

into the model in the multiple corpora setting is nontrivial. The last challenge is how to determine the cluster numbers. In time varying text data, a cluster may emerge and disappear. Consequently, the cluster numbers may change through time. It is awkward to require users to specify a cluster number at each time epoch for each corpus. Therefore a mechanism is preferred to automatically determine all the numbers of clusters.

The traditional clustering approaches deal with a single static data corpus. Hence the direct use of a general global clustering model on all data may fail to represent the diversity both over time and across different corpora. Beyond the classical clustering approaches, the most recent efforts only focus on tackling two sub problems. One is *learning from multiple correlated text corpora*, which aims to discover the related content across different text corpora as well as the distinctive information in each corpus [22, 26, 17]. Another is *learning from a time-varying data corpus*, which aims to discover the evolving patterns in the corpus as well as the snapshot clusters at each time epoch [3, 8, 9, 1, 16]. Both types of approaches are not sufficient to tackle the above challenges.

To deal with above challenges, in this paper, we propose an *evolutionary hierarchical Dirichlet process* (EvoHDP) model, which extends the hierarchical Dirichlet process (HDP) [22] to a time evolving scenario. In EvoHDP, each HDP is built for multiple corpora at each time epoch, and the time dependencies are incorporated into adjacent epochs under the Markovian assumption. Specifically, the dependency is formulated by mixing two distinct Dirichlet processes (DPs). One is the DP model for the previous epoch, and the other is an updating DP model. To infer the EvoHDP model, we also propose a *cascaded Gibbs sampling* scheme. The proposed EvoHDP model can effectively discover cluster evolution patterns over time and across corpora. Moreover, the cluster numbers can automatically be determined due to the infinity property of DP.

2. RELATED WORK

In this section, we briefly introduce three categories of work related to this paper, including learning from multiple correlated data corpora, learning evolution patterns from a time-varying corpus, and some initial efforts involving multiple dynamic data.

In the research of learning from multiple correlated data corpora, HDP is a milestone [22]. It extended DP [2, 23] to model multiple correlated data corpora. In HDP, each data corpus is modeled by an infinite DP mixture model, and the infinite set of mixing clusters is shared among all corpora. Later works [16, 6] relax the assumption of HDP and incorporate more correlations between different corpora. Besides HDP, other efforts [4, 19, 26, 17] on this research topic are devoted to the extensions of the Latent Dirichlet Allocation (LDA) topic model [5]. However, none of them consider the problem of automatically determining the cluster/topic numbers. In fact, as pointed out in [22], HDP can be used as an LDA-based topic model, where the number of clusters can be automatically inferred from data. Therefore, HDP is more practical when users have little knowledge about the content to be analyzed.

In the research of learning evolutionary clusters from a time-varying corpus, *evolutionary clustering* [8, 9, 32, 1, 30, 31] is a new research topic. Evolutionary clustering aims to preserve the smoothness of clustering results over time, while fitting the data of each epoch. Among the above works, the approaches in [1, 30, 31] utilized DP to automatically determine the cluster numbers. In fact, incorporating time dependencies into DP mixture models is a hot topic in the research of Bayesian nonparametric [20, 10, 7, 16, 15, 14]. Moreover, some works have focused on extending LDA to dynamic topic models [3, 27, 25]. We noted that even though the title

of [16] is similar to this paper, it actually presents an evolutionary DP mixture model for a single dynamic corpus.

To the best of our knowledge, there are four works that seemingly involve multiple time-varying data but actually handle different problems [28, 29, 11, 21]. Wang et al. [28] focused on detecting the simultaneous *busting* of some topics in multiple text streams. They did not concentrate on the evolving patterns but only the busting behavior of topics. Wang et al. [29] extended [28] to extract common topics from multiple text streams. They regarded that the underlying models of all streams are the same except that time delays exist between different streams. Hence they adjusted the timestamps of all documents to synchronize multiple streams and then learned a common topic model. Their assumption pays more attention to aligning topics of different streams and thus degenerates the topic diversity among different streams. Leskovec et al. [11] focused on tracking the spreading behaviors of short phrases (or “*memes*”) across the web to represent news cycles. Although memes act as signatures of clusters/topics, they are not enough to represent clusters/topics as the context information is lost. Tang et al. [21] worked on the dynamic multi-mode network with several types of actors. They aimed to provide a partition for each type of actors at each time epoch. There are two typical features in their problem: (1) the data to be handled is relational data, i.e., linkages between actors are required; (2) the partitions over time should be on a same set of actors while relationships among them vary over time. None of the above four works attempted to discover the cluster evolution over time and across corpora. On the contrary, this is exactly the focus of this paper. In addition, none of them handled the problem of automatically determining the cluster/topic numbers, which is one of the major considerations of our work.

3. PRELIMINARIES

In this section, we briefly introduce DP and HDP.

A DP can be considered as a distribution of a random probability measure¹ G , and we write $G \sim \text{DP}(\alpha_0, G_0)$, where α_0 is a positive *concentration* parameter, and G_0 is a *base measure*. Sethuraman [18] showed that a measure G drawn from a DP is discrete, by the following *stick-breaking construction*:

$$\{\phi_k\}_{k=1}^{\infty} \stackrel{i.i.d.}{\sim} G_0, \boldsymbol{\pi} \sim \text{GEM}(\alpha_0), G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}. \quad (1)$$

The discrete set of atoms $\{\phi_k\}_{k=1}^{\infty}$ are drawn from the base measure, and $\text{GEM}(\alpha_0)$ refers to such a process: $\tilde{\pi}_k \sim \text{Beta}(1, \alpha_0)$, $\pi_k = \tilde{\pi}_k \prod_{i=1}^{k-1} (1 - \tilde{\pi}_i)$. We use boldface $\boldsymbol{\pi} = (\pi_k)_{k=1}^{\infty}$ to represent the vector, which will be followed in the rest of the paper. δ_{ϕ_k} is a probability measure concentrated at ϕ_k . After observing draws $\theta_1, \dots, \theta_{n-1}$ from G , the posterior of G is still a DP

$$G|\theta_1, \dots, \theta_{n-1} \sim \text{DP}\left(\alpha_0 + n - 1, \frac{m_k \delta_{\phi_k} + \alpha_0 G_0}{\alpha_0 + n - 1}\right), \quad (2)$$

where m_k is the number of draws in $\{\theta_i\}_{i=1}^{n-1}$ taking the same value ϕ_k . This posterior preserves the possibility of drawing a new distinct value from G_0 but puts more concentration on observed values.

HDP uses multiple DPs to model multiple correlated corpora. In HDP, a global measure G_0 is drawn from a DP (γ, H) , with concentration parameter γ and base measure H . Then, a set of measures $\{G_j\}_j$ is drawn from a DP with base measure G_0 . Each G_j models the corpus j . Such a process is summarized as

$$G_0 \sim \text{DP}(\gamma, H), \quad G_j|G_0, \alpha_0 \stackrel{i.i.d.}{\sim} \text{DP}(\alpha_0, G_0). \quad (3)$$

Given the global measure G_0 , G_j 's are conditionally dependent. Having G_j, n_j data samples $\{x_{ji}\}_{i=1}^{n_j}$ in each corpus j are drawn from

¹In general, we can regard the measure as a distribution.

