Hierarchical evolving Dirichlet processes for modeling nonlinear evolutionary traces in temporal data

Article *in* Data Mining and Knowledge Discovery · February 2016

DOI: 10.1007/s10618-016-0454-1

5 authors, including:

Chuan Zhou
Chinese Academy of Sciences
38 PUBLICATIONS **202** CITATIONS

Noname manuscript No.
(will be inserted by the editor)

Hierarchical Evolving Dirichlet Processes for Modeling Nonlinear Evolutionary Traces in Temporal Data

Peng Wang, Peng Zhang, Chuan Zhou, Zhao Li

Received: date / Accepted: date

Abstract Clustering analysis aims to group a set of similar data objects into the same cluster. Due to the dynamic nature of temporal data, clusters often exhibit complicated patterns such as *birth, branch and death*. However, most existing temporal clustering models assume that clusters evolve as a linear chain, and they cannot model and detect branching of clusters. In this paper, we present Evolving Dirichlet Processes (EDP for short) to model nonlinear evolutionary traces behind temporal data, especially for temporal text collections. In the setting of EDP, temporal collections are divided into epochs. In order to model topic branching over time, EDP allows each cluster in an epoch to form Dirichlet Processes (DP) and uses a combination of the cluster-specific DPs as the prior for cluster distributions in the next epoch. To model hierarchical temporal data, such as online document collections, we propose a new class of Evolving Hierarchical Dirichlet Processes (EHDP for short) which extends the Hierarchical Dirichlet Processes (HDP) to model evolving temporal data. We design an online learning framework based on Gibbs sampling to infer the evolutionary traces of clusters over time. In experiments, we validate that EDP and EHDP can capture nonlinear evolutionary traces of clusters on

Peng Wang
Alibaba Group, Hangzhou, 311121, China & Institute of Information Engineering, CAS, Beijing 100193, China.
E-mail: peng860215@qq.com

Peng Zhang
University of Technology, Sydney, NSW 2007, Australia
E-mail: peng.zhang@uts.edu.au

Chuan Zhou
Institute of Information Engineering, CAS, Beijing 100193, China.
E-mail: zhouchuan@iie.ac.cn

Zhao Li
Alibaba Group, Hangzhou, 311121, China
E-mail: lizhao.lz@alibaba-inc.com

both synthetic and real-world text collections and achieve better results than its peers.

Keywords Dirichlet processes, temporal topic models, Chinese restaurant processes, soft clustering for texts.

1 Introduction

With the increase of online data such as blogs, news stories and user behavior records, there is an urgent demand for developing automatic analysis models that can organize their contents. Clustering methods [9,12] have been proved effective tools to conduct the analysis by projecting each data point into a latent space, i.e., clusters or latent topics. For temporal data, clusters tend to evolve over time. Most models can only organize topics in linear chains [21, 4,3,14,7,44]. However, in reality, clusters over time often exhibit complicated nonlinear evolutions[36,19], which has not been fully addressed before. Therefore, in this paper, we study the problem of modeling nonlinear evolutionary traces behind clusters.

Nonlinear structural evolution commonly occurs behind temporal data, where clusters may *birth, evolve, branch and die out* over time. We use the topics of the papers from the NIPS conferences between the years 2000 to 2012 as our motivating example. Due to the rapid development of the research on *neural networks*, new topics rise while obsolete research topics disappear constantly. Figure 1 shows that the distributions of paper topics evolve gradually during the 13 years. Moreover, a mature research topic often evolves into several new topics. Figure 2 shows a specific example of the non-linear topic evolutionary traces where some topics branch into new independent topics.

Temporal data such as the NIPS data above are dynamic in nature and modeling evolutionary traces of topics (clusters) should incorporate three facets of ***dynamics*** as follows,

- **Number of clusters.** As clusters birth, branch and die over time, the number of clusters is not fixed. The model should be able to adjust the number of clusters accordingly. It also implies that the traditional parametric Bayesian models [12,11] would fail in modeling temporal data because they require to set the number of clusters manually.
- **Cluster parameters.** For a cluster which lasts for a couple of epochs, its parameters evolve over time. As shown in Figure 1, the research on *neural networks* has been lasting for many years, but the actual research topics change from one to another continuously. To describe this difficulty, the parameters of a topic should be smoothed over time to ensure the topic membership stability [3,45], and the local features of a topic should also be exploited.
- **Clusters distribution.** The dynamics of cluster distributions plays a key role in capturing the evolving trend behind temporal data. Due to the observation that a rich topic often gets richer, the distribution of a cluster

