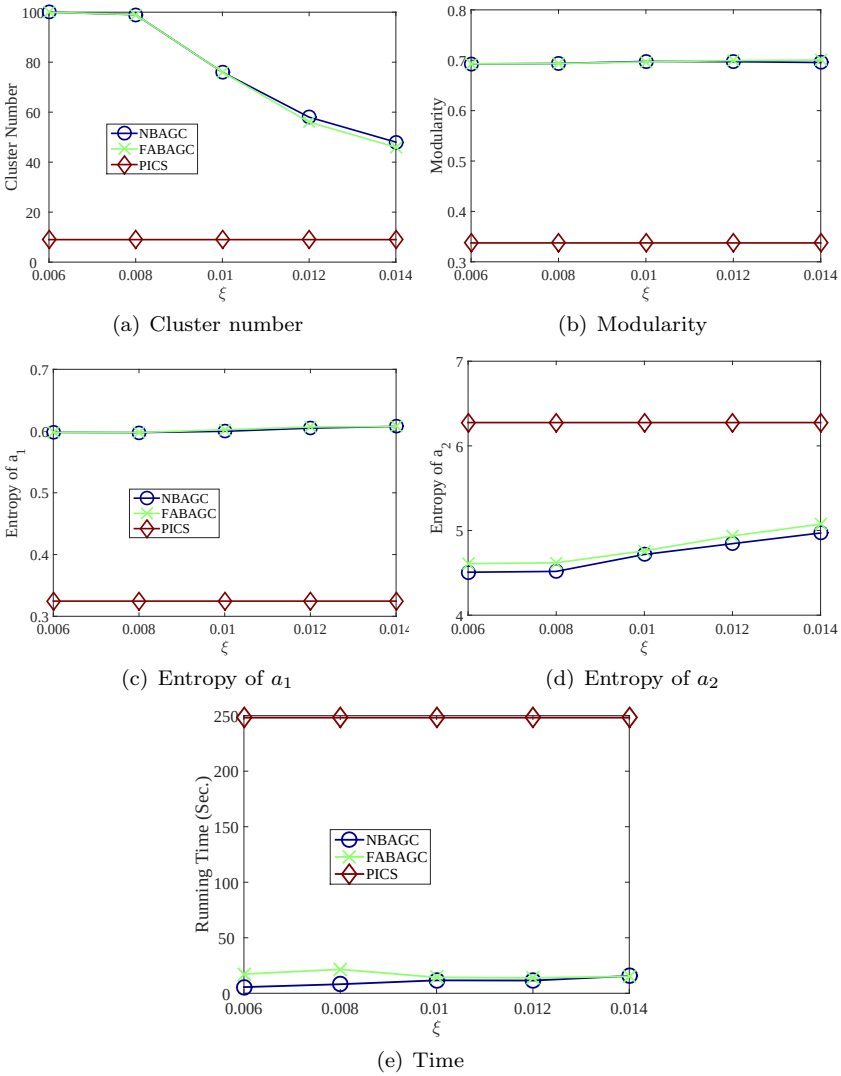
(c) Entropy (d) Time

Fig. 4. Influence of $K^{(0)}$ on clustering Political Blogs ($\xi = 0.006$)



(a) FABAGC (b) PICS

Fig. 5. Community structure on Political Blogs ($K^{(0)} = 40, \xi = 0.006$). On this binary dataset, the yellow color indicates edges in structure part and one of two attribute values in attribute part.