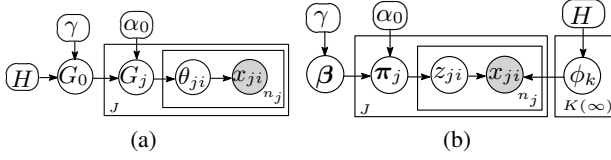


Figure 2: Graphical representation for HDP: circles denote random variables, oval nodes denote parameters, shaded nodes denote observed variables, and plates indicate replication. (a) HDP. (b) The stick-breaking construction of HDP.

the following mixture model

$$\{\theta_{ji}\}_i \stackrel{i.i.d.}{\sim} G_j, \quad x_{ji} \sim F(x|\theta_{ji}), \quad (4)$$

where $F(x|\theta_{ji})$ is a distribution parameterized by θ_{ji} to generate x_{ji} , e.g., that of an exponential family. Eqs. (3) and (4) together define the *hierarchical Dirichlet process mixture model*². The graphical representation for an HDP is described in Fig. 2(a).

Moreover, according to Eq. (1), G_0 has the form $G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$, where $\phi_k \stackrel{i.i.d.}{\sim} H$, $\beta \sim \text{GEM}(\gamma)$. Then it is shown in [22] that G_j can be constructed as

$$G_j = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\phi_k}, \quad \pi_j | \beta, \alpha_0 \sim \text{DP}(\alpha_0, \beta), \quad (5)$$

which means that different corpora share the same set of distinct atoms [22]. This is the stick-breaking construction of HDP, and the corresponding graphical model is shown in Fig. 2(b).

4. EVOLUTIONARY HDP

In this section, we begin with the introduction of the EvoHDP model, and then show how to infer the model using a proposed Gibbs sampling technique.

We first introduce the data settings and some notations which are useful for subsequent discussions. There are J corpora varying over T time epochs. Considering the possibility that no data observed for some corpora at an epoch, we denote the number of corpora at epoch t is J^t . At each epoch t , there are n_j^t data samples in corpus j , and we denote a data sample (e.g., a document³) as x_{ji}^t . We assume the underlying model to generate x_{ji}^t for corpus j at epoch t is an infinite mixture model

$$p_j^t(x|G_j^t) = \int G_j^t(\theta) f(x|\theta) d\theta,$$

where $G_j^t = \sum_{k=1}^{\infty} \pi_{jk}^t \delta_{\phi_k}$ and f is the density of a distribution $F(x|\theta)$. We call the density parameterized by a distinct atom ϕ_k as a mixing component, which describes a cluster.⁴

4.1 Model

We model the multiple correlated time-varying corpora as a series of HDPs with time dependencies, as shown in the graphical presentation of Fig. 3(a). Specifically, at each time epoch t , we use an HDP to model the multiple correlated corpora at that epoch and then put time dependencies between adjacent epochs based on the Markovian assumption. To build an overall bookkeeping of components for all epochs, we let these HDPs share an identical discrete base measure G , and G is drawn from $\text{DP}(\xi, H)$ with H as the base measure. We call G the *overall measure*. Moreover, for an HDP to

²We just call ‘‘hierarchical Dirichlet process mixture model’’ as HDP for short in this paper.

³Generally a data sample can be quite different. For example, when using HDP to formulate LDA topic model, a data sample is a word.

⁴When x represents a word, and $F(x|\theta)$ is a multinomial distribution over a finite word vocabulary, the distribution F is often regarded as a ‘‘topic’’ [5].

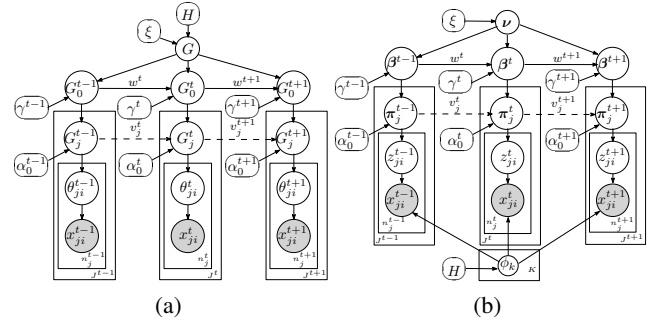


Figure 3: The graphical representation for the EvoHDP model. (a) The original representation. (b) The stick-breaking construction.

model the corpora at epoch t , we use G_0^t to denote the global measure at that epoch, and call it the *snapshot global measure*. Then, the local measure G_j^t for the j -th corpus at time epoch t is called the *snapshot local measure*. In this way, an EvoHDP has one more layer than the original HDP [22].

The key issue of EvoHDP is how to incorporate time dependencies between adjacent epochs. We introduce two types of dependencies into different layers in EvoHDP to model the different evolving manner. In the following we will interpret this model step-by-step to explain why we use this scheme and how it would be useful.

The first is the dependency of snapshot global measure G_0^t on G_0^{t-1} . Since G_0^t is the measure for the components in all corpora at time t , the difference between G_0^t and G_0^{t-1} reflects the evolving of the global components in all corpora. We call this *global time dependency*.

The second is the time dependency within a corpus, i.e., the dependency of the snapshot local measure G_j^t on G_j^{t-1} . Since G_j^t is the measure for the components within corpus j at time epoch t , the difference between G_j^t and G_j^{t-1} reflects the evolving of components within the corpus j . Then we call this *intra-corporus time dependency*.

In Fig. 3(a), we use dashed lines to represent the second type of dependencies, since in some cases (e.g. in HDP based LDA), there are no intra-corporus dependencies.

The generation process of EvoHDP is as follows.

1. Draw an overall measure $G \sim \text{DP}(\xi, H)$. G plays a role of the overall component bookkeeping for all corpora at all epochs.

2. For each epoch t :

2.1. Draw the snapshot global measure G_0^t according to the overall measure G and the previous snapshot global measure G_0^{t-1} :

$$G_0^t \sim \text{DP}(\gamma^t, w^t G_0^{t-1} + (1 - w^t)G). \quad (6)$$

2.2. Draw the snapshot local measures $\{G_j^t\}_{j=1}^{J^t}$. Each G_j^t for corpus j at epoch t is drawn according to the snapshot global measure G_0^t and the previous snapshot local measure G_j^{t-1} :

$$G_j^t \sim \text{DP}(\alpha_0^t, v_j^t G_j^{t-1} + (1 - v_j^t)G_0^t). \quad (7)$$

2.3. For data samples $\{\{x_{ji}^t\}_{i=1}^{n_j^t}\}_{j=1}^{J^t}$, draw the parameters of the component densities and generate the data samples:

$$\theta_{ji}^t \stackrel{i.i.d.}{\sim} G_j^t, \quad x_{ji}^t \sim F(x|\theta_{ji}^t),$$

where $F(x|\theta_{ji}^t)$ is a distribution parameterized by θ_{ji}^t , e.g., that from an exponential family.

Compared to the original HDP model, two levels of time dependencies are incorporated in by Eq. (6) and Eq. (7). When we set all w^t and v_j^t to zero, the EvoHDP is a three-layer HDP.

