Evolutionary Clustering by Hierarchical Dirichlet Process with Hidden Markov State

Tianbing Xu[1] Zhongfei (Mark) Zhang[1]
[1]Dept. of Computer Science
State Univ. of New York at Binghamton
Binghamton, NY 13902, USA
{txu,zhongfei,blong}@cs.binghamton.edu

Philip S. Yu[2] Bo Long[1]
[2]Dept. of Computer Science
Univ. of Illinois at Chicago
Chicago, IL 60607, USA
psyu@cs.uic.edu

Abstract

This paper studies evolutionary clustering, which is a recently hot topic with many important applications, noticeably in social network analysis. In this paper, based on the recent literature on Hierarchical Dirichlet Process (HDP) and Hidden Markov Model (HMM), we have developed a statistical model HDP-HTM that combines HDP with a Hierarchical Transition Matrix (HTM) based on the proposed Infinite Hierarchical Hidden Markov State model (iH^{2}MS) as an effective solution to this problem. The HDP-HTM model substantially advances the literature on evolutionary clustering in the sense that not only it performs better than the existing literature, but more importantly it is capable of automatically learning the cluster numbers and structures and at the same time explicitly addresses the correspondence issue during the evolution. Extensive evaluations have demonstrated the effectiveness and promise of this solution against the state-of-the-art literature.

1 Introduction

Evolutionary clustering is a recently identified new and hot research topic in data mining. Evolutionary clustering addresses the evolutionary trend development regarding a collection of data items that evolves over the time. From time to time, with the evolution of the data collection, new data items may join the collection and existing data items may leave the collection; similarly, from time to time, cluster structure and cluster number may change during the evolution. Due to the nature of the evolution, model selection must be solved as part of a solution to the evolutionary clustering problem at each time. Consequently, evolutionary clustering poses a greater challenge than the classic clustering problem as many existing solutions to the latter problem typically assume that the model selection, is still an open problem in the clustering literature.

In evolutionary clustering, one of the most difficult and challenging issues is to solve the correspondence problem. The correspondence problem refers to the correspondence between different local clusters across the times due to the evolution of the distribution of the clusters, resulting in cluster-cluster correspondence and cluster transition correspondence issues. All the existing methods in the literature fail to address the correspondence problems explicitly.

On the other hand, solutions to the evolutionary clustering problem have found a wide spectrum of applications for trend development analysis, social network evolution analysis, and dynamic community development analysis. Potential and existing applications include daily news analysis to observe news focus change, blog analysis to observe community development, and scientific publications analysis to identify the new and hot research directions in a specific area. Consequently, evolutionary clustering has recently become a very hot and focused research topic.

In this paper, we have shown the new statistical model HDP-HTM that we have developed as an effective solution to the evolutionary clustering problem. In this new model, we assume that the cluster structure at each time is a mixture model of the clusters for the data collection at that time; in addition, clusters at different times may share common clusters, resulting in explicitly addressing the cluster-cluster correspondence issue. we adopt the Hierarchical Dirichlet Processes (HDP) [26] with a set of common clusters at the top level of the hierarchy and the local clusters at the lower level at each different times sharing the top level clusters. Further, data and clusters evolve over the time with new clusters and new data items possibly joining the collection and with existing clusters and data items possibly leaving the collection at different times, leading to the cluster structure and the number of clusters evolving over the time. Here, we use the state transition matrix to explicitly reflect the cluster-to-cluster transitions between different times, re-

sulting in explicitly effective solution to the cluster transition correspondence issue. Consequently, we propose the Infinite Hierarchical Hidden Markov State model (iH^{2}MS) to construct the Hierarchical Transition Matrix (HTM) at different times to capture the cluster-to-cluster transition evolution.

The specific contributions of this work are highlighted as follows: (1) We have applied the recent literature on Dirichlet process and Hidden Markov Model (HMM) to solve the evolutionary clustering problem with a specific new model HDP-HTM. (2) This new model as an effective solution to evolutionary clustering substantially advances the literature in the sense that it is capable of automatically learning the number of clusters and the cluster structure at each time and at the same time addressing the cluster correspondence problem explicitly during the evolution, which makes this solution practical in many evolutionary clustering applications. (3) We have demonstrated the superiority of this solution to the existing state-of-the-art literature in both synthetic data and real Web daily news data in the case of the evolutionary document clustering application.

2 Related Work

Evolutionary Clustering, contrast to static clustering, is an emerging research topic, which processes temporal data to generate a sequence of clusterings across the time. In data mining community, Chakrabarti et al. [9] were the first to address this problem. They proposed a framework to capture the history information quantitatively measured by the *history quality*, and the current information measured by the *snapshot quality*. Within this framework, they designed two specific algorithms: evolutionary k-mean and evolutionary agglomerative hierarchical clustering. Later, Chi et al. [10] proposed two evolutionary spectral clustering algorithms PCM and PCQ, by incorporating the *temporal smoothness* to regularize a cost function. These methods significantly advance the evolutionary clustering literature; however, they are not able to handle the very typical scenario where the number of clusters at different times varies with the evolution. Recently, Tang et al. [25] proposed an evolutionary clustering solution, based on spectral clustering framework, to handle dynamic multi-mode networks problems, where both actor memberships and interactions evolve over the time. However, the solution still assumes a fixed number of communities(clusters) over the time.

Dirichlet Process (DP) is a distribution over the probability measure on the parameter space Θ first discussed by Ferguson [13]. An explicit stick-breaking construction definition of DP is given by Sethuraman [23] as follows:

$$G = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$$

where G is a random measure drawn from $DP(\alpha, H)$ with the concentration parameter α, δ_{ϕ_k} is the concentration measure on ϕ_k sampling from the base measure H, β_k is the "stick-breaking weights" discussed in detail later. Blackwell and MacQueen [5] have introduced another representation of DP, as the predictive distribution of the events is proportional to the frequency of an existing event or to a concentration parameter for an unpresented event. Similarly, this representation may be further illustrated as the Chinese Restaurant Process (CRP) [1] to explicitly demonstrate the "clustering property" as the "distribution on partition".