(a) Cluster number (b) Modularity

(c) Entropy of a_1 (d) Entropy of a_2

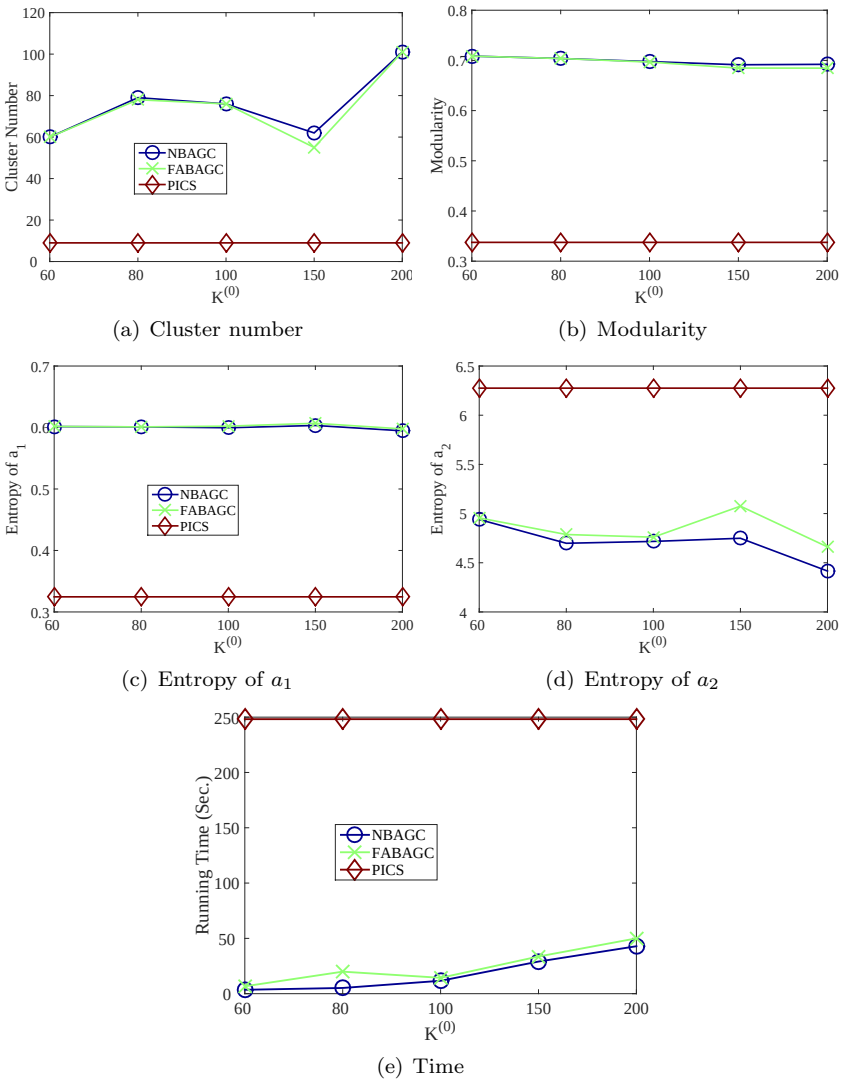
(e) Time

Fig. 6. Influence of ξ on clustering DBLP10K ($K^{(0)} = 100$)

homogeneous attribute values are observed within clusters, while the high entropy value with PICS is verified by nearly equally distributed attribute values within the biggest cluster on the top.

7.3.2. Performance on DBLP10K

We now evaluate our approaches on the DBLP10K dataset. As with the previous experiment, we first fix $K^{(0)} = 100$ and let $\xi = \{0.006, 0.008, 0.01, 0.012, 0.014\}$, and then fix $\xi = \{0.01\}$ and let $K^{(0)} = \{60, 80, 100, 150, 200\}$. Figures 6 and 7 report the clustering results, which are similar for the two settings. The change in cluster number for our approaches now is that they obtain a greater cluster num-