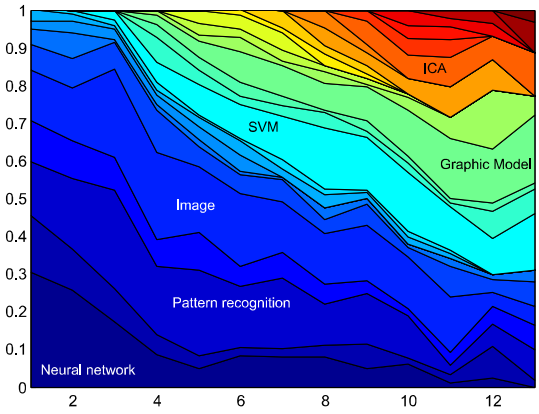


Fig. 1 Topic distributions of the NIPS papers published from year 2000 to 2012. The research topics of NIPS evolve gradually from Neural networks, to Image/Pattern recognition, SVM, ICA and Graphical models during the 13 years. The data set is from `http://www.cs.nyu.edu/~roweis/data.html`

is likely to burst or gradually decrease over time. The non-linear model should cover this scenario and smooth the cluster distributions over time.

The above challenges are highly interdependent. Most approaches proposed for evolutionary clustering reduce the above challenges to the smoothness constrains over time in terms of cluster distribution, cluster parameters, and the number of clusters [3,33]. However, according to our observation, it would be more advantageous for evolutionary clustering models to incorporate new features brought by the nonlinear evolution.

- **Uncertain number of branches.** For each cluster in nonlinear evolutionary clustering model, there are potentially unlimited number of inheritor clusters. Therefore, we need to dynamically determine the number of clusters. Also, we need to control propensity of creating new clusters; otherwise, evolutionary trees can be huge and the trace recovered may become meaningless.
- **Parameter evolution of branching clusters.** Most previous temporal clustering models assume that the parameters of a cluster evolve in random-walk [3,4], and they adopt linear time series models, e.g., Kalman filter [34], to infer the parameter of evolutionary clusters. These methods may be inapplicable when clusters branch and form evolutionary trees.

A handful of temporal topic models have been proposed to model evolutionary clusters. Blei et.al. [11] proposed the Dynamic Topic Models, which is an extension of the Latent Topic Models in temporal scenarios. However, parametric Bayesian models require setting the number of topics manually. Non-parametric Bayesian models [3,23,21,33] can auto-learn the number of cluster over time, so they are widely used to model evolutionary documents in recent years. Ahmed et. al. proposed the Recurrent Chinese Restaurant Processes (RCRP) [3] and the Recurrent Chinese Restaurant Franchise Processes (RCRF) [4] which can recover the storyline of topics in linear chains. However, both RCRP and RCRF [4] fall short in modeling nonlinear evolutionary patterns because each topic is allowed to have at most one inheritor. Sun et al.[36]

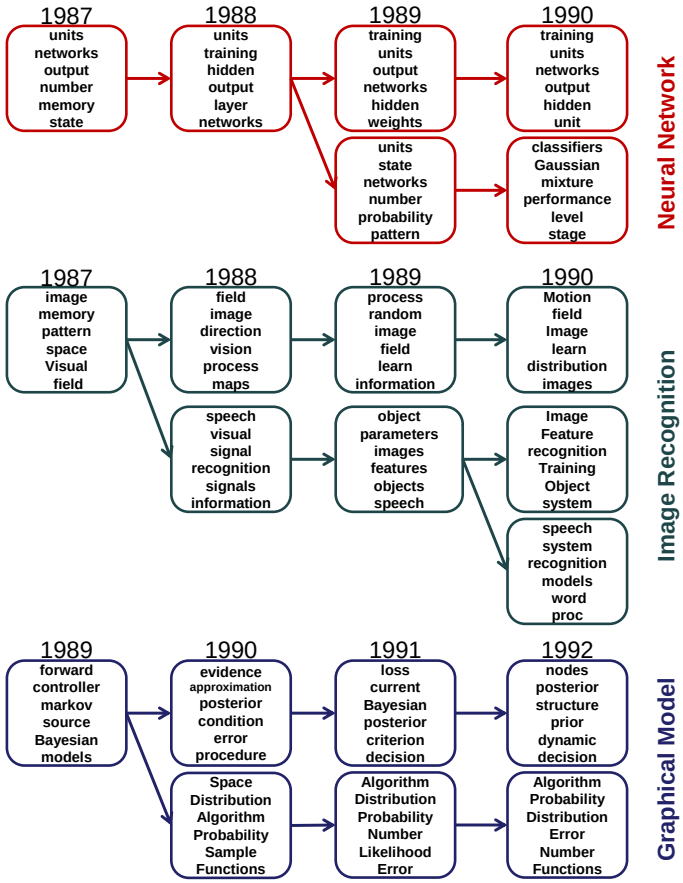


Fig. 2 An illustration of the non-linear topic evolutionary traces behind the NIPS papers.