In the mixture models, when the number of the mixture components is unknown *a priori*, DP may be incorporated as a prior distribution on the parameters formulated as the Dirichlet Process Mixture Model (DPM) [2, 12], which is able to handle an *infinite* or *arbitrary* number of components and to automatically learn such a number from the observations. In order to share the mixture components across the groups of the mixture models, Teh et al. [26] have proposed the Hierarchical Dirichlet Process model, where a hierarchical Bayesian approach is used to place a global Dirichlet process prior on the parameter space as the discrete base measure for the lower level DPMs.

There are many noticeable applications of DP based models in text modeling and topic analysis. Blei et al. [7] proposed the well-known Latent Dirichlet Allocation (LDA), for the text modeling and clustering with a known, constant number of the topics set in advance. For the topic evolution analysis, Blei et al. [6] have designed the probabilistic models to develop reasonable solutions. Based on LDA, Griffiths et al. [16] tried to identify "hot topics" and "cold topics" by the text temporal dynamics. The number of the topics was decided by the Bayesian Model selection. Recently, Wang et al. [27] introduced a LDA-style topic model to represent the time as an observed continue variable attempting to capture the topic evolutionary trends. However, all these models failed to automatically learn the number of the topics (i.e., the clusters) according to the underlying data evolutionary distribution. Further, they failed to address the correspondence issues during the evolution.

Hidden Markov Model (HMM) [22] has been widely used to model sequential or temproal data. HMM has a *finite* hidden state space, governed by a *finite* transition probability matrix which obeys the Markov dynamics, and a sequence of the observations generated by these hidden states along the time. When turning to an *infinite* or *arbitrary* hidden state space, Beal et al. [4] have first introduced the Infinite Hidden Markov Model (iHMM) also known as HDP-HMM by using the Dirichlet Process as a row in the state transition matrix. Teh et al. [26] have demonstrated that HDP may recast iHMM and have provided the Gibbs sampling inference mechanism for it.

Recently, Fox et al. [14] have revisited the HDP-HMM

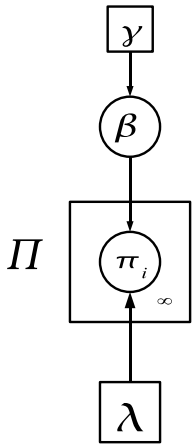


Figure 1. The iH^{2}MS Model

and have developed methods which allow more efficient and effective learning from realistic time series data. Ni et al. [21] have proposed a new hierarchical nonparametric Bayesian model by imposing a nested Dirichlet Process prior on the base distributions of the iHMMs to learn the sequential data. The standard approach to learning HMM is an EM-based algorithm [11] specifically known as Baum-Welch algorithm [3]. MacKay [19] has introduced a Bayesian learning procedure called the variational approximation to handle the overfitting problem in Baum-Welch algorithm. More recently, Gael et al. [15] have introduced a new inference algorithm for iHMM called the beam sampling which is more efficient and robust. Xu et al. [28] proposed two models DPChain and HDP-EVO as solutions to evolutionary clustering. Both models are able to learn the number of clusters and dynamic structures during the evolution.

3 Infinite Hierarchical Hidden Markov State Model(iH^{2}MS)

Here, we propose a new infinite hierarchical hidden Markov state model (iH^{2}MS) for Hierarchical Transition Matrix (HTM) and provide an update construction scheme based on this model. Figure 1 illustrates this model.

3.1 iH^{2}MS Representation

Tradtionally, Hidden Markov Model (HMM) has a *finite* state space with K hidden states, say $\{1, 2, \dots K\}$. For the hidden state sequence $\{s_1, s_2, \dots s_T\}$ up to time T, there is a K by K state transition probability matrix $\mathbf{\Pi}$ governed by Markov dynamics with all the elements $\pi_{i,j}$ of each row $\boldsymbol{\pi}_i$ summed to 1.

$$\pi_{i,j} = p(s_t = j | s_{t-1} = i)$$

The initial state probability for state i is $p(s_1 = i)$ with the summation of all the initial probabilities equal to 1. For observation x_t in the observation sequence $\{x_1, x_2, \dots x_T\}$, given state $s_t \in \{1, 2, \dots, K\}$, there is a parameter ϕ_{s_t} drawn from the base measure H which parameterizes the observation likelihood probability.

$$x_t | s_t \sim F(\phi_{s_t})$$

However, when dealing with a countable *infinite* state space, $\{1, 2, \dots K, \dots\}$, we must adopt a new model similar to that in [4] for a state transition probability matrix with an *infinite* matrix dimension. Thus, the dimension of the state transition probability matrix now has become infinite. $\boldsymbol{\pi}_i$, the i-th row of the transition probability matrix $\mathbf{\Pi}$, may be represented as the mixing proportions for all the next infinite states, given the current state. Thus, we model it as a DP with an infinite dimension with the summation of all the elements in a row normalized to 1, which leads to an infinite number of DPs' construction for an infinite transition probability matrix.

With no further prior knowledge on the state sequence, a typical prior for the transition probability may be the symmetric Dirichlet distributions. Similar to [26], we intend to construct a hierarchical Dirichlet model to keep different rows of the transition probability matrix to share part of the prior mixing proportions of each state at the top level. Consequently, we adopt a new state model, Infinite Hierarchical Hidden Markov State model (iH^{2}MS), to construct the Infinite Transition Probability Matrix which is called the Hierarchical Transition Matrix (HTM).

Similar to HDP [26], we draw a random probability measure on the infinite state space $\boldsymbol{\beta}$ as the top level prior from $stick(\gamma)$ represented as the mixing proportions of each state.

$$\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty} \quad \beta_k = \beta_k{}' \prod_{l=1}^{k-1} (1 - \beta_l{}') \quad \beta_k{}' \sim Beta(1, \gamma) \tag{1}$$

Here, the mixing proportion of state k, β_k, may also be interpreted as the prior mean of the transition probabilities leading to state k. Hence, $\boldsymbol{\beta}$ may be represented as the prior random measure of a transition probability DP.