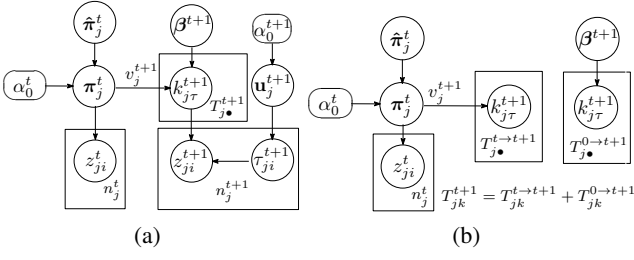


Figure 4: The generation mechanism of G_j^{t+1} . (a) A part insight into the generation process of restaurant j at epoch $t + 1$. (b) The generation of tables of restaurant j at epoch $t + 1$.

Following the convention of DP related models [22, 24, 16], we will also provide other two perspectives and a restaurant-metaphor for the EvoHDP. They help us better understand the model and lay the foundations of the inference scheme introduced in Sec. 4.4.

4.2 The Stick-Breaking Construction

According to the stick-breaking construction (Eq. (1)) of DP, we can write the explicit form of G :

$$G = \sum_{k=1}^{\infty} v_k \delta_{\phi_k}, \quad v \sim \text{GEM}(\xi).$$

Consequently, according to Eqs. (6) and (5), G_0 has the form

$$G_0^t = \sum_{k=1}^{\infty} \beta_k^t \delta_{\phi_k}, \quad \beta^t \sim \text{DP}(\gamma^t, \hat{\beta}^t), \quad (8)$$

where $\hat{\beta}^t = w^t \beta^{t-1} + (1 - w^t) v$.

Similarly, we can also write the form of G_j^t as

$$G_j^t = \sum_{k=1}^{\infty} \pi_{jk}^t \delta_{\phi_k}, \quad \pi_j^t \sim \text{DP}(\alpha_0^t, \hat{\pi}_j^t), \quad (9)$$

where $\hat{\pi}_j^t = v_j^t \pi_j^{t-1} + (1 - v_j^t) \beta^t$.

In this way, we obtain the stick-breaking construction for EvoHDP, which is shown in Fig. 3(b), where z_{ji}^t is the index of the component emitting x_{ji}^t .

According to this perspective, the EvoHDP provides a prior in which the snapshot models $\{G_0^t\}$, $\{G_j^t\}_{t,j}$ of all corpora at all epochs share the same infinite set of mixing components $\{\phi_k\}_{k=1}^{\infty}$. The differences among these snapshot models lie in the mixing weights.

4.3 Hierarchical Infinite Mixture Model and A Restaurant Franchise Metaphor

Based on the stick-breaking construction for a DP, if we continue to use the stick-breaking constructions to represent π_j^t and β^t drawn from the two DPs in Eqs. (8) and (9), we obtain the hierarchical infinite mixture model of EvoHDP. This perspective clearly interprets the generation mechanism of EvoHDP.

We begin with the metaphor following the Chinese restaurant franchise (CRF) for HDP [22]. A corpus j is called a *restaurant*, and a global atom k is called a *dish*. We use a *day* to refer to a time epoch. We focus on the generation of component indicator z_{ji}^t . Having z_{ji}^t , the left generation process of x_{ji}^t is straightforward, which is drawn from $F(x|\phi_{z_{ji}^t})$.

4.3.1 Generation of Snapshot Local Measure G_j^t

We first show the generation mechanism of each snapshot local measure $G_j^{t+1} = \sum_{k=1}^{\infty} \pi_{jk}^{t+1} \delta_{\phi_k}$, i.e., the behavior of restaurant j in day $t + 1$. Since π_j^{t+1} is drawn from $\text{DP}(\alpha_0^{t+1}, \hat{\pi}_j^{t+1})$ as shown in Eq. (9), we can represent this DP using the stick-breaking construction as

$$\pi_j^{t+1} = \sum_{\tau=1}^{\infty} u_{j\tau}^{t+1} \delta_{k_{j\tau}^{t+1}}, \quad \{k_{j\tau}^{t+1}\}_{\tau} \stackrel{i.i.d.}{\sim} \hat{\pi}_j^{t+1}, \quad u_j^{t+1} \sim \text{GEM}(\alpha_0^{t+1}), \quad (10)$$

which is illustrated in Fig. 4(a). We call τ a *table*, then u_j^{t+1} is a

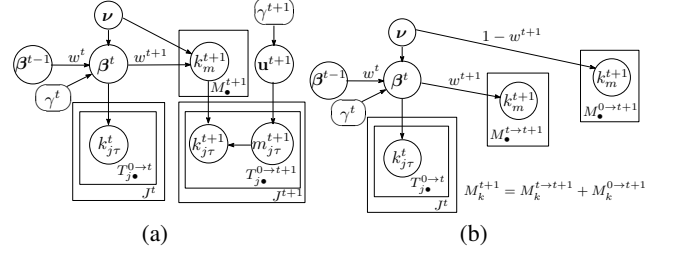


Figure 5: The generation mechanism of G_0^{t+1} . (a) A part insight into the generation process of the snapshot global measure. (b) The generation of meta-tables of the franchise at epoch $t + 1$.

distribution on tables. We can explain Eq. (10) as a peculiar way of dish serving in a restaurant. First, waiters place infinite number of tables. On each table τ , a dish $k_{j\tau}^{t+1}$ is selected from the *local table-dish menu* $\hat{\pi}_j^{t+1}$ by waiters. Then, when a customer i enters in restaurant j , he selects a table τ_{ji}^{t+1} from the *local customer-table menu* u_j^{t+1} (with probability $u_{j\tau_{ji}^{t+1}}^{t+1}$) and enjoys the dish $k_{j\tau_{ji}^{t+1}}^{t+1}$ on the table. Consequently, the component indicator $z_{ji}^{t+1} = k_{j\tau_{ji}^{t+1}}^{t+1}$. Moreover, $\hat{\pi}_j^{t+1}$ plays the role of the *local customer-dish menu*, which indicates the customers' preference on dishes in day $t + 1$.

We then explain how a dish is placed on a table by the waiter. From Eq. (10), we see that $\{k_{j\tau}^{t+1}\}_{\tau} \stackrel{i.i.d.}{\sim} \hat{\pi}_j^{t+1}$, i.e., a dish is drawn from the local table-dish menu $\hat{\pi}_j^{t+1}$. Notice that $\hat{\pi}_j^{t+1} = v_j^{t+1} \pi_j^t + (1 - v_j^{t+1}) \beta^{t+1}$ is a mixture of two menus, where π_j^t is the local customer-dish menu in this restaurant yesterday and β^{t+1} is the *global franchise menu* of current day recommended by the franchise manager. Then for a table τ , a waiter select a dish $k_{j\tau}^{t+1}$ from yesterday's local customer-dish menu π_j^t with probability v_j^{t+1} while from current day's global franchise menu β^{t+1} with probability $1 - v_j^{t+1}$. This means in the franchise, a restaurant designs its localized menu by considering both yesterday's local taste and current day's franchise menu.