took one step forward and proposed the DP-NetClus model which can infer complex relation between topics. However, DP-NetClus is unable to control the scale of branches of topics. As a result, the evolutionary traces recovered by DP-NetClus are generally incomprehensible. To sum up, to the best of our knowledge, existing temporal clustering models cannot fully solve the proposed nonlinear topic evolution problem.

In this paper, we propose a new class of Evolving Dirichlet Processes (EDP in short) which can model nonlinear cluster branching over time. In the setting of EDP, temporal data are divided into epochs. In order to model cluster branching over time, EDP lets each cluster in an epoch form a Dirichlet process and uses a combination of the cluster-specific DPs as the prior for topic distribution in the next epoch. In doing so, a cluster can branch into several new clusters, and we can control the expected scale of inheritors by using the concentration factor in DP.

Many applications face hierarchical data where each data record is generated from a mixture of clusters. For example, text collections are organized as corpus-document-word levels, where each document is generated with a mixture of topics. To model evolutionary traces behind hierarchical data sets, we propose the Evolving Hierarchical Dirichlet Processes (EHDP). EHDP uses the similar evolutionary scheme of EDP, but they extend Hierarchical Dirichlet Processes (HDP)[38,39] for modeling hierarchical data in each epoch.

We propose an inference algorithm for both EDP and EHDP models based on the Gibbs sampler. As the sample-based inference algorithm are generally time demanding, we design an online learning framework, which can update the evolutionary traces of a cluster incrementally. In experiments, we demonstrate that EDP and EHDP can capture nonlinear evolutionary traces of clusters/topics on both synthetic and real-world data sets and outperform the state-of-the-art methods.

Teh et.al. proposed the Hierarchical Dirichlet Processes (HDP) [38,39] which can theoretically handle hierarchical data with arbitrary levels. Previous models, such as RCRF [4] and DP-NetClus [36], can also handle two-level hierarchical text collections. This hierarchical feature would not alter the evolutionary pattern of temporal data, but it will increase the complexity of the inference procedure [28]. For real-world applications, the volume of temporal data set is generally infinite. So a scalable inference algorithm is preferred.

The rest of the paper is organized as follows. Section 2 surveys related work. Section 3 introduces the problem settings and reviews necessary backgrounds. Section 4 introduces the Evolving Dirichlet processes. Section 5 describes the Evolving Hierarchical Dirichlet processes for text collections. And we present a online learning framework which can incrementally updating the evolutionary trace in EDP and EHDP. Section 6 conducts experiments. We conclude the paper in Section 7.

2 Related Work

Temporal topic models. Temporal topic models [11,47,44,23,18,8,49,27, 17] aim to not only recover latent clusters of words in each epoch, but also recover dynamic patterns of distributions and parameters of topics over time. Dynamic Topic Model (DTM)[11] is an extension of LDA for modeling temporal data sets. DTM falls into the category of parametric topic models. So the number of topics is fixed over time. In fact, all parametric topic models such as the topics over time model (ToT) [44,50] and the trend analysis model (TAM) [23] face the same problem.

Nonparametric models introduce a concentration factor to control the scale of topics instead of fixing the number of topics. So the structure of topics is determined by data. Most of the Nonparametric dynamic topic models are developed from Hierarchical Dirichlet processes. We have introduced RCRP and RCRF processes [3,4], which can recover the topic chains from text collections. Similarly Chen et.al. proposed the Dependent Hierarchical Normalized

Random Measures [15] which can capture the power-law property of topics and topic-word distributions, the topics are also organized as chains. Other models, such as the dynamic hierarchical Dirichlet processes [33], the order-based Dirichlet processes [21], the dependent Dirichlet processes [30,29], the coalescent models [13,25,26] also suffer this problem. Some researchers have tried to recover the complex topic connections between epochs, such as the DP-NetClus [36] and incremental Gibbs sampling algorithm. However, DP-NetClus cannot control the number of branches. The test results show that most of the topics inherit from a couple of topics in previous epoch, and evolutionary traces are ambiguous. The Dirichlet diffusion trees [32], which is also a nonparametric model, can model the density of data with a tree structure. But the density tree of static data is fundamentally different with the evolutionary tree of temporal data.