For the $i-$th row of the transition matrix $\mathbf{\Pi}$, $\boldsymbol{\pi}_i$, we sample it from $DP(\lambda, \boldsymbol{\beta})$ with a smaller concentration parameter λ implying a larger variability around the mean measure $\boldsymbol{\beta}$. The stick-breaking representation for $\boldsymbol{\pi}_i$ is as follows:

$$\boldsymbol{\pi}_i = (\pi_{i,k})_{k=1}^{\infty} \quad \pi_{i,k} = \pi_{i,k}{}' \prod_{l=1}^{k-1} (1 - \pi_{i,l}{}') \quad \pi_{i,k}{}' \sim Beta(1, \lambda) \tag{2}$$

Specifically, $\pi_{i,k}$ is the state transition probability from the previous state i to the current state k as $p(s_t = k|s_{t-1} = i)$.

Now, each row of the transition probability matrix is represented as a DP which shares the same reasonable prior on the mixing proportions of the states. For a new row corresponding to a new state k, we simply draw a transition probability vector $\boldsymbol{\pi}_k$ from $DP(\lambda, \boldsymbol{\beta})$, resulting in constructing a countably infinite transition probability matrix.

3.2 Extention of iH^{2}MS

The transition probability constructed by iH^{2}MS may be further extended to the scenario where there are more than one state at each time. Suppose that there is a countably infinite global state space $\boldsymbol{S} = \{1, 2, \ldots, K, \ldots\}$ including states in all the state space $\boldsymbol{S}_t$ at each time t, where $\boldsymbol{S}_t \subseteq \boldsymbol{S}$. For any state $s_t \in \boldsymbol{S}_t$ at time t and state $s_{t-1} \in \boldsymbol{S}_{t-1}$ at time $t-1$, we may adopt the transition probability $\pi_{i,k}$ to represent $p(s_t = k|s_{t-1} = i)$. Similarly, $\pi_{i,k}$ here still has the property that there is a natural tendency for a transition to appear more frequently at the current time if such a transition appears more frequently at a previous time. Therefore, it is reasonable to model a row of transition as a DP with an infinite dimension. We will discuss this extension in detail later.

3.3 Maximum Likelihood Estimation of HTM

Let $\boldsymbol{X}$ be the observation sequence, which includes all the observations $\boldsymbol{X}_t$ at each time t, where $\boldsymbol{X}_t \subseteq \boldsymbol{X}$. Now, the question is how to represent the countably infinite state space in a hierarchical state transition matrix (HTM). Notice that, at each time, there is in fact a finite number of observations $\boldsymbol{X}_t$; the state space $\boldsymbol{S}_t$ at each time t must be arbitrarily finite even though conceptually the global state space $\boldsymbol{S}$ may be considered as countably infinite. Further, we adopt the stick-breaking representation for the DP prior [26, 18] to iteratively handle an arbitrary number of the states and accordingly the transition probability matrix up to time t.

Suppose that up to time t there are K current states and we use $K+1$ to index a potentially new state. Then $\boldsymbol{\beta}$ may be represented as:

$$\boldsymbol{\beta} = \{\beta_1, \ldots \beta_K, \beta_u\} \quad \beta_u = \sum_{k=K+1}^{\infty} \beta_k \quad \sum_{k=1}^{K} \beta_k + \beta_u = 1 \tag{3}$$

Given $\boldsymbol{\beta}$, the Dirichlet prior measure of i-th row of the transition probability matrix $\boldsymbol{\pi}_i$ has a dimension $K+1$. The last element β_u is the prior measure of the transition probability from state i to an unpresented state u. The prior distribution of $\boldsymbol{\pi}_i$ is $Dir(\lambda\beta_1, \ldots \lambda\beta_K, \lambda\beta_u)$.

When a new state is instantiated, we sample b from $Beta(1, \gamma)$, and set the new proportions for the new state $K^{new} = K+1$ and another potentially new state $K^{new}+1$ as:

$$\beta_{K^{new}} = b\beta_u \quad \beta_u^{new} = (1-b)\beta_u \tag{4}$$

Now, K is updated as K^{new}, β_u as β_u^{new}, and the number of the states may continue to increase if yet another new state is instantiated, resulting in a countably infinite transition probability matrix.

Since up to time t, there are K states in the hidden state space, it is possible to adopt Baum-Welch algorithm [22, 19] to estimate the posterior of the transition probability matrix $\boldsymbol{\Pi}$ by the maximum likelihood Optimization. Similar to [19], we have a Dirichlet prior on each row of $\boldsymbol{\Pi}$, $\boldsymbol{\pi}_i$. Consequently, we have the M-step for updating π_{ik} according to the standard Baum-Welch optimization equation:

$$\pi_{i,k} = p(s_t = k|s_{t-1} = i) = \begin{cases} \frac{n_{i,k}^t + \lambda\beta_k}{n_i^t + \lambda} & k \text{ is an exisiting state} \\ \frac{\lambda\beta_u}{n_i^t + \lambda} & k \text{ is a new state} \end{cases} \tag{5}$$

where approximately $n_{i,k}^t$ is the expected number of the transitions from state i to state k up to time t, n_i^t is the expected number of the transitions out of state i up to time t. For a state of any common observation x between two adjacent times τ and $\tau+1$ up to time t,

$$n_{i,k}^t = \sum_{\tau=1}^{t-1} \delta(s_{\tau,x}, i)\delta(s_{\tau+1,x}, k) \quad n_i^t = \sum_{k=1}^{K} n_{i,k}^t$$

where $s_{\tau,x}$ and $s_{\tau+1,x}$ capture the correpondence between the states with the same data item at adjacent times. Here we use Kronecker-delta function ($\delta(a,b) = 1$ iff $a = b$ and 0 otherwise) to count the number of the state transitions for all the common observations up to time t.

Conceptually, in Eq. 5 we may consider $\lambda\beta_k$ as the pseudo-observation of the transition from state i to k (i.e., the strength of the belief for the prior state transition), and $\lambda\beta_u$ as the probability of a new state transitioned from state i. Eq. 5 is equivalent to Blackwell-MacQueen urn Scheme [5] to sample a state. Thus, this posterior maximum likelihood estimation of the transition probability to a state is equal to probability of such a state posterior sampled by the polya urn scheme [5] given a sequence of the states and observations up to time t.