Obviously, it is possible that multiple tables have a same dish. In restaurant j in day $t + 1$, we denote the number of tables with dish k as T_{jk}^{t+1} . Then the total number of tables in restaurant j in day $t + 1$ is denoted as $T_j^{t+1} = \sum_k T_{jk}^{t+1}$. Then for a dish k , among the T_{jk}^{t+1} tables, there are $T_{jk}^{t \rightarrow t+1}$ tables whose dishes are selected from yesterday's local customer-dish menu π_j^t and $T_{jk}^{0 \rightarrow t+1}$ tables whose dishes are selected from current day's global franchise menu β^{t+1} . Then, $\forall k$, we have $T_{jk}^{t+1} = T_{jk}^{t \rightarrow t+1} + T_{jk}^{0 \rightarrow t+1}$. This mechanism is shown in Fig. 4(b).

4.3.2 Generation of Snapshot Global Measure G_0^t

Then we show the generation mechanism of the snapshot global measure $G_0^{t+1} = \sum_{k=1}^{\infty} \beta_k^{t+1} \delta_{\phi_k}$, i.e., how the franchise manager recommends the global franchise menu β^{t+1} to all restaurants. The procedure is explained in Fig. 5(a). Since β^{t+1} is drawn from $\text{DP}(\gamma^{t+1}, \hat{\beta}^{t+1})$ as shown in Eq. (8), β^{t+1} can also be represented using the stick-breaking construction as

$$\beta^{t+1} = \sum_{m=1}^{\infty} u_m^{t+1} \delta_{k_m^{t+1}}, \quad \{k_m^{t+1}\}_m \stackrel{i.i.d.}{\sim} \hat{\beta}^{t+1}, \quad u^{t+1} \sim \text{GEM}(\gamma^{t+1}). \quad (11)$$

We call each m a *metatable*, then u^{t+1} is a distribution on metatables. The manager has infinite number of empty metatables beforehand and he selects a dish k_m^{t+1} for each metatable m from the *metatable-dish menu* $\hat{\beta}^{t+1}$ of day $t + 1$. Remind that when a waiter in restaurant j place a table τ , with probability $1 - v_j^{t+1}$, he selects the dish $k_{j\tau}^{t+1}$ according to the global franchise menu β^{t+1} . Now he just

need to select a metatable $m_{j^t}^{t+1}$ according to the *waiter-metatable menu* \mathbf{u}^{t+1} , and then the dish $k_{m_{j^t}^{t+1}}$ on the metatable is the one he should select. Hence the global franchise menu $\boldsymbol{\beta}^{t+1}$ also plays the role of the *waiter-dish menu*, which indicates how a waiter selects dishes for tables.

The dishes on the metatables of the franchise in day $t + 1$ also come from two menus, i.e., the yesterday's global franchise menu $\boldsymbol{\beta}^t$ and an *overall menu* \mathbf{v} . The overall menu \mathbf{v} reflects common taste of the franchise. It is also possible that different metatables have a same dish and we denote the number of metatables with dish k as M_k^{t+1} . Among the M_k^{t+1} metatables, there are $M_k^{t \rightarrow t+1}$ metatables whose dishes are selected from yesterday's franchise menu, and $M_k^{0 \rightarrow t+1}$ metatables whose dishes are selected from the overall menu \mathbf{v} . Then $\forall k$, we have $M_k^{t \rightarrow t+1} = M_k^{t \rightarrow t+1} + M_k^{0 \rightarrow t+1}$.

4.4 A Cascaded Gibbs Sampler

The hierarchical infinite mixture model and the restaurant franchise metaphor actually define a Gibbs sampling scheme for EvoHDP. We can derive a cascaded Gibbs sampling procedure by sequentially sampling following variables.

4.4.1 Sampling \mathbf{v}

According to the generation of metatables introduced in Fig. 5(b), what are drawn from the overall measure $G = \sum_{k=1}^{\infty} \nu_k \delta_{\phi_k}$ are the dishes on the metatables of all days designed by the franchise manager. We denote the number of all these metatables with dish k as $\mathcal{M}_k = \sum_t M_k^{0 \rightarrow t}$, and the total number of metatables drawn from G as $\mathcal{M}_\bullet = \sum_k \mathcal{M}_k$. As $G \sim \text{DP}(\xi, H)$, assume we have known the count variables $\{\mathcal{M}_k\}_k$, according to the property Eq. (2), the posterior of G is also a DP:

$$G|\xi, H, \{\mathcal{M}_k\}_{k=1}^K \sim \text{DP}\left(\xi + \mathcal{M}_\bullet, \frac{H + \sum_{k=1}^K \mathcal{M}_k \delta_{\phi_k}}{\xi + \mathcal{M}_\bullet}\right),$$

where K is the number of distinct dishes on all metatables. According to Sec. 5.2 of [22], G can be represented as

$$G = \sum_{k=1}^K \nu_k \delta_{\phi_k} + \nu_u G_u, \quad G_u \sim \text{DP}(\xi, H) \quad (12)$$

$$\mathbf{v} = (\nu_1, \dots, \nu_K, \nu_u) \sim \text{Dirichlet}(\mathcal{M}_1, \dots, \mathcal{M}_K, \xi). \quad (13)$$

This *augmented* representation reformulates original infinite vector \mathbf{v} to an equivalent finite one with length $K + 1$. Then \mathbf{v} is sampled from the Dirichlet distribution of Eq. (13).

Notice that in the following, G_0^t and G_j^t are also represented using above augmented representation as

$$G_0^t = \sum_{k=1}^K \beta_k^t \delta_{\phi_k} + \beta_u^t G_u, \quad G_j^t = \sum_{k=1}^K \pi_{jk}^t \delta_{\phi_k} + \pi_{ju}^t G_u, \quad G_u \sim \text{DP}(\xi, H).$$

Then $\boldsymbol{\beta}^t$ and $\boldsymbol{\pi}^t$ are both represented as finite vectors

$$\boldsymbol{\beta}^t = (\beta_1^t, \dots, \beta_K^t, \beta_u^t), \quad \boldsymbol{\pi}^t = (\pi_{j1}^t, \dots, \pi_{jK}^t, \pi_{ju}^t).$$

4.4.2 Sampling $\boldsymbol{\beta}^t$ and $\boldsymbol{\pi}^t$

According to Fig. 5(b), what are drawn from $\boldsymbol{\beta}^t$ include two parts. One part is the $T_{jk}^{0 \rightarrow t} = \sum_{j,k} T_{jk}^{0 \rightarrow t}$ dishes on the $T_{jk}^{0 \rightarrow t}$ tables in all restaurants of day t . The second part is the $M_{\bullet}^{t \rightarrow t+1}$ dishes on the $M_{\bullet}^{t \rightarrow t+1}$ metatables of next day $t + 1$. We call all these tables and metatables drawn from $\boldsymbol{\beta}^t$ as *pseudo-tables*. We denote the number of pseudo-tables with the same dish k as \mathcal{T}_k^t , then $\mathcal{T}_k^t = T_{\bullet k}^{0 \rightarrow t} + M_k^{t \rightarrow t+1}$. As $\boldsymbol{\beta}^t \sim \text{DP}(\gamma^t, \boldsymbol{\beta}^t)$, and assuming we have obtained the count variables $\{\mathcal{T}_k^t\}_k$, the posterior of $\boldsymbol{\beta}^t$ is also a DP. Similar to Eq. (13), $\boldsymbol{\beta}^t$ can also be sampled from a Dirichlet distribution

$$(\beta_u^t, \beta_1^t, \dots, \beta_K^t) \sim \text{Dirichlet}(\tilde{\gamma}^t \cdot (\tilde{\beta}_u^t, \tilde{\beta}_1^t, \dots, \tilde{\beta}_K^t)), \quad (14)$$

where $\tilde{\gamma}^t = \gamma^t + \mathcal{T}_\bullet^t$, and

$$\tilde{\beta}_k^t = \frac{1}{\tilde{\gamma}^t} (\gamma^t w^t \beta_k^{t-1} + \gamma^t (1 - w^t) \nu_k + \mathcal{T}_k^t), \quad (15)$$

$$\tilde{\beta}_u^t = \frac{1}{\tilde{\gamma}^t} (\gamma^t w^t \beta_u^{t-1} + \gamma^t (1 - w^t) \nu_u). \quad (16)$$