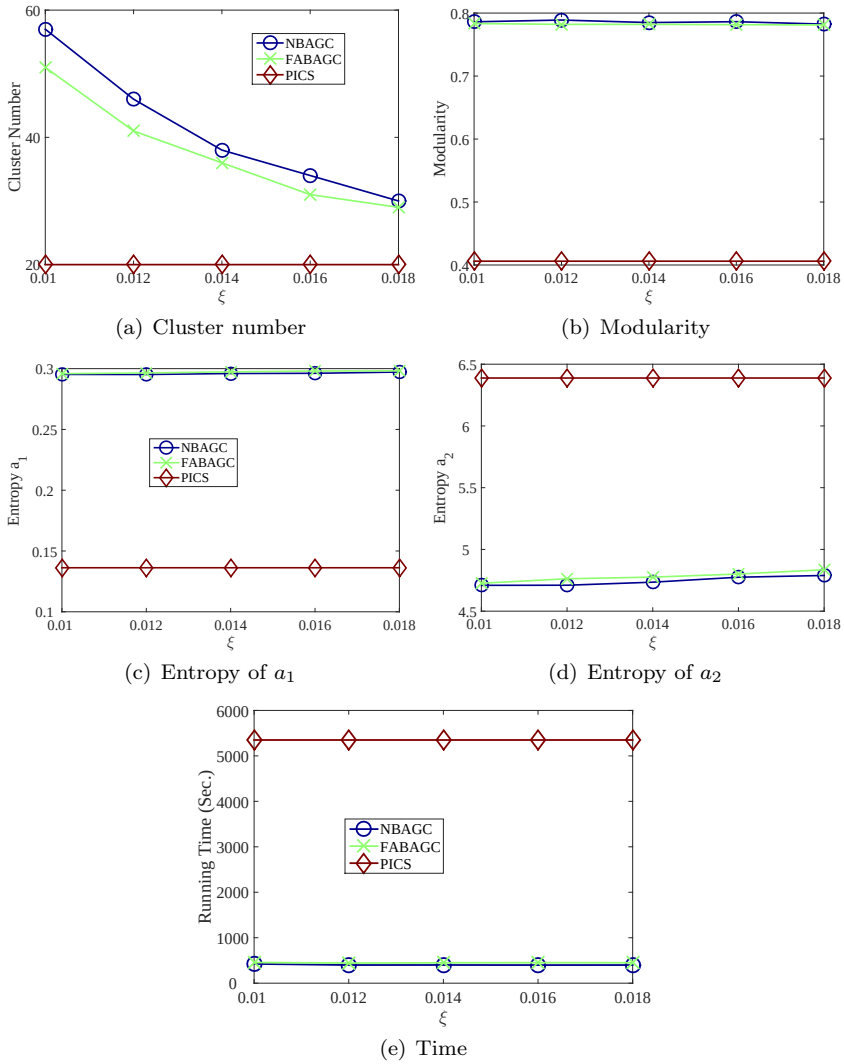
(a) Cluster number (b) Modularity

(c) Entropy of a_1 (d) Entropy of a_2

(e) Time

Fig. 7. Influence of $K^{(0)}$ on clustering DBLP10K ($\xi = 0.01$)

ber than PICS. The greater cluster number is reasonable, because the authors from the four large research communities should be categorized into much finer research sub-communities when authors' research topics are considered together with their collaboration relationship. PICS obtains a higher modularity than in the case of the Political Blogs dataset, because the DBLP dataset belongs to social networks which are often assortative (Newman, 2002). However, our approaches still achieve a much higher modularity value than PICS. For entropy, our approaches and PICS present different tradeoffs between two attributes: our approaches attain a higher entropy value for attribute a_1, i.e., *prolific*, than PICS; while PICS attains a higher entropy value for attribute a_2, i.e., *primary topic*,