4 HDP Incorporated with HTM (HDP-HTM)

To capture the state(cluster) transition correspondence during the evolution at different times, we propose the HTM; at the same time, we must capture the state-state

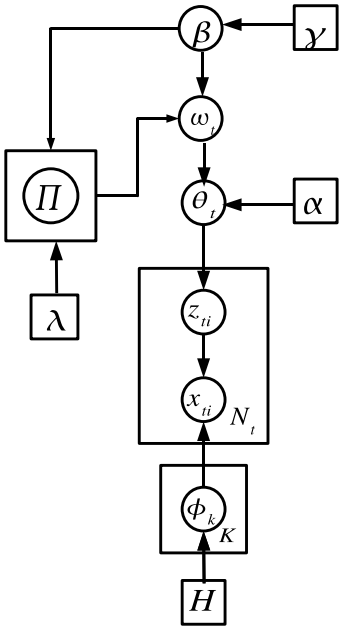


Figure 2. The HDP-HTM Model

(cluster-cluster) correspondence, which may be handled by a hierarchical model with the top level corresponding to the global states [1] and the lower level corresponding to the local states, where it is natural to model the statistical process as HDP [26]. Consequently, we propose to combine HDP with HTM as a new HDP-HTM model, as illustrated in Figure 2.

4.1 Model Representation

Let the global state space $\boldsymbol{S}$ denote the global cluster set, which includes all the states $\boldsymbol{S}_t \subseteq \boldsymbol{S}$ at all the times t. The global observation set $\boldsymbol{X}$ includes all the observations $\boldsymbol{X}_t$ at each time t, of which each data item i is denoted as $x_{t,i}$.

We draw the global mixing proportion from the global states $\boldsymbol{\beta}$ with the stick-breaking representation using the concentration parameter γ from Eq. 1. The global measure G_0 may be represented as:

$$G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$$

where ϕ_k is drawn from the base probability measure H with pdf h, and δ_{ϕ_k} is the concentration measure on ϕ_k.

Different from HDP, here we must consider the evolution of the data and the states (i.e., the clusters). The distribution of the clusters at time t is not only governed by the global measure G_0, but also is controlled by the data and cluster evolution in the history. Consequently, we make an assumption that the data and the clusters at time t are generated from the previous data and clusters, according to the mixture proportions of each cluster and the transition probability matrix. The global prior mixture proportions for the clusters is $\boldsymbol{\beta}$, and the state transition matrix $\boldsymbol{\Pi}$ provides the information of the previous state evolution in the history up to time t. Now, the expected number of the data items generated by cluster k is proportional to the number of data items in the clusters in the history multiplied by the transition probabilities from these clusters to state k; specificially, the mean mixture proportion for cluster k at time t, $\boldsymbol{\omega}_t$, is defined as follows:

$$\omega_{t,k} = \sum_{j=1}^{\infty} \beta_j \pi_{j,k}$$

More precisely, $\boldsymbol{\omega}_t$ is further obtained by:

$$\boldsymbol{\omega}_t = \boldsymbol{\beta} \cdot \boldsymbol{\Pi} \tag{6}$$

Clearly, by the transition probability property, $\sum_{k=1}^{\infty} \omega_{t,k} = 1$, $\sum_{k=1}^{\infty} \pi_{i,k} = 1$ and the stick-breaking property $\sum_{j=1}^{\infty} \beta_j = 1$.

$$\sum_{k=1}^{\infty} \omega_{t,k} = \sum_{k=1}^{\infty} \sum_{j=1}^{\infty} \beta_j \pi_{j,k} = \sum_{j=1}^{\infty} \beta_j \sum_{k=1}^{\infty} \pi_{j,k} = \sum_{j=1}^{\infty} \beta_j = 1$$

Thus, the mean mixture proportion $\boldsymbol{\omega}_t$ may be taken as the new probability measure at time t on the global cluster set. With the concentration parameter α, we draw the mixture proportion vector $\boldsymbol{\theta}_t$ from $DP(\alpha, \boldsymbol{\omega}_t)$.

$$\boldsymbol{\theta}_t | \alpha, \boldsymbol{\omega}_t \sim DP(\alpha, \boldsymbol{\omega}_t)$$

Now, at time t, the local measure G_t shares the global clusters parameterized by $\boldsymbol{\phi} = (\phi_k)_{k=1}^{\infty}$ with the mixing proportion vector $\boldsymbol{\theta}_t$.

$$G_t = \sum_{k=1}^{\infty} \theta_{t,k} \delta_{\phi_k}$$

At time t, given the mixture proportion of the clusters $\boldsymbol{\theta}_t$, we draw a cluster indicator $z_{t,i}$ for data item $x_{t,i}$ from a multinomial distribution:

$$z_{t,i} | \boldsymbol{\theta}_t \sim Mult(\boldsymbol{\theta}_t)$$

Once we have the cluster indicator $z_{t,i}$, data item $x_{t,i}$ may be drawn from distribution F with pdf f, parameterized by $\boldsymbol{\phi}$ from the base measure H.

$$x_{t,i} | z_{t,i}, \boldsymbol{\phi} \sim f(x | \phi_{z_{t,i}})$$

Finally, we summarize the data generation process for HDP-HTM as follows.

[1] Each state is represented as a distinct cluster.

1. Sample the cluster parameter vector $\boldsymbol{\phi}$ from the base measure H. The number of the parameters is unknown *a priori*, but is determined by the data when a new cluster is needed.

2. Sample the global cluster mixture vector $\boldsymbol{\beta}$ from $stick(\gamma)$.

3. At time t, compute the mean measure $\boldsymbol{\omega}_t$ for the global cluster set by $\boldsymbol{\beta}$ and $\boldsymbol{\Pi}$ according to Eq. 6.

4. At time t, sample the local mixture proportion $\boldsymbol{\theta}_t$ by $DP(\alpha, \boldsymbol{\omega}_t)$.

5. At time t, sample the cluster indicator $z_{t,i}$ from $Mult(\boldsymbol{\theta}_t)$ for data item $x_{t,i}$.

6. At time t, sample data item $x_{t,i}$ from $f(x|\phi_{z_{t,i}})$ given cluster indicator $z_{t,i}$ and parameter vector $\boldsymbol{\phi}$.