The sampling of $\boldsymbol{\pi}^t$ is similar to that of $\boldsymbol{\beta}^t$:

$$(\pi_{ju}^t, \pi_{j1}^t, \dots, \pi_{jK}^t) \sim \text{Dir}(\tilde{\alpha}_0^t \cdot (\tilde{\pi}_{ju}^t, \tilde{\pi}_{j1}^t, \dots, \tilde{\pi}_{jK}^t)), \quad (17)$$

where $\tilde{\alpha}_0^t = \alpha_0^t + \mathcal{N}_{j\bullet}^t$, and

$$\tilde{\pi}_{jk}^t = \frac{1}{\tilde{\alpha}_0^t} (\alpha_0^t v^t \pi_{jk}^{t-1} + \alpha_0^t (1 - v^t) \beta_k^t + \mathcal{N}_{jk}^t), \quad (18)$$

$$\tilde{\pi}_{ju}^t = \frac{1}{\tilde{\alpha}_0^t} (\alpha_0^t v^t \pi_{ju}^{t-1} + \alpha_0^t (1 - v^t) \beta_u^t). \quad (19)$$

4.4.3 Sampling z_{ji}^t

Given $\boldsymbol{\pi}^t$, sampling z_{ji}^t is straightforward:

$$p(z_{ji}^t = k | x_{ji}^t, \dots) \propto p(z_{ji}^t = k | \boldsymbol{\pi}^t) p(x_{ji}^t | z_{ji}^t = k, \dots), \quad (20)$$

where $k \in \{1, \dots, K, u\}$. u refers to the index for the new component as introduced in Eq. (12). When $k = u$ is sampled, we add a new component into the component bookkeeping. In Eq. (20), the first item is a prior $p(z_{ji}^t = k | \boldsymbol{\pi}^t) = \pi_{jk}^t$ and the second item is a likelihood

$$p(x_{ji}^t | z_{ji}^t = k, \dots) = \int f(x_{ji}^t | \phi_k) p(\phi_k | X_{-t,ji}^k, H) d\phi_k, \quad (21)$$

where $X_{-t,ji}^k$ is the set of all the samples having been assigned to component k , other than x_{ji}^t .

4.4.4 Sampling T_{jk}^t and M_k^t

As described in Sec. 4.4.1 and Sec. 4.4.2, the posterior of \mathbf{v} , $\boldsymbol{\beta}^t$ and $\boldsymbol{\pi}^t$ depend on the count variables T_{jk}^t and M_k^t .

Remind that the count variable $T_{j\bullet}^t = \sum_k T_{jk}^t$ is the number of all tables in restaurant j at epoch t . These tables are occupied by the $\mathcal{N}_j^t = n_j^t + T_{j\bullet}^{t \rightarrow t+1}$ *pseudo-customers*. A pseudo-customer with dish k must have sat at a table with dish k . Hence $\forall k$, the $\mathcal{N}_{jk}^t = n_{jk}^t + T_{jk}^{t \rightarrow t+1}$ pseudo-customers cluster into T_{jk}^t tables. As shown in [22], given \mathcal{N}_{jk}^t , the table number T_{jk}^t can be sampled from a *Chinese Restaurant Process* (CRP)

$$T_{jk}^t | \beta_k^t, \pi_{jk}^t, \mathcal{N}_{jk}^t \sim \text{CRP}(\alpha_0^t v_j^t \pi_{jk}^{t-1} + \alpha_0^t (1 - v_j^t) \beta_k^t, \mathcal{N}_{jk}^t).$$

Notice that $\mathcal{N}_{jk}^t = n_{jk}^t + T_{jk}^{t \rightarrow t+1}$. As n_{jk}^t is the number of customers with dish k , it can be counted from the component assignments. To know \mathcal{N}_{jk}^t , we also need to know $T_{jk}^{t \rightarrow t+1}$, which depends on T_{jk}^{t+1} in turn. Thus the variable T_{jk}^t can be sampled from following recursive procedure:

$$(T_{jk}^{t \rightarrow t+1}, T_{jk}^{0 \rightarrow t+1}) \sim \text{Multinomial}(T_{jk}^{t+1}, [p, 1 - p]) \quad (22)$$

$$\mathcal{N}_{jk}^t = n_{jk}^t + T_{jk}^{t \rightarrow t+1} \quad (23)$$

$$T_{jk}^t | \beta_k^t, \pi_{jk}^t, \mathcal{N}_{jk}^t \sim \text{CRP}(\alpha_0^t v_j^t \pi_{jk}^{t-1} + \alpha_0^t (1 - v_j^t) \beta_k^t, \mathcal{N}_{jk}^t), \quad (24)$$

where $p = \frac{v_j^{t+1} \pi_{jk}^t}{(1 - v_j^{t+1}) \beta_k^{t+1} + v_j^{t+1} \pi_{jk}^t}$.

Similarly, we obtain the recursive sampling procedure for M_k^t :

$$(M_k^{t \rightarrow t+1}, M_k^{0 \rightarrow t+1}) \sim \text{Multinomial}(M_k^{t+1}, [q, 1 - q]) \quad (25)$$

$$\mathcal{T}_k^t = T_{\bullet k}^{0 \rightarrow t} + M_k^{t \rightarrow t+1} \quad (26)$$

$$M_k^t | \nu_k, \beta_k^t, \mathcal{T}_k^t \sim \text{CRP}(\gamma^t w^t \beta_k^{t-1} + \gamma^t (1 - w^t) \nu_k, \mathcal{T}_k^t), \quad (27)$$

where $q = \frac{w^{t+1} \beta_k^t}{(1 - w^{t+1}) \nu_k + w^{t+1} \beta_k^t}$.

4.4.5 Sampling Hyper-parameters

The concentration parameters of DPs, i.e., ξ , $\{\gamma^t\}$ and $\{\alpha_0^t\}$, can also be sampled by putting a vague gamma prior on them

$$\xi \sim Ga(a_\xi, b_\xi), \gamma^t \sim Ga(a_\gamma, b_\gamma), \alpha_0^t \sim Ga(a_\alpha, b_\alpha). \quad (28)$$

The sampling method is the same as that in [22]. Moreover, the time dependency parameters w^t and v_j^t can be taken as controlling parameters or also sampled using the method in [16] by putting a Beta prior for them and sampling from the posterior.