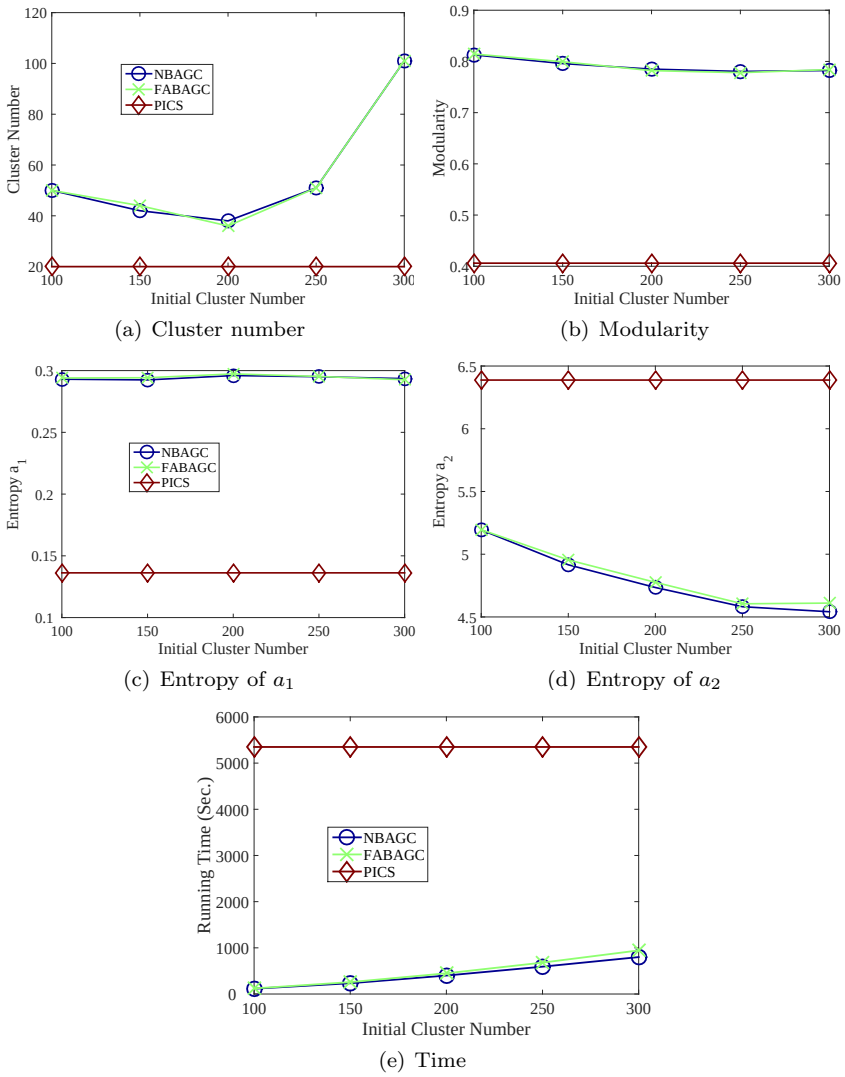
(a) Cluster number (b) Modularity

(c) Entropy of a_1 (d) Entropy of a_2

(e) Time

Fig. 8. Influence of ξ on clustering DBLP84K ($K^{(0)} = 200$)

than ours. We note that the attribute "*prolific*" has 3 values, while the attribute "*primary topic*" consists of 100 research topics. As a result, PICS is prone to getting the vertices with the same value in attribute a_1 clustered together, since it is easier to attain a lower encoding cost for attribute a_1 with only 3 values than for attribute a_2 with 100 values. In contrast, for our approaches, putting authors with the same research topics into the same cluster will give a higher likelihood than putting those of the same prolificacy. In terms of the clustering efficiency, both NBAGC and FABAGC are significantly more efficient than PICS.



(a) Cluster number (b) Modularity

(c) Entropy of a_1 (d) Entropy of a_2

(e) Time

Fig. 9. Influence of $K^{(0)}$ on clustering DBLP84K ($\xi = 0.014$)

7.3.3. Performance on DBLP84K

Last we examine the performance of our algorithms on the largest DBLP84K dataset in comparison with PICS. First, we fix $K^{(0)} = 200$ and vary ξ in $\{0.01, 0.012, 0.014, 0.016, 0.018\}$. The performance results are reported in Figure 8, in terms of the number of final clusters, modularity, attribute entropy and running time. On the number of final clusters, we observe consistencies with previous datasets as follows. It decreases with cluster granularity parameterized by ξ, and slightly fewer clusters are returned by FABAGC than those by NBAGC. Importantly, both NBAGC and FABAGC find more reasonable num-

ber of clusters than PICS which outputs 20 clusters, because 20 clusters seem not enough to yield a good attributed graph clustering in our definition. The reason is that scholars in this dataset are from 15 research fields already, which combining with graph structure and node attributes dictates that it's more likely that a good clustering would contain more than 20 clusters. Based on modularity values returned by each algorithm, this dataset presents a stronger community structure than previous ones. However, our algorithms, NBAGC and FABAGC, still maintain the advantage of a large performance gap about 0.4 over PICS in modularity, which again shows that the communities found by them are significantly stronger than those by PICS. In terms of entropy, similar to the case of DBLP10K dataset, different performance tradeoffs between two attributes occur among our algorithms and PICS on this dataset. We also observe that despite different number of clusters found finally, modularity and entropy values remain quite stable for our algorithms. This is because during the iterative process each cluster to be shrunk are merged into the neighboring one with more similar attribute values (accounting for stable entropy) and there are fewer edges crossing them after sufficient number of iterations (accounting for stable modularity). As for efficiency, our algorithms are two orders of magnitude faster than PICS.