4.2 Model Learning

We denote $n_{i,j}^t$ as the number of the state transitions from state i to j between two adjacent times up to time t. Let $n_{t,k}$ be the number of the data items belonging to cluster k at time t, $n_{t,k}^{-i}$ be the number of the data items belonging to cluster k except $x_{t,i}$ at time t, and n_t be the number of all the data items at time t. Similar to HDP [26], let $m_{t,k}$ be the number of the tables (i.e., the local clusters) belonging to the global cluster k at time t, and m_k be the number of the tables (i.e., the local clusters) belonging to the global cluster k across all the times. Finally, let $\boldsymbol{x}_t$ be the data collection at time t.

In order to handle an infinite or arbitrary number of the states (i.e., clusters), we adopt the stick-breaking mechanism similar to what we have done in Sec. 3.3. Assume that there are K existing clusters. The global mixture proportion $\boldsymbol{\beta} = \{\beta_1, \ldots, \beta_K, \beta_u\}$ with β_u being the proportion for an unrepresented cluster; when a new cluster is instantiated, the vector $\boldsymbol{\beta}$ is updated according to the stick-breaking construction in Eq. 4 to ensure the normalized summation equal to 1 with the probability 1. In addition, the transition probability matrix $\boldsymbol{\Pi}$ is in the dimension of $K+1$ by $K+1$, resulting in $\boldsymbol{\omega}_t$ also in dimension of 1 by $K+1$ with the last element $\omega_{t,u}$ as the proportion of the unrepresented cluster.

Here, we adopt the EM [11] framework to learn the model by combining Markov Chain Monte Carlo (MCMC) method [20] to make an inference for the auxiliary variable $\boldsymbol{m}$ and other necessary variables at the E step and maximum likelihood estimation of $\boldsymbol{\Pi}$ at the M step. Similar to HDP with the direct assignment posterior sampling, we also need to sample the global mixture proportion $\boldsymbol{\beta}$ from $\boldsymbol{m}$. We update the transition probability matrix $\boldsymbol{\Pi}$ by the counter statistic $n_{i,j}^t$ up to time t according to Eq. 5. We no longer need to sample $\boldsymbol{\theta}_t$ because we may just sample the cluster assignment $\boldsymbol{z}_t$ at time t by integrating out $\boldsymbol{\theta}$. Similarly, by the conjugacy of h and f, it is not necessary to sample the parameter ϕ_k for cluster k.

Sampling $\boldsymbol{z}_t$

Since $\boldsymbol{\theta}_t$ is distributed as $DP(\alpha, \boldsymbol{\omega}_t)$, while the cluster indicator $z_{t,i}$ is in a multinomial distribution with the parameter $\boldsymbol{\theta}_t$, it is convenient to integrate out $\boldsymbol{\theta}_t$ by the conjugacy property. Thus, the conditional probability of the current cluster assignment $z_{t,i}$ for the current data item $x_{t,i}$ given the other assignments $\boldsymbol{z}_{t,-i} = \boldsymbol{z}_t \setminus z_{t,i}$ and the Dirichlet parameters $\boldsymbol{\omega}_t$ and α is:

$$p(z_{t,i} = k|\boldsymbol{z}_{t,-i}, \boldsymbol{\omega}_t, \alpha) = \begin{cases} \frac{n_{t,k}^{-i} + \alpha\omega_{t,k}}{n_t - 1 + \alpha} & \text{if } k \text{ is an existing cluster} \\ \frac{\alpha\omega_{t,u}}{n_t - 1 + \alpha} & \text{if } k \text{ is a new cluster} \end{cases} \tag{7}$$

By Gibbs Sampling [8], we need to compute the conditional probability $z_{t,i}$ given the other cluster assignment $\boldsymbol{z}_{t,-i}$, the observation $\boldsymbol{x}_t$ at time t, and the Dirichlet parameters $\boldsymbol{\omega}_t$ and α.

$$p(z_{t,i} = k|\boldsymbol{z}_{t,-i}, \boldsymbol{x_t}, \boldsymbol{\omega}_t, \alpha) = \begin{cases} \frac{n_{t,k}^{-i} + \alpha\omega_{t,k}}{n_t - 1 + \alpha} f_k^{-i}(x_{t,i}) & \text{if } k \text{ is an existing cluster} \\ \frac{\alpha\omega_{t,u}}{n_t - 1 + \alpha} f_k^{-i}(x_{t,i}) & \text{if } k \text{ is a new cluster} \end{cases} \tag{8}$$

where $f_k^{-i}(x_{t,i})$ is the conditional likelihood of $x_{t,i}$ given the other data items $\boldsymbol{x}_{t,-i}$ under cluster k, which by the conjugacy property of h and f could be computed by integrating out the cluster parameter ϕ_k for cluster k.

$$f_k^{-i}(x_{t,i}) = \int f(x_{t,i}|\phi_k) \cdot h(\phi_k|\{x_{t,j} : z_{t,j} = k, j \neq i\}) d\phi_k \tag{9}$$

Sampling $\boldsymbol{m}$

Again similar to HDP, in order to sample $\boldsymbol{m}$, we must first sample $\boldsymbol{m}_t$, the number of the tables (i.e., the local clusters) for the clusters at time t [26]. After sampling of $\boldsymbol{z}_t$, $n_{t,k}$ is updated accordingly. By [2, 26], we may sample $\boldsymbol{m}$ according to the following Gibbs Sampling [8]:

$$p(m_{t,k} = m|\boldsymbol{z}_t, \boldsymbol{m}^{-t,k}, \boldsymbol{\beta}, \alpha) = \frac{\Gamma(\alpha\beta_k)}{\Gamma(\alpha\beta_k + n_{t,k})} S(n_{t,k}, m)(\alpha\beta_k)^m \tag{10}$$

where $\boldsymbol{m}^{-t,k} = \boldsymbol{m} \setminus m_{t,k}$

Sampling $\boldsymbol{\beta}$

Given $\boldsymbol{m}$, the posterior distribution of $\boldsymbol{\beta}$ is:

$$\boldsymbol{\beta}|\boldsymbol{m}, \gamma \sim Dir(m_1, \ldots, m_K, \gamma) \tag{11}$$

where K is the number of the existing clusters up to time t. Consequently, it is trivial to sample $\boldsymbol{\beta}$ according to Eq. 11.