According to the sampling method for groups of variables described above, there are recursive dependencies along hierarchies and time epochs. We follow the dependencies of different sets of variables and design a cascaded Gibbs sample scheme. The procedure is summarized in Algorithm 1.

Algorithm 1 A cascaded Gibbs sampling scheme (one iteration)

```

1: for  $t = T$  to 1 do
2:   for  $j = 1$  to  $J^t$  do
3:     Sampling  $Z_j^t$  according to Sec. 4.4.3.
4:      $\forall k = 1, \dots, K$ , sampling count variables  $T_{jk}^{t \rightarrow t+1}$ ,  $T_{jk}^{0 \rightarrow t+1}$ , and  $T_{jk}^t$ 
       according to Eq. (22-24).
5:   end for
6:    $\forall k = 1, \dots, K$ , sampling count variables  $M_k^{t \rightarrow t+1}$ ,  $M_k^{0 \rightarrow t+1}$ , and  $M_k^t$ 
       according to Eq. (25-27).
7: end for
8: Sampling concentration parameters  $\xi, \gamma^t$  and  $\alpha_0^t$ .
9: Sampling  $\nu$  according to Sec. 4.4.1.
10: for  $t = 1$  to  $T$  do
11:   Sampling  $\beta^t$  according to Sec. 4.4.2.
12:   for  $j = 1$  to  $J^t$  do
13:     Sampling  $\pi_j^t$  according to Sec. 4.4.2.
14:   end for
15: end for

```

4.5 Global and Local Components

We not only need the component assignments Z , but also the component parameters $\{\phi_k\}$. As introduced in Eq. (21), $\{\phi_k\}$ have been integrated out in the sampling process. Having assignments Z , we can obtain the posterior of a ϕ_k :

$$p(\phi_k | \{x_{ji}^t | z_{ji}^t = k, \forall t, j, i\}, H) \propto p(\phi_k | H) p(\{x_{ji}^t | z_{ji}^t = k\} | \phi_k).$$

This distribution is a ‘‘global’’ one conditioned on data of all corpora from all epochs. In textual data, it denotes a global component k in the entire textual collection. If we limite the data in a corpus j at epoch t , we also obtain the posterior of ϕ_k as a local component

$$p(\phi_k | \{x_{ji}^t | z_{ji}^t = k, \forall i\}, H) \propto p(\phi_k | H) p(\{x_{ji}^t | z_{ji}^t = k, \forall i\} | \phi_k).$$

This is the component k as a local one in corpus j at epoch t .

5. EXPERIMENTS ON SYNTHETIC DATA

In this section, we use a synthetic data set from *mixtures of multinomial distributions* to test our approach.

5.1 Data

The data set consists of three time-evolving corpora covering 5 time epochs. Hence $\forall t, J^t$, the number of corpora at epoch t , is 3. There are totally $K = 8$ components (dishes) involved in all corpora. Each dish k is a 2-dimensional multinomial distribution $F(x|\phi_k) = \text{Multinomial}(x; \phi_k)$, with density

$$f(x|\phi_k) = \frac{\sum_{d=1}^D x_d!}{\prod_{d=1}^D x_d!} \prod_{d=1}^D \phi_{k,d}^{x_d},$$

where x is a D -dimensional nonnegative integer vector, and ϕ_k is a D -dimensional nonnegative real vector with $\sum_d \phi_{k,d} = 1$. Here

$D = 2$ and we set $\sum_d x_d = 200$. All the ϕ_k 's are listed in Tab. 1. Each corpus j at epoch t is a uniform mixture of 3 tables, and each table τ is associated with a dish indexed by $k_{j\tau}^t$. We denote the dish associated with table τ as $k_{j\tau}^t$. Hence this local model can be represented by

$$p_j^t(x) = \sum_{\tau=1}^3 \frac{1}{3} \text{Multinomial}(x; \phi_{k_{j\tau}^t}^t).$$

Then n_j^t samples are drawn from this mixture model, which compose the corpus j at epoch t . More details of the data set are shown in Tab. 1. In the conjunction area of ‘‘Tables ($k_{j_1}^t, k_{j_2}^t, k_{j_3}^t$)’’, the triple at row t and column j are the three dish indices to compose the local mixture model. In such a data set, different corpora overlap on some components, and the underlying models evolve over time.

5.2 Evaluation Criteria

We introduce several numerical criteria to evaluate two types of performances. The first type is the static performance on fitting training data and predicting held-out data. The second type is the temporal performance on preserving correlation between epochs, including the correlation within a corpus and that across different corpora.

5.2.1 Static Criteria

Two criteria are used to evaluate the static performance, the *normalized mutual information* (NMI) and *perplexity*.

NMI measures coherence between the clustering assignments and the true category labels. A higher value on NMI indicates a better clustering result. For each corpus j at each epoch t , having component assignments for all data samples, we can compute the value NMI_j^t for the corpus. Then we use average value $\sum_{t,j} NMI_j^t / \sum_{t,j} 1$ as the final result on the criterion of NMI.

Perplexity is a standard metric in information retrieval. We denote the training set as X_{train} and the held-out test set as X_{test} , then the per-sample perplexity of a model is defined based on the likelihood of ‘‘generating the test set given the training set’’:

$$\text{Perplexity} = \exp\left(-\frac{1}{n_{test}} \sum_{t,j,i} \log p(x_{ji, test}^t | \text{Model}, X_{train})\right),$$

where $x_{ji, test}^t$ the i -th data sample in corpus j at time epoch t and n_{test} is the size of test data set. In text modeling, to eliminate the fluctuation caused by the different lengths of documents, the *perword-perplexity* is often used instead, i.e., n_{test} is the number of all the tokens in the test document collection. In this paper, we use $\log(\text{perword-perplexity})$ and call it *LogPerp*. A lower value on LogPerp indicates better prediction performance.

5.2.2 Temporal Criteria

We define three types of temporal criteria to evaluate the time dependency and model smoothness between time epochs.

Intra-corpus temporal correlation (IntraCorr) is defined to describe the average correlation between adjacent epochs within a corpus:

$$\text{IntraCorr} \triangleq \sum_{t,j} \mathbf{E}[(\pi_j^t)^\top (\pi_j^{t+1})] / \sum_{t,j} 1.$$

Inter-corpora temporal correlation (InterCorr) is defined to describe the average correlation between different corpora of adjacent epochs:

$$\text{InterCorr} \triangleq \sum_{t,j,l} \mathbf{E}[(\pi_j^t)^\top (\pi_l^{t+1})] / \sum_{t,j,l} 1, l \neq j.$$

Global temporal correlation (GCorr) is defined to describe the correlation between global distributions G_0^t of adjacent epochs:

$$\text{GCorr} \triangleq \sum_t \mathbf{E}[(\beta^t)^\top (\beta^{t+1})] / \sum_t 1$$

Above three criteria measure the time dependencies from the as-

Table 1: Synthetic data set.