There are other methods beyond the Bayesian framework that can trace evolution of topics or clusters[16]. Such as the information cartography proposed by Shahaf et.al. [35], and evolutionary co-clustering [51]. These methods also cannot recover tree-like evolutionary trace of topics.

Hierarchical clustering models. Hierarchical clustering models [38,43,37, 40,9,2] aim to model hierarchical structures within group of data. Teh et al. [38] proposed the Hierarchical Dirichlet Processes (HDP). HDP is designed to model hierarchical data, such as text collections. HDP can model hierarchical data with any depth. Blei et al. [9] proposed the nested Chinese Restaurant Processes (nCRP) which assumes that a group of data can be divided into clusters and each cluster can be further divided into smaller clusters recursively. All these clusters can be organized as a tree in which child clusters belonged to their parent clusters. nCRP achieves great success in analyzing text collections and user profile data [2]. The model inspires to extend the conjugate prior design for modeling parameter evolution in EDP and EHDP.

Model learning. Learning algorithms play an important role in nonparametric models. They can greatly improve the performance and scalability of the models. Most models use either Gibbs sampling[31,37,28,48,22] or variational inference for learning. Teh et al. [38] proposed three Gibbs samplers to handle the CRF process. Li et al. [28] also presented an effective method which greatly reduces the sampling cost. Compared with the Gibbs sampler, variational inference for nonparametric topic models [10,39] is generally much faster but can only reach approximate results.

Online inference is another important issue for temporal topic models [6, 1,42]. As temporal document collections keeps arriving, we need to constantly update the topic models. Ahmed et al. adopted the sequential Monte Carlo method for online estimation of RCRP [1], which is executed in parallel by allocating particles to cores. Similarly, Wang et.al. [42] proposed online inference model for HDP with mini-batch update. We also propose the online inference algorithm for our EDP and EHDP models.

3 Preliminary

In this section, we first introduce the nonlinear evolution problem and then briefly review the Dirichlet Processes and Hierarchical Dirichlet Processes to make the paper self-contained.

3.1 Problem Settings

In this paper, we aim to model an ordered set of data $\mathbf{X} = \{\mathbf{x_1}, \cdots, \mathbf{x_T}\}$, where T is the number of epochs and $\mathbf{x}_t$ denotes the data at epoch t. $\mathbf{x_t} = (\mathbf{x_{t,i}})_{i=1}^{N_t}$, where N_t is the number of data records at epoch t. The temporal data are partially exchangeable, i.e., data within the same epoch are exchangeable but cannot be exchanged across epochs. We let $\theta_{t,i}$ denote the cluster identifier associated with data $x_{t,i}$. For a cluster k in epoch t, its parameter is denoted as $\phi_{t,k}$. We assume that the cluster can emerge, branch and die over time. *Our goal is to discover the nonlinear evolutionary traces of clusters* $\mathbf{\Phi}$.

For hierarchical data, we focus on temporal text collections. To distinguish with the previous setting, here we use $\mathbf{W} = \{\mathbf{w_1}, \cdots, \mathbf{w_T}\}$ to denote the text collections. $\mathbf{w_t} = (\mathbf{w_{td}})_{d=1}^{N_t}$, where N_t is the number of documents in epoch t. Each document contains N_{td} words, i.e., $\mathbf{w_{t,d}} = (w_{tdi})_{i=1}^{N_{td}}$. θ_{tdi} denotes the topic identifier for each word w_{tdi}. Other settings are the same with the non-hierarchical scenario.

3.2 Dirichlet Processes

Dirichlet processes is a distribution over distributions. It can be taken as an infinite-dimension generalization of the Dirichlet distribution. In the same way as the Dirichlet distribution is the conjugate prior for the categorical distribution, the Dirichlet processes is the conjugate prior for infinite, nonparametric discrete distributions.

A DP, denoted as $DP(\gamma, H)$, is parameterized by a base measure H, and a concentration parameter γ. We use $G \sim DP(\gamma, H)$ for drawing a distribution G from DP. G is an infinite discrete distribution which can be treated as the cluster distribution. By using the DP as the prior for cluster distribution, we can obtain a nonparametric mixture model which allows infinite number of clusters. The generative process for the DP mixture model proceeds as follows:

$$\begin{aligned} G|\gamma, H \sim DP(\gamma, H), \qquad & \theta_i|G \sim G, \\ \phi_k|H \sim H, \qquad & x_i \sim F(\phi_{\theta_i}) \end{aligned} \tag{1}$$

where F is a given pdf parameterized by ϕ_{θ_i}. For instance in the case of generating discrete data, F is multinomial pdf, and ϕ is the multinomial parameter.