Updating the Transition Matrix $\boldsymbol{\Pi}$

After we have the knowledge of the sequence of the states and observations at different times, especially the new knowledge at time t, we may adopt the maximum likelihood

estimation to construct the posterior transition probability matrix $\mathbf{\Pi}$. After sampling $\boldsymbol{z}_t$, the state (i.e., the cluster) assignment at time t is changed, leading to updating $n_{i,j}^t$ accordingly. Consequently, the matrix $\mathbf{\Pi}$ is updated according to Eq. 5.

Hyperparameter Sampling

In the HDP-HTM model, there are the concentration hyperparameters $\Theta = \{\alpha, \gamma, \lambda\}$. According to [26, 12], we may sample these parameters by the Gamma distribution with the constant Gamma parameters discussed in detail in Sec. 5.

Finally, we summarize the EM framework as follows:

1. Initialize the transition matrix $\mathbf{\Pi}$, as well as $\boldsymbol{\beta}$, $\boldsymbol{m}$, and $\boldsymbol{z}_t$; compute $\boldsymbol{\omega}_t$ by taking the product of $\mathbf{\Pi}$ and $\boldsymbol{\beta}$.
2. The E-Step at time t:

   Sample the hyperparameters α, γ, and λ from the Γ distribution.

   Sample $\boldsymbol{z}_t$ based on $\mathbf{\Pi}$, $\boldsymbol{\beta}$, and α according to Eq. 8.

   Sample $\boldsymbol{m}$ based on $\boldsymbol{z}_t$ and $\boldsymbol{\beta}$ according to Eq. 10.

   Sample $\boldsymbol{\beta}$ based on $\boldsymbol{m}$ and γ according to Eq. 11.
3. The M-Step at time t:

   Update $\mathbf{\Pi}$ based on $\boldsymbol{z}_t$, $\boldsymbol{\beta}$, and λ according to Eq. 5.
4. Iterate 2 to 3 until convergence.

5 Experimental Evaluations

We have evaluated the HDP-HTM model in an extensive scale against the state-of-the-art literature. We compare HDP-HTM in performance with evolutionary spectral clustering PCM and PCQ algorithms [10] and HDP [26] for the synthetic data and the real data in the application of document evolutionary clustering; for the experiments in text data evolutionary clustering, we have also evaluated the HDP-HTM model in comparison with LDA [7, 17] in addition. In particular, the evaluations are performed in three datasets, a synthetic dataset, the 20 NewsGroups dataset, and a Google daily news dataset we have collected over a period of 5 continuous days.

5.1 Synthetic Dataset

We have generated a synthetic dataset in a scenario of evolutionary development. The data are a collection of mixture models with the number of the clusters unknown *a priori* with a smooth transition over the time during the evolution. Specifically, we simulate the scenario of the evolution over 10 different times with each time's collection according to a DPM model with 200 2-dimensional Gaussian distribution points. 10 Gaussian points in $\mathbf{N}(\mathbf{0}, \mathbf{2I})$ are set as the 10 global clusters' mean parameters. Then 200 Gaussian points within a cluster are sampled with this cluster's mean parameter and deviation parameter sampling from $\mathbf{N}(\mathbf{0}, \mathbf{0.2I})$, where $\mathbf{I}$ is identify matrix. After the generation of such a dataset, we obtain the number of the clusters and the cluster assignments as the ground truth. We intentionally generate different numbers of the clusters at different times, as shown in Figure 5.

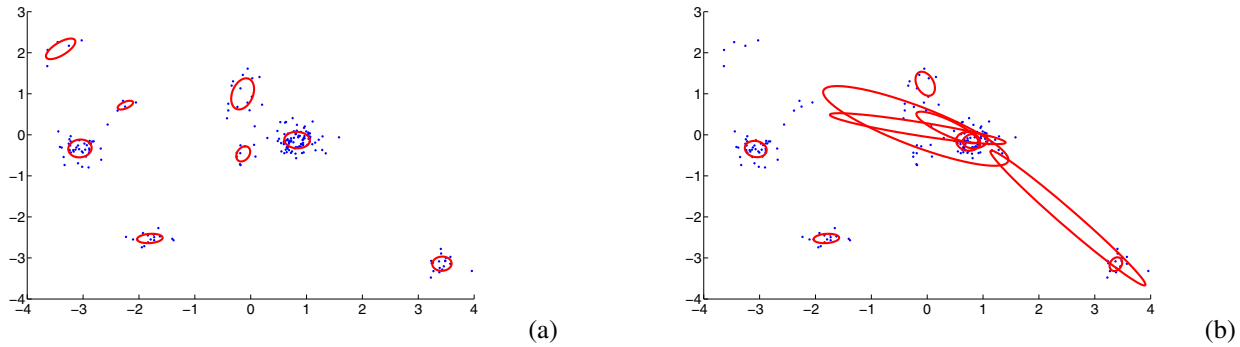
In the inference process, we tune the hyperparameters as follows. In each iteration, we use the vague Gamma priors [12] to update α, λ, and γ from $\Gamma(1, 1)$. Figure 3 shows an example of the clustering results between HDP-HTM and PCQ at time 8 for the synthetic data. Clearly, HDP-HTM has much better performance than PCQ in this synthetic data.

For a more systematic evaluation on this synthetic dataset, we use NMI (Normalized Mutual Information) [24] to quantitatively compare the clustering performances among all the four algorithms (HDP-HTM, HDP, PCM, and PCQ). NMI measures how much information two random distribution variables (computed clustering assignment and groundtruth clustering assignment) share, the larger the better with 1 as normalized maximum value. Figure 4 documents the performance comparison. From this figure, the average NMI values across the 10 times for HDP-HTM and HDP are 0.86 and 0.78, respectively, while those for PCQ and PCM are 0.70 and 0.71, respectively. HDP works worse than HDP-HTM for the synthetic data. The reason is that HDP model is unable to capture the cluster transition correspondence during the evolution among the data collections across the time in this case while HDP-HTM is able to explicitly solve for this correspondence problem; on the other hand, HDP still performs better than PCQ and PCM as HDP is able to learn the cluster number automatically during the evolution.