Second, we fix $\xi = 0.014$ and let $K^{(0)}$ vary in $\{100, 150, 200, 250, 300\}$. The results are shown in Figure 9. Similar to the case of the DBLP10K dataset, final cluster numbers are relatively stable compared to those by varying ξ, except for the biggest value of $K^{(0)}$. For $K^{(0)} = 300$, the final cluster number is 100. Since the maximum iteration number is 200 on this dataset, this shows that algorithms have not converged because only one cluster is shrunk for each iteration. However, since other smaller values of $K^{(0)}$ result in a relatively stable final cluster number, it implies that it's not necessary to set a large one like $K^{(0)} = 300$. The nearly stable behavior of modularity and entropy of attribute "prolific" can be interpreted as in the case of varying ξ. For the entropy of attribute "primary topic", the curves seem different on DBLP10K and DBLP84K. On DBLP10K, it roughly follows the rule that bigger final cluster number results in lower entropy value, which is though not inevitable but reasonable because a big cluster number generally means small clusters and thus low diversity of attribute values. The same happens on DBLP84K when $K^{(0)} \geq 200$. However, when $K^{(0)} \leq 200$, it's an opposite case, which may be caused by achieving an overall balance between graph structure and attributes, because the modularity is slightly decreasing accordingly. Efficiency-wise, the time cost for our algorithms increases with $K^{(0)}$, which is expected since larger values of $K^{(0)}$ induce a higher complexity (see Section 5.3). However, their great efficiency advantages over PICS still remain for this case.

8. Conclusions and Future Work

We studied the model selection problem in attributed graph clustering. Unlike most existing works which treat the cluster number as input, we targeted it, together with clustering, as output. We proposed two approaches, nonparametric Bayesian approach and asymptotic approach, based on a Bayesian model for attributed graph clustering, BAGC (Xu et al., 2012). The first approach uses Dirichlet process (Teh, 2010) for model selection, and we developed an efficient algorithm, NBAGC, by variational Bayesian inference. The second approach solves the problem based on an up-to-date model selection criterion,

factorized information criterion (FIC) (Fujimaki and Morinaga, 2012), and we developed an efficient algorithm, FABAGC, by factorized asymptotic Bayesian inference (FAB) (Fujimaki and Morinaga, 2012). Our experimental results verify that our approaches significantly outperform the state-of-the-art algorithm, PICS (Akoglu et al., 2012), in terms of both clustering quality (including cluster number, community structure and attribute homogeneity) and clustering efficiency, for clustering both synthetic datasets (with ground truth) and real datasets.

The future work along this line includes adopting the two approaches to other types of attributed graphs such as those with weighted edges and/or continuous attributes, developing their online versions to accommodate streaming data, extending them to overlapping or subspace clustering with attributed graphs, and so on.

Acknowledgements. The authors would like to thank the anonymous reviewers of the paper for their valuable comments that help significantly improve the quality of the paper.

References

Akoglu, L., Tong, H., Meeder, B. and Faloutsos, C. (2012), Pics: Parameter-free identification of cohesive subgroups in large attributed graphs, *in* 'SDM', pp. 439–450.

Banerjee, B., Bovolo, F., Bhattacharya, A., Bruzzone, L., Chaudhuri, S. and Mohan, B. K. (2015), 'A new self-training-based unsupervised satellite image classification technique using cluster ensemble strategy', *IEEE Geosci. Remote Sensing Lett.* **12**(4), 741–745.

Beal, M. J. (2003), Variational Algorithms for Approximate Bayesian Inference, PhD thesis, Gatsby Computational Neuroscience Unit, University College London.

Bishop, C. M. (2006), *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Bothorel, C., Cruz, J. D., Magnani, M. and Micenková, B. (2015), 'Clustering attributed graphs: models, measures and methods', *CoRR* **abs/1501.01676**.

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

Ester, M., Ge, R., Gao, B. J., Hu, Z. and Ben-Moshe, B. (2006), Joint cluster analysis of attribute data and relationship data: the connected k-center problem, *in* 'SDM'.

Fujimaki, R. and Hayashi, K. (2012), Factorized asymptotic bayesian hidden markov models, *in* 'ICML'.

Fujimaki, R. and Morinaga, S. (2012), Factorized asymptotic bayesian inference for mixture modeling, *in* 'AISTATS', pp. 400–408.