There are three equivalent construction procedures for DP: Polya urn construction, stick-breaking construction, and Chinese Restaurant Process. Here

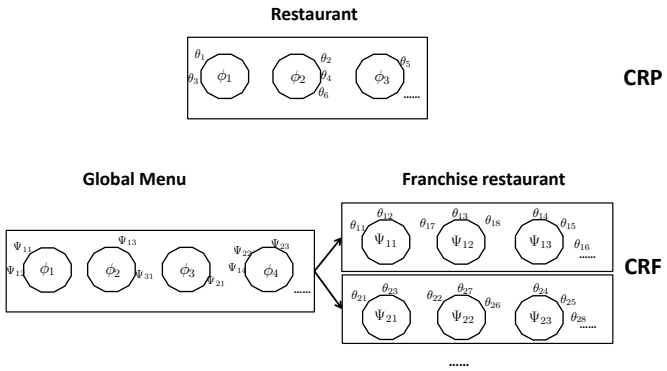


Fig. 3 An illustration of CRP and CRF.

we introduce the Chinese Restaurant Processes (CRP), as we will present an analog construction for our evolutionary clustering model in Section 4.

Construct DP with CRP. The generative process of CRP is described by imagining a restaurant serving with infinite number of tables. In this metaphor, a customer corresponds to a datum x; and a table ϕ corresponds to a cluster (as depicted in Figure 3). And the process of a customer picking a table corresponds to generating a datum for a cluster. CRP proceeds as follows. When a customer comes into the restaurant, he chooses a table. The probability of choosing the table k is proportional to n_k, which denotes the number of customers already sitting around the table. While there is nonzero probability, which is proportional to γ, that the customer picks an empty table ϕ_{K^+}. Picking a new table corresponds to the creation of a new cluster, whose parameter is generated as $\phi_{K^+} \sim H$. So in the limit of $\gamma \to 0$, the data are all concentrated on a single cluster, while in the limit of $\gamma \to \infty$, each datum forms a cluster all by itself. The process can be summarized as:

$$\theta_i|\theta_{1:i-1}, H, \gamma \sim \sum_k \frac{n_k^{(i)}}{i-1+\gamma}\delta(\phi_k) + \frac{\gamma}{i-1+\gamma}H \tag{2}$$

CRP generates a static data set with unbounded number of clusters in this procedure.

3.3 Hierarchical Dirichlet Processes.

Hierarchical Dirichlet processes (HDP) is an extension of DP for hierarchical data. Take text collections as an example, where texts are organized with corpus-document-word layers. HDP defines a global random measure G, which describe the topic distribution at the corpus level. G generates a set of random measures G_d, one for each document. G_d models the topic distributions at the document level. Each word w_{di} is associated with a topic θ_{di} which is sampled

from G_d. The topics are shared across documents, as G_d are drawn from G as $G_d \sim DP(\alpha, G)$, where α is a concentration factor. The global measure G is also generated from a corpus-level DP as $G \sim DP(\gamma, H)$. In summary, we define the generative process of HDP as follows,

$$\begin{aligned} G|\gamma, H \sim DP(\gamma, H), \quad G_d|\alpha, G \sim DP(\alpha, G) \\ \theta_{di}|G_d \sim G_d, \quad \phi_k|H \sim H, \\ w_{di}|\theta_{di} \sim F(\phi_{\theta_{di}}), \end{aligned} \tag{3}$$

Construct HDP with CRF. HDP can be constructed with the Chinese Restaurant Franchise processes (CRF), which is depicted in Figure 3. In the metaphor of CRF, a restaurant franchise corresponds to a corpus, and each restaurant corresponds to a document. A global menu of dishes in the restaurant corresponds to a topic ϕ_k. A customer corresponds to a word in a document. And the process of a customer picking a table corresponds to generating a word with a topic. All the customers for the table b of restaurant d share the topic ψ_{db}. In particular, we need to maintain the counts of customers and tables. n_{dbk} denotes the number of customers in the restaurant d at table b eating dish k. m_{dk} denotes the number of tables in the restaurant d serving dish k. And the marginal counts are represented with dots. Thus, $n_{db\cdot}$ represents the number of customers in the restaurant d at table b, and so on.