Since one of the advantages of the HDP-HTM model is to be able to learn the number of the clusters and the clustering structures during the evolution, we report this performance for the HDP-HTM compared with HDP on this synthetic dataset in Figure 5. Here, we define the expected number of the clusters at each time as the average number of the clusters in all the posterior sampling iterations after the burn-in period. Thus, these numbers are not necessarily integers. Clearly, both models are able to learn the cluster numbers, with HDP-HTM having a better performance than HDP. Since both PCQ and PCM do not have this capability, they are not included in this evaluation.

5.2 Real Dataset

In order to showcase the performance of HDP-HTM model on real data applications, we apply HDP-HTM



(a) (b)

Figure 3. Illustrated Clustering results of HDP-HTM (a) and PCQ (b) for the synthetic data

to a subset of the 20 Newsgroups data [2]. We intentionally set the number of the clusters at each time as the same number to accommodate the comparing algorithms PCQ and PCM which have this assumption of the same cluster number over the evolution. Also we select 10 clusters (i.e., topics) from the dataset (alt.atheism, comp.graphics, rec.autos, rec.sport.baseball, sci.crypt, sci.electronics, sci.med, sci.space, soc.religion.christian, talk.politics.mideast), with each having 100 documents. To "simulate" the corresponding 5 different times, we then split the dataset into 5 different collections, each of which has 20 documents randomly selected from each clusters. Thus, each collection at a time has 10 topics to generate words. We have preprocessed all the documents with the standard text processing for removing the stop words and stemming the remaining words.

To apply the HDP-HTM and HDP models, a symmetric Dirichlet distribution is used with the parameter 0.5 for the prior base distribution H. In each iteration, we update α, γ, and λ in HDP-HTM, from the gamma priors $\Gamma(0.1, 0.1)$. For LDA, α is set 0.1 and the prior distribution of the topics on the words is a symmetric Dirichlet distribution with concentration parameter 1. Since LDA only works for one data collection and requires a known cluster number in advance, we explicitly apply LDA to the data collection with the ground truth cluster number as input at each time.

Figure 6 reports the overall performance comparison among all the five methods using NMI metric again. Clearly HDP-HTM outperforms PCQ, PCM, HDP, and LDA at all the times; in particular, the difference is substantial for PCQ and PCM. Figure 7 further reports the performance on learning the cluster numbers at different times for HDP-HTM compared with HDP. Both models have a reasonble performance in automatically learning the cluster number at each time in comparison with the ground truth, with HDP-HTM having a clearly better performance than HDP in average.

[2] http:kdd.ics.uci.edu/databases/20newsgroups/

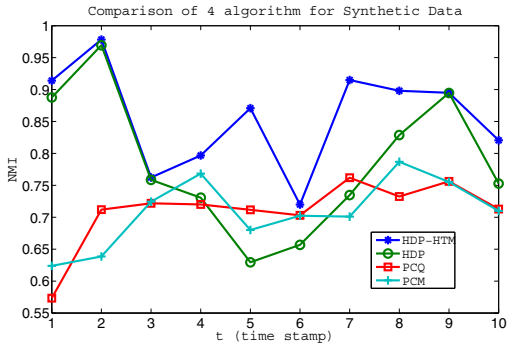


Figure 4. The NMI performance comparison of the four algorithms on the synthetic dataset

In order to truly demonstrate the performance of HDP-HTM in comparison with the state-of-the-art literature on a real evolutionary clustering scenario, we have manually collected Google News articles for a continuous period of five days with both the data items (i.e., words in the articles) and the clusters (i.e., the news topics) evolving over the time. The evolutionary ground truth for this dataset is as follows. For each of the continuous five days, we have the number of the words, the number of the clusters, the number of the documents as (6113, 5, 50), (6356, 6, 60), (7063, 5, 50), (7762, 6, 60), and (8035, 6, 60), respectively. In order to accommodate the assumption of PCM and PCQ that the cluster number stays the same during the evolution, but at the same time in order to demonstrate the capability of HDP-HTM to automatically learn the cluster number at each evolutionary time, we intentionally set the news topic number (i.e., the cluster number) at each day's collection to have a small variation deviation during the evolution. Again, in order to compare the text clustering capability of LDA [7, 17] with a known topic number in advance, we use the ground truth cluster number at each time as the input to

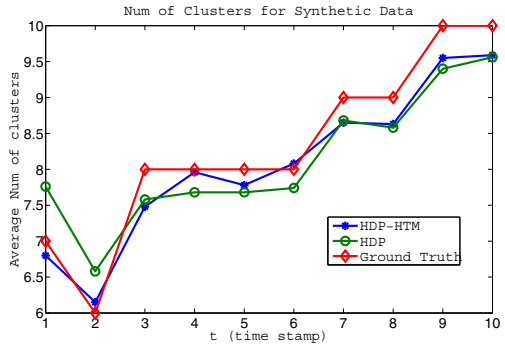


Figure 5. The cluster number learning performance of the HDP-HTM in comparison with HDP on the synthetic dataset

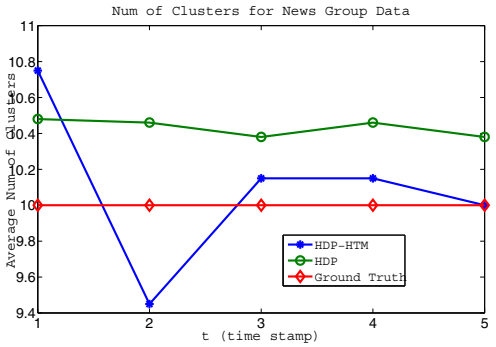


Figure 7. Cluster number learning performance of HDP-HTM in comparison with HDP on the 20 Newsgroups dataset