Ghahramani, Z. and Beal, M. J. (1999), Variational inference for bayesian mixtures of factor analysers, *in* 'NIPS', pp. 449–455.

Henderson, K., Eliassi-Rad, T., Papadimitriou, S. and Faloutsos, C. (2010), Hcdf: A hybrid community discovery framework, *in* 'SDM', pp. 754–765.

Henderson, K., Gallagher, B., Eliassi-Rad, T., Tong, H., Basu, S., Akoglu, L., Koutra, D., Faloutsos, C. and Li, L. (2012), Rolx: structural role extraction & mining in large graphs, *in* 'The 18th ACM SIGKDD International Conference

on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012', pp. 1231–1239.

Hofmann, T. (1999), Probabilistic latent semantic indexing, *in* 'SIGIR', pp. 50–57.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. and Saul, L. K. (1999), 'An introduction to variational methods for graphical models', *Machine Learning* **37**(2), 183–233.

Karypis, G. and Kumar, V. (1998), 'A fast and high quality multilevel scheme for partitioning irregular graphs', *SIAM J. Sci. Comput.* **20**(1), 359–392.

Kurihara, K., Welling, M. and Teh, Y. W. (2007), Collapsed variational dirichlet process mixture models, *in* 'IJCAI', pp. 2796–2801.

Lazarsfeld, P. F. and Henry, N. W. (1968), *Latent structure analysis*, Houghton Mifflin.

Lu, Z., Sun, X., Wen, Y., Cao, G. and Porta, T. F. L. (2015), 'Algorithms and applications for community detection in weighted networks', *IEEE Trans. Parallel Distrib. Syst.* **26**(11), 2916–2926.

Luo, G. (2015), 'A review of automatic selection methods for machine learning algorithms and hyper-parameter values'.

Miller, J. W. and Harrison, M. T. (2013), A simple example of dirichlet process mixture inconsistency for the number of components, *in* 'Advances in Neural Information Processing Systems 26', pp. 199–206.

Moser, F., Ge, R. and Ester, M. (2007), Joint cluster analysis of attribute and relationship data without a-priori specification of the number of clusters, *in* 'KDD', pp. 510–519.

Nallapati, R., Ahmed, A., Xing, E. P. and Cohen, W. W. (2008), Joint latent topic models for text and citations, *in* 'KDD', pp. 542–550.

Newman, M. E. (2002), 'Assortative mixing in networks', *Phys. Rev. Lett.* **89**(20), 208701.

Newman, M. E. J. and Girvan, M. (2004), 'Finding and evaluating community structure in networks', *Physical Review E* **69**, 066113.

Ng, A. Y., Jordan, M. I. and Weiss, Y. (2001), On spectral clustering: Analysis and an algorithm, *in* 'Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]', pp. 849–856.

Nowicki, K. and Snijders, T. A. (2001), 'Estimation and prediction for stochastic blockstructures', *Journal of the American Statistical Association* **96**(455), 1077–1087.

Papadopoulos, A., Rafailidis, D., Pallis, G. and Dikaiakos, M. D. (2015), Clustering attributed multi-graphs with information ranking, *in* 'Database and Expert Systems Applications - 26th International Conference, DEXA 2015, Valencia, Spain, September 1-4, 2015, Proceedings, Part I', pp. 432–446.

Semertzidis, T., Rafailidis, D., Strintzis, M. G. and Daras, P. (2015), 'Large-scale spectral clustering based on pairwise constraints', *Inf. Process. Manage.* **51**(5), 616–624.

Steinhaeuser, K. and Chawla, N. V. (2008), Community detection in a large real-world social network, *in* 'Social Computing, Behavioral Modeling, and Prediction', pp. 168–175.

Strehl, A. and Ghosh, J. (2002), 'Cluster ensembles — a knowledge reuse framework for combining multiple partitions', *Journal of Machine Learning Research* **3**, 583–617.

Sun, Y., Aggarwal, C. C. and Han, J. (2012), 'Relation strength-aware clustering of heterogeneous information networks with incomplete attributes', *PVLDB* **5**(5), 394–405.

Teh, Y. W. (2010), Dirichlet process, *in* 'Encyclopedia of Machine Learning', pp. 280–287.