In CRF process, the conditional distribution for the word's table/topic selection θ_{di} given $\theta_{d1}, \cdots, \theta_{d,i-1}$ and G is given in Eq.(4),

$$\theta_{di}|\theta_{d,1:i-1}, \alpha, G \sim \sum_{b=1}^{m_{d\cdot}} \frac{n_{db\cdot}}{i-1+\alpha}\delta_{\psi_{db}} + \frac{\alpha}{i-1+\alpha}G \tag{4}$$

Where ψ_{db} is the topic on table b for restaurant d. If a term in the first summation is chosen then we set $\theta_{di} = \psi_{db}$. If the second term is chosen, then we increase $m_{d\cdot}$ by one, draw a dish for the new table b^{new} as $\psi_{db^{new}} \sim G$ and set $\theta_{di} = \psi_{db^{new}}$. Now we can instantiate G and obtain the conditional distribution of $\psi_{db^{new}}$ as in Eq.(5).

$$\begin{aligned} \psi_{db^{new}}|\psi_{1:d-1,\cdot}, \psi_{d,1:m_{d\cdot}-1}, \gamma, H \sim \\ \sum_{k \in K} \frac{m_{\cdot k}}{m_{\cdot\cdot}+\gamma}\delta_{\phi_k} + \frac{\gamma}{m_{\cdot\cdot}+\gamma}H \end{aligned} \tag{5}$$

If we draw $\psi_{db^{new}}$ via choosing a term in the summation on the right-hand side of this equation, we set $\psi_{dm_{d\cdot}} = \phi_k$. If the second term is chosen then we increase K by one, draw $\phi_K \sim H$ and set $\psi_{db^{new}} = \phi_K$. Eq.(3), Eq.(4) and Eq.(5) together describe the CRF construction of HDP. By this procedure, HDP can generate hierarchical data set with unbounded number of clusters.

In Sections 4 and 5, we present EDP and EHDP which extend DP and HDP for modeling nonlinear evolving temporal data.

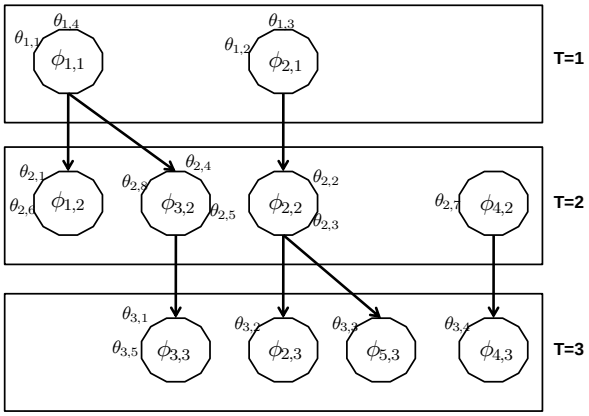


Fig. 4 An illustration of EDP. The figure depicts the construction procedure based on CRP. The circles represent clusters. The clusters form three independent evolutionary trees over three epochs.

4 Evolving Dirichlet Processes

In this section, we first introduce the Evolving Dirichlet Processes (EDP) which aim to model the nonlinear evolutionary trace of temporal data. Then we introduce how to model the evolution of cluster parameters and the inference algorithm based on Gibbs sampling.

4.1 Generation Procedure of EDP

The key to discover nonlinear evolutionary traces is to model cluster branching. In EDP, we let each cluster form a specific DP for the next epoch, so a cluster can branch multiple inheritors in the next epoch. And we adopt the posterior of parent cluster as the base measure for the cluster specific DP so as to smooth the cluster parameter over time. In the sequel, we present the generation procedure of EDP.

EDP is operated in epochs. Let G_t denote the distribution of clusters and their parameters in epoch t. As a time-varying variable, G_t is distributed conditioned on G_{t-1} and H, and H is the base measure which serves as the prior for topic parameters. As a cluster either inherits pre-existing cluster or emerges as a orphan cluster, G_t contains two parts. The first part is $DP(\gamma, H)$ which generates orphan clusters. The second part is a set of cluster-specific Dirichlet processes which can generate branch clusters. For generating a datum $x_{t,i}$, we first pick a cluster $\theta_{t,i}$ based on G_t, then generate its value based on cluster parameter $\phi_{t,\theta_{t,i}}$. In summary, EDP can be formally defined as follows (the graph model of EDP is given in Figure 5),