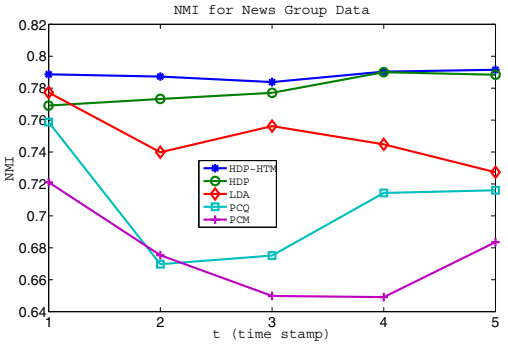


Figure 6. The NMI performance comparison among the five algorithms on the 20 Newsgroups dataset

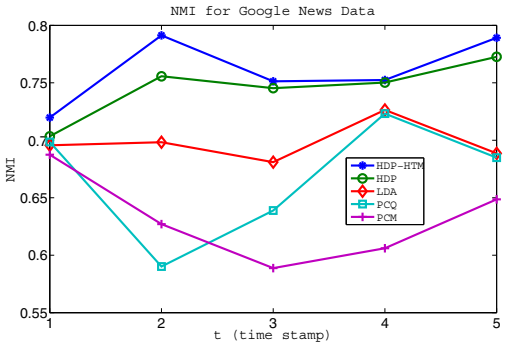


Figure 8. The NMI performance comparison for all the five algorithms on the Google News dataset

LDA. The parameter tuning process is similar to that in the experiment using the 20 Newsgroup dataset.

Figure 8 reports the NMI based performance evaluations among the five algorithms. Again, HDP-HTM outperforms PCQ, PCM, HDP, and LDA at all the times, especially substantially better than PCQ, PCM, and LDA. PCQ and PCM fail completely in most of the cases as they assume that the number of the clusters remains the same during the evolution, which is not true in this scenario.

Figure 9 further reports the performance on learning the cluster numbers for different times for HDP-HTM compared with HDP model. In this dataset, HDP-HTM has a much better performance than HDP to learn the cluster numbers automatically at all the times.

6 Conclusions

In this paper, we have addressed the evolutionary clustering problem. Based on the recent literature on DP based models and HMM, we have developed the HDP-HTM model as an effective solution to this problem. HDP-HTM model substantially advances the evolutionary clustering literature in the sense that it not only performs better than the existing literature, but more importantly it is able to automatically learn the dynamic cluster numbers and the dynamic clustering structures during the evolution, which is a common scenario in many real evolutionary clustering applications. In addition, HDP-HTM also explicitly addresses the correspondence issue whereas all the existing solutions fail to do so. Extensive evaluations demonstrate the effectiveness and the promise of HDP-HTM in comparison with the state-of-the-art literature.

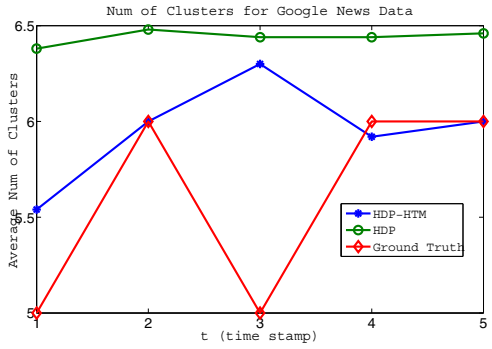


Figure 9. The cluster number learning performance of HDP-HTM in comparison with HDP on the Google News dataset

7 Acknowledgement

This work is supported in part by NSF (IIS-0535162 and IIS-0812114). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

[1] D. Aldous. Exchangeability and related topics. *Ecole de Probabilites de Saint-Flour*, (XIII):1–198, 1983.

[2] C. Antoniak. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

[3] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chins. *The Annals of Mathematical Statistics*, 39(1):164–171, 1970.

[4] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen. The infinite hidden markov model. In *NIPS 14*, 2002.

[5] D. Blackwell and J. MacQueen. Ferguson distributions via plya urn schemes. *The Annals of Statistics*, 1(2):353–355, 1973.

[6] D. Blei and J. Lafferty. Dynamic topic models. In *In Proceedings of the 23rd International Conference on Machine Learning*, 2006.

[7] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, January 2003.

[8] G. Casella and E. I. George. Explaining the gibbs sampler. *The American Statistician*, 46(3):167–174, Aug. 1992.

[9] D. Chakrabarti, R. Kumar, and A. Tomkins. Evolutionary clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 554–560, 2006.

[10] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 153–162, 2007.

[11] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[12] M. D. Escobar and M. West. Bayesian density estimation and inference using mixtures. *The Annals of Statistics*, 90:577–588, 1995.

[13] T. S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973.

[14] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. Developing a tempered hdp-hmm for systems with state persistence. *Technical Report*, Nov. 2007.

[15] J. V. Gael, Y. Saatci, Y. W. Teh, and Z. Ghahramani. Beam sampling for the infinite hidden markov model. In *25th International Conference on Machine Learning*, 2008.

[16] T. L. Griffiths and M. Steyvers. Finding scientific topics. In *Proceedings of the National Academy of Sciences*, pages 5228–5235. Feb., 2004.

[17] G. Heinrich. Parameter estimation for text analysis. *Technical Report*, 2004.

[18] H. Ishwaran and L. F. JAMES. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–, 2001.

[19] D. J. MacKay. Ensemble learning for hidden markov models. *Technical Report*, 1997.

[20] R. M. Neal. Probabilistic inference using markov chain monte carlo methods. *Technical Report*, (CRG-TR-93-1), September 1993.

[21] K. Ni, L. Carin, and D. Dunson. Multi-task learning for sequential data via ihmms and the nested dirichlet process. In *ICML*, pages 689–696, 2007.

[22] L. Rabiner. A tutorial on hidden markov models and selected applications inspeech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[23] J. Sethuraman. A constructive definition of dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

[24] A. Strehl and J. Ghosh. Cluster ensembles a knowledge reuse framework for combining partitionings. In *Proceedings of AAAI*, 2002.

[25] L. Tang, H. Liu, J. Zhang, and Z. Nazeri. Community evolution in dynamic multi-mode networks. In *KDD*, pages 677–685, 2008.

[26] Y. Teh, M. B. M. Jordan, and D. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2007.

[27] X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433, 2006.

[28] T. Xu, Z. Zhang, P. Yu, and B. Long. Dirichlet process based evolutionary clustering. In *ICDM*, 2008.