Global components (dishes)								
k	1	2	3	4	5	6	7	8
$\phi_{k,1}$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
Local components (tables) and corpora sizes								
t	Tables (k_1^t, k_2^t, k_3^t)			Corpora sizes n_j^t				
	$j=1$	$j=2$	$j=3$	$j=1$	$j=2$	$j=3$		
$t=1$	1, 2, 3	2, 3, 4	3, 4, 5	500	300	400		
$t=2$	2, 3, 4	3, 4, 5	4, 5, 6	510	320	430		
$t=3$	3, 4, 5	4, 5, 6	5, 6, 7	520	320	430		
$t=4$	4, 5, 6	5, 6, 7	6, 7, 8	530	340	450		

Table 2: Results on the synthetic data: local components and sizes.

t	HDP: $k(\text{size})$		
	$j=1$	$j=2$	$j=3$
$t=1$	1 (111), 2 (118), 6 (121)	2(70), 14(5), 16(66), 17(69)	4(98), 6(85), 14(97)
$t=2$	3 (110), 5 (120), 14 (1), 16 (126)	5(66), 14(73), 16(85)	5(111), 10(84), 14(106)
$t=3$	9 (119), 12 (131), 16 (114)	8(2), 10(64), 14(84), 16(1), 17(79)	8(90), 10(103), 14(115)
$t=4$	12 (117), 13 (104), 14 (149), 16 (1)	11(56), 14(82), 15(99)	7(112), 8(91), 10(110), 14(2)

t	EvoHDP: $k(\text{size})$		
	$j=1$	$j=2$	$j=3$
$t=1$	1(111), 2(121), 3(118)	2(66), 3(70), 7(72), 8(2)	2(82), 7(106), 8(92)
$t=2$	2(130), 3(110), 7(116), 8(1)	2(82), 7(79), 8(60), 3(3)	5(78), 7(113), 8(107), 2(3)
$t=3$	2(112), 7(135), 8(115), 3(2)	5(62), 7(82), 8(82), 2(2), 4(2)	4(88), 5(103), 8(108), 7(9)
$t=4$	5(93), 7(129), 8(146), 2(3)	4(83), 5(72), 8(77), 7(5)	4(95), 5(109), 6(103), 8(8)

pect of correlation. Higher values on them are favored when static performances are similar.

We also define another three types of criteria from the aspect of divergence between adjacent epochs’ distributions. On the contrary, lower values on them are favored when static performances are similar. They are defined as follows.

Intra-corpora temporal KL divergence (IntraKL)

$$\text{IntraKL} \triangleq \sum_{t,j} \mathbf{E} [\text{KL}(\pi_j^t | \pi_j^{t+1})] / \sum_{t,j} 1.$$

Inter-corpora temporal KL divergence (InterKL)

$$\text{InterKL} \triangleq \sum_{t,j,l} \mathbf{E} [\text{KL}(\pi_j^t | \pi_l^{t+1})] / \sum_{t,j,l} 1, l \neq j.$$

Global temporal KL divergence (GKL)

$$\text{GKL} \triangleq \sum_t \mathbf{E} [\text{KL}(\beta^t | \beta^{t+1})] / \sum_t 1.$$

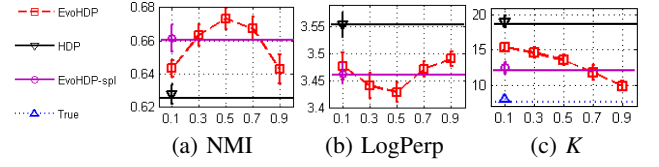
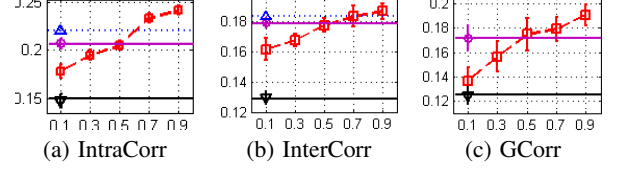
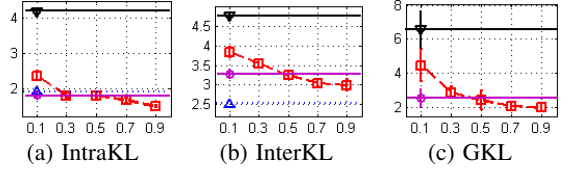
where $\text{KL}(\cdot | \cdot)$ is the KL-divergence between two distributions.

In all the temporal criteria defined above, the expectations are calculated by MCMC samples obtained during the cascaded Gibbs sampling process.

5.3 Settings and Results

Except for EvoHDP, we also ran a three-layer HDP, which is just a special case of EvoHDP when all the time dependencies are removed. All the settings for EvoHDP and HDP are the same. The component model $F(x|\phi_k)$ was set to a multinomial distribution, and the base measure H was set to the conjugate prior for F , a symmetric Dirichlet distribution with parameter 0.5. The vague gamma priors in Eq. (28) for the concentration parameters of EvoHDP and HDP were set to be $Ga(10.0, 1.0)$. An identical set of randomly generated component assignments was used to initialize both EvoHDP and HDP. In addition, to study the impact of time dependency parameters w^t and v_j^t , we also set $w^t = v_j^t = \omega$ as a controlling parameter and swept ω in $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. We call the EvoHDP model with $w^t = v_j^t = \omega$ as “EvoHDP”, and call the EvoHDP model with w^t and v_j^t also sampled during the inference as “EvoHDP-spl”.

For the Gibbs sampling procedure of both models, we ran 20 chains, and set the burn-in time as 1000 for each chain. After the burn-in, from each chain another 50 MCMC samples were preserved, then we obtained 1000 MCMC samples to calculate the evaluation criteria. The models were evaluated via 10-fold cross validation, and all criteria results were averaged on the 10 rounds.

**Figure 6: Results on the synthetic data set: static performances, averaged on 10-fold cross validation.****Figure 7: Results on the synthetic data set: temporal correlations, averaged on 10-fold cross validation.****Figure 8: Results on the synthetic data set: temporal divergences, averaged on 10-fold cross validation.**

For the legends in Figs. 6–9, the red dashed lines with errorbars are the results of HDP, and the purple lines with circles are the results of EvoHDP-spl, and the black lines with down-triangles are the results of HDP. In addition, the blue dotted lines with up-triangles are true values.

The static performances with different ω ’s are illustrated in Fig. 6. In a wide range of ω , EvoHDP achieves better results than HDP on NMI and LogPerp. The estimated component numbers are shown in Fig. 6(c). We see that HDP tends to fit data by splitting into more components, and EvoHDP seems to eliminate this phenomenon.

The temporal performances are illustrated in Figs. 7 and 8. The “true” values on the correlation and divergence criteria are calculated via the ground truth labels. EvoHDP achieves higher correlations and lower divergences than HDP, where larger ω gives stronger dependency between adjacent epochs. However, when ω becomes larger than a threshold (e.g. larger than 0.7), it seems it hurts the correlation criteria between epochs.

Tab. 3 and Tab. 2 further illustrate details of the clustering results. The two tables list a typical clustering result in one trial with $\omega = 0.8$. HDP produces much more components, and lots of them are actually similar and should be merged together.