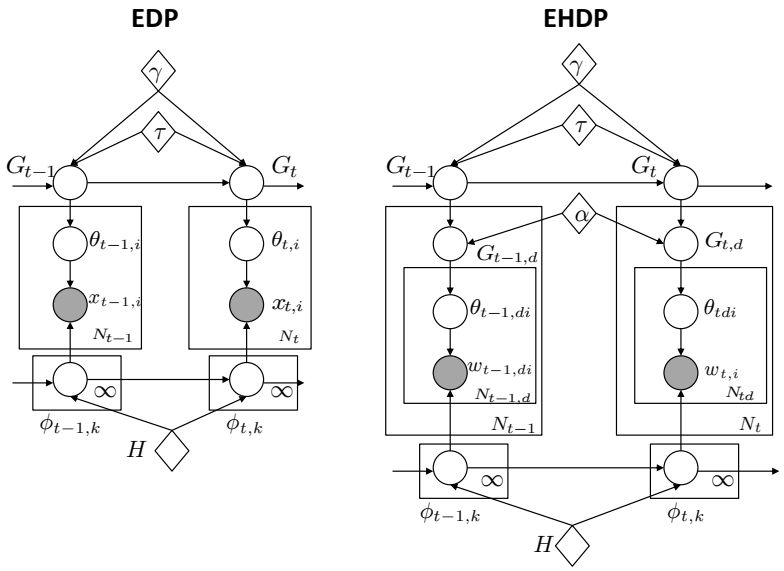


Fig. 5 the graphical models of EDP and EHDP. The diamonds represent hyper-parameters, hollow circles represent latent states of the models, and solid circles represent observations.

$$
\begin{aligned}
G_t|\gamma,\tau,G_{t-1},H \sim & \sum_{k\in G_{t-1}} \beta_{t-1,k} DP(\tau, H_{\phi_{t-1,k}})+ \\
& \beta_{tO} DP(\gamma, H), \\
\theta_{t,i}|G_t \sim & G_t, \\
x_{t,i}|\theta_{t,i} \sim & F(\phi_{t,\theta_{t,i}})
\end{aligned}
\tag{6}
$$

where $\beta_{t-1,k}$ and $\beta_{t,O}$ are the weights of CRPs.

4.2 Construct EDP with CRP

Eq.(6) describes the generation procedure of EDP. Each DP in Eq.(6) can be constructed with the Chinese Restaurant Processes as described in Section 3.2. In the procedure, two problems need to be settled. First, how to determine the weight of DP, i.e., $\beta_{t-1,k}$ and $\beta_{t,O}$. Second, what is the conditional distribution for a data record's cluster selection $\theta_{t,i}$ based on G_t.

Weight of DPs in EDP. If a cluster is large, it is likely to have more branches. We assume that the weight of $DP(\tau, H_{\phi_{t-1,k}})$, i.e., $\beta_{t-1,k}$, is proportional to the number of records belonging to cluster $\phi_{t-1,k}$ in $t-1$, and the weight of the $DP(\gamma, H))$, i.e., $\beta_{t-1,O}$, is proportional to γ. So, G_t can be expressed with $\beta_{t-1,k}$ and $\beta_{t,O}$ instantiated, as in Eq.(7),

$$G_t \sim \frac{1}{\gamma + \eta \sum_k n_{t-1,k}} \big(\sum_k \eta n_{t-1,k} CRP(\tau, H_{\phi_{t-1,k}}) + \gamma CRP(\gamma, H) \big), \tag{7}$$

where the $DP(\cdot)$ is instantiated with the CRP. $\eta \in [0, 1]$ is a decaying factor which influences the smoothness of cluster distribution over time. Specially, for $t = 0$, G_0 is constructed in the same way as CRP.

Conditional distribution of $\theta_{t,i}$. The conditional distribution for a data record's cluster selection $\theta_{t,i}$ can be calculated with G_t instantiated, which further requires instantiating $CRP(\gamma, H)$ and $CRP(\tau, H_{\phi_{t-1,k}})$. Given the cluster assignment of data in epochs $t-1$ and t, i.e. $\theta_{t-1,\cdot}$ and $\theta_{t,1\cdots i-1}$. $CRP(\gamma, H)$ can be expressed as in Eq.(8),

$$\theta_i | \theta_{1:i-1}, H, \gamma \sim \sum_{o' \in O_t} \frac{n_{t,o'}}{\sum_{o' \in O_t} n_{t,o'} + \gamma} \delta_{\phi_{t,o'}} + \frac{\gamma}{\sum_{o' \in O_t} n_{t,o'} + \gamma} H, \tag{8}$$

where O_t is the set of orphan clusters. And $CRP(\tau, H_{\phi_{t-1,k}})$ can be expressed as in Eq.(9),

$$\theta_i | \theta_{1:i-1}, H_{\phi_{t-1,k}}, \tau \sim \sum_{s' \in S_{t-1,k}} \frac{n_{t,s'}}{\sum_{s' \in S_{t-1,k}} n_{t \cdot s'} + \tau} \delta(\phi_{t,s'}) + \frac{\tau}{\sum_{s' \in S_{t-1,k}} n_{t,s'} + \tau} H_{\phi_{t-1,k}} \tag{9}$$

where $S_{t-1,k}$ is a set of variants of $\phi_{t-1,k}$.