Vretos, N., Solachidis, V. and Pitas, I. (2011), 'A mutual information based face clustering algorithm for movie content analysis', *Image Vision Comput.* **29**(10), 693–705.

Xu, Z. and Ke, Y. (2016*a*), Effective and efficient spectral clustering on text and link data, *in* 'Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016', pp. 357–366.

Xu, Z. and Ke, Y. (2016*b*), 'Stochastic variance reduced riemannian eigensolver', *CoRR* **abs/1605.08233**.

Xu, Z., Ke, Y. and Wang, Y. (2014), A fast inference algorithm for stochastic blockmodel, *in* '2014 IEEE International Conference on Data Mining, ICDM 2014, Shenzhen, China, December 14-17, 2014', pp. 620–629.

Xu, Z., Ke, Y., Wang, Y., Cheng, H. and Cheng, J. (2012), A model-based approach to attributed graph clustering, *in* 'SIGMOD Conference', pp. 505–516.

Xu, Z., Ke, Y., Wang, Y., Cheng, H. and Cheng, J. (2014), 'GBAGC: A general bayesian framework for attributed graph clustering', *TKDD* **9**(1), 5:1–5:43.

Xu, Z., Zhao, P., Cao, J. and Li, X. (2016), Matrix eigen-decomposition via doubly stochastic riemannian optimization, *in* 'Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016', pp. 1660–1669.

Yang, J., McAuley, J. J. and Leskovec, J. (2013), Community detection in networks with node attributes, *in* 'ICDM', pp. 1151–1156.

Yang, T., Jin, R., Chi, Y. and Zhu, S. (2009), Combining link and content for community detection: a discriminative approach, *in* 'KDD', pp. 927–936.

Yu, S., Yu, K., Tresp, V. and Kriegel, H.-P. (2006), Variational bayesian dirichlet-multinomial allocation for exponential family mixtures, *in* 'ECML', pp. 841–848.

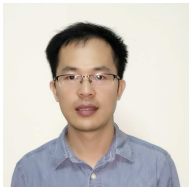
Zanghi, H., Volant, S. and Ambroise, C. (2010), 'Clustering based on random graph model embedding vertex features', *Pattern Recognition Letters* **31**(9), 830–836.

Zhou, T., Lü, L. and Zhang, Y. (2009), 'Predicting missing links via local information', *The European Physical Journal B-Condensed Matter and Complex Systems* **71**(4), 623–630.

Zhou, Y., Cheng, H. and Yu, J. X. (2009), 'Graph clustering based on structural/attribute similarities', *PVLDB* **2**(1), 718–729.

Zobay, O. (2009), 'Mean field inference for the dirichlet process mixture model', *Electronic Journal of Statistics* **3**, 507–545.

Author Biographies



Zhiqiang Xu received his Ph.D. degree in computer engineering from the Nanyang Technological University, Singapore. He is currently a postdoctoral fellow with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology. Prior to that, he was a research scientist with the Institute of Infocomm Research, A*STAR, Singapore. His research interests include data mining and machine learning.

James Cheng is with the Department of Computer Science and Engineering, the Chinese University of Hong Kong, Hong Kong. His current research interests include big data infrastructures and applications, distributed computing, distributed machine learning, and large-scale network analysis.

Xiaokui Xiao received the PhD degree in computer science from the Chinese University of Hong Kong in 2008. He is currently an associate professor at the Nanyang Technological University (NTU), Singapore. Before joining NTU in 2009, he was a postdoctoral associate at the Cornell University. His research interests include data privacy and spatial databases.

Ryohei Fujimaki (Ph.D.) is a research fellow at NEC's Data Science Research Laboratories. He is in charge of developing NEC's advanced analytics technologies and their applications. In addition to his industrial and business contributions, he has been publishing high-quality papers in such premier conferences as KDD, NIPS, ICML and so on.

Yusuke Muraoka is a researcher of NEC Corporation. He received MS degree in Mathematical Informatics from University of Tokyo in 2009. His research area is machine learning, data mining and their business applications such as advanced analytics and predictive analytics.

Correspondence and offprint requests to: Zhiqiang Xu, Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Saudi Arabia. Email: zhiqiang.xu@kaust.edu.sa