6. EXPERIMENTS ON REAL DATA

In this section, we report the experiments on a real online document collection. This data set consists of 103,986 text articles queried from a search engine, *Boardreader*⁵, in which the time stamps of the articles are ranged from July 2008 to December 2008, using 20 financial companies’ names, e.g., “AIG insurance”, “Bank of America”, “State Farm”, etc. All these articles come from three types of public websites, i.e., news, blogs and message boards. We used the *Mallet* [13] package to pre-process the data set. We removed the stop words and rare words (appearing less than 10 times in the whole collection). After that, the vocabulary size of this text data set was $W = 77,999$. Term frequencies were extracted to represent each article. We organized the data set into

⁵<http://boardreader.com/>

Table 3: Results on the synthetic data: global components.

HDP																	EvoHDP									
k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	k	1	2	3	4	5	6	7	8
$\phi_{k,1}$.10	.19	.19	.41	.40	.30	.80	.70	.50	.60	.60	.40	.60	.50	.70	.30	.40	$\phi_{k,1}$.10	.30	.19	.71	.61	.81	.40	.51
Sizes	111	188	110	98	297	206	112	183	119	361	56	248	104	714	99	393	148	Sizes	111	601	303	268	517	103	846	798

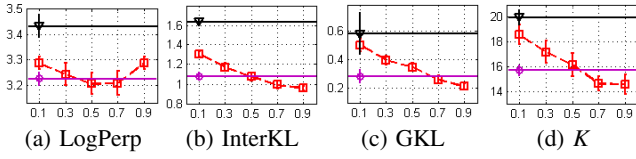


Figure 9: Results on real bank data set, averaged on 10 rounds.

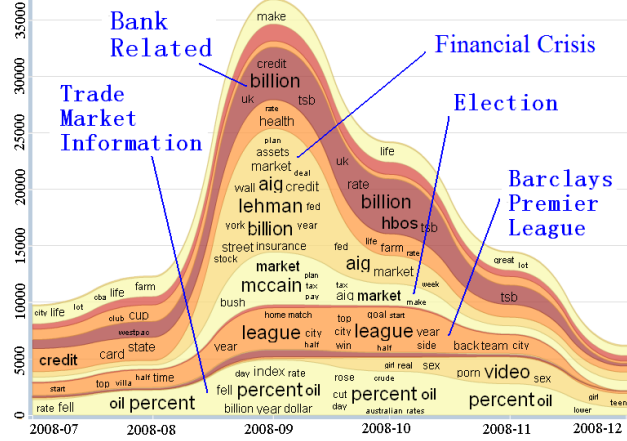


Figure 10: Overview of the overall clusters in all text corpora of all epochs. Each colored stripe represents a cluster, whose height is the number of articles assigned to the cluster. For each cluster, the top keywords of this cluster in each month are placed on the stripe, where the size of a keyword is proportional to its frequency in the cluster.

6 epochs by months, and each epoch was set as a month. Hence we obtained a data set including 3 corpora along the 6 epochs. We regarded each article (represented by a vector of term frequencies) as a draw from a multinomial distribution with dimension W , i.e., $F(x|\phi_k)$ is a multinomial distribution. The base measure H was set to the conjugate Dirichlet prior with symmetric parameter 0.5.

To compare EvoHDP with HDP using numerical criteria, we randomly sampled a smaller subset, which consisted of 5,353 articles with 19,420 distinct words as the vocabulary. Experimental settings and parameters were the same as those in Sec. 5.3. All the results were also averaged on 10-fold cross validation. The numerical results are shown in Fig. 9. Since there is no ground truth labels for the data set, NMI can not be calculated and only the results of LogPerp are given. Limited to space, only two temporal criteria are provided here, i.e., InterKL, and GKL. Consistent with the results on toy data, EvoHDP achieves lower LogPerp values and higher correlations (low divergence values). The phenomenon is also observed here that HDP tends to split up into more components (Fig. 9(d)) to fit data and overlook the correlations.

To have a clear insight into the evolution patterns in the multiple correlated corpora discovered by EvoHDP, we further leveraged the time-based topic visualization proposed by Liu et al. [12] to visually illustrate the analysis results on all the 103,986 text articles in several different views.

First, we present the overview of all text corpora including news, blogs and message boards, which is shown in Fig. 10. We can see five active clusters about “financial crisis”, “election”, “trade market information”, “Barclays premier league” and “bank related”. These five clusters are all finance related. For example, the “elec-

tion” cluster tells about Obama and McCain’s debate on bailout agreement, and the “financial crisis” cluster tells about the bankrupt events of largest companies such as Lehman and AIG.

Then, we present the clusters of different corpora, i.e. news, blogs and message boards respectively. In Figs. 11 (a),(b), and (c), we present the absolute values of document numbers in each cluster at each epoch. In Figs. 11 (d),(e), and (f), we present the mixing proportions of clusters in a corpus at each epoch.

From these figures we find several interesting patterns. (1) **The three corpora are similar but diversity exists.** All of the three corpora are interested in “financial crisis”. However, blogs and message boards focus more on “election” than news; news and message boards focus more on “Barclays premier league”; and news focuses more on general “bank related information” and “trade market information”. This may be because news tends to report more time-sensitive events, while blogs and message boards like to have a deep discussion on a particular event or the progress of an affair. (2) **Correlations over time and across different corpora.** For each corpus, the clusters change smoothly along time. Since we added the time dependencies between adjacent epochs, the cluster content does not change too much. Moreover, there are some clusters shared across different corpora. (3) **Cluster emerging and disappearing.** The cluster “financial crisis” emerged in news and message boards in September due to the bankrupt of two financial companies, Lahman and AIG. The cluster “election” emerged in blogs and message boards in September 2008 due to the televised presidential debate between Obama and McCain. We also note that there emerged a strange cluster (“ads/noise”) in blogs from October, with keywords like “movies”, “videos”, “sex” etc. . When we digged into the content, we found that the articles were full of noisy advertisement information. This may be because the blogs became very hot after September 2008, and then more and more automatic robots came to the site and presented nonsense information. In addition, we also observe that the cluster “Barclays premier league” disappeared in December, after Barclays determined to renew its sponsorship of the premier league and a new league season started.

To better compare the evolving behaviors of different clusters, we select three clusters from different corpora in the same view. The three clusters “Barclays premier league”, “election”, and “financial crisis” are shown in Fig. 12(b), Fig. 12(a) and Fig. 1, respectively. From the three figures, besides clearer witness of the previous three patterns, another two patterns are also discovered. (4) **Cluster evolution within a corpus.** First, the strength of a cluster in corresponding corpus varies overtime, which can be clearly observed in all the three figures. Second, the most frequent keywords of a cluster also vary over time, reflecting the evolving of the content of a cluster. Taking the cluster “election” (Fig. 12(a)) as an example, in blogs, in August, the keywords “Obama”, “Bush”, “presidential”, “democratic” indicated the normal features before an election. However, in September, “McCain” appeared as the hottest keywords as Republicans nominated John McCain for president in September 4th. Besides, crisis related keywords such as “crisis”, “aig”, “(wall) street” became hot due to the break-out of financial crisis in this month. More such features can also be observed in later months and other clusters. (5) **Cluster evolution across different corpora.** In Fig. 1 and Fig. 12(a), it is clear that both “crisis” and “election” clusters were first active in blogs, and then became popular in news and message boards.