It is important to notice that $CRP(\tau, H_{\phi_{t-1,k}})$ and $CRP(\gamma, H)$ are different in two aspects.

- First, the base measure H in $CRP(\gamma, H)$ is a vague prior, while that in $CRP(\tau, H_{\phi_{t-1,k}})$ is determined by $\phi_{t-1,k}$ (we will give describe how to calculate $H_{\phi_{t-1,k}}$ in Section 4.3). For a orphan cluster, we don't have specific knowledge about it, so a vague prior is preferred. But the parameter of an inheritor cluster tends to be similar with its parent.
- Second, the concentration factor γ in $CRP(\gamma, H)$ is larger than τ in $CRP(\tau, H_{\phi_{t-1,k}})$. Because a larger concentration factor leads to more clusters generated by the CRP. The number of variants for a cluster is generally small, while that of an orphan cluster is often large.

Combining Eqs.(7), (8) and (9), the conditional distribution of $\theta_{t,i}$ can be written as in Eq.(10),

$$\begin{aligned} &\theta_{t,i} | \{\theta_{t-1}, \cdot\}, \theta_{t,1:i-1}, H, H_{\phi_{t-1,k}}, \eta, \tau, \gamma \propto \\ &\sum_{k \in \{\phi_{t-1,\cdot}\}} \frac{\eta n_{t-1,k}}{R_{t-1}} \Big(\sum_{s' \in S_{t-1,k}} \frac{n^{(i)}_{t,s'}}{\Delta_{t-1,k}} \delta_{\phi_{t,s'}} + \frac{\tau}{\Delta_{t-1,k}} H_{\phi_{t-1,k}} \Big) \\ &+ \frac{\gamma}{R_{t-1}} \Big(\sum_{o' \in O_t} \frac{n_{t,o'}}{\Lambda_{t-1}} \delta(\phi_{t,o'}) + \frac{\gamma}{\Lambda_{t-1}} H \Big) \end{aligned} \tag{10}$$

where $R_{t-1} = \eta \sum_{k \in \{\phi_{t-1,\cdot}\}} n_{t-1,k} + \gamma$, $\Delta_{t-1,k} = \tau + \sum_{s' \in S_{t-1,k}} n^{(i)}_{t,s'}$, and $\Lambda_{t-1} = \sum_{o' \in O_t} n_{t,o'} + \gamma$.

Finally, we present the how to describe the birth, branch, and die-out of clusters in EDP: 1) if the word distribution of topic $\phi_{t,k}$ is drawn from that of a previous topic $\phi_{t-1,k}$, then $\phi_{t,k}$ is a branch of $\phi_{t-1,k}$; 2) if the word distribution of topic $\phi_{t,k}$ is drawn from that of a vague prior H, then we call $\phi_{t,k}$ is the birth of a topic; 3) if a previous topic $\phi_{t-1,k}$ have no branches in the next epoch, then $\phi_{t-1,k}$ dies-out.

4.3 Modeling Evolving Cluster Parameters.

We adopt the conjugate prior cascade to model the evolution of clusters' parameters in EDP. In Bayesian models, hyper-parameters of a conjugate prior correspond to having observed a certain number of pseudo observations with properties specified by the parameters [20]. Data in ancestor clusters can serve as pseudo observations for the current cluster. So for a cluster, the posterior distribution of its parent cluster can serve as the conjugate prior. Furthermore, the posterior of the current cluster can be fetched by absorbing observations to the prior. Random measure H serves as the prior for all the orphan clusters. Given H and observations, the posterior of clusters can be calculated in a cascaded manner. In the conjugate prior cascade, a decaying factor $\lambda \in [0, 1]$ is introduced to let the model gradually forget observations in previous clusters. The process is similar to the idea of Bayesian sequential updating (BSU for short). However, they are different in two aspects. Firstly, our parameter evolution scheme gradually forgets the historical data, while BSU keeps all the records. Secondly, our scheme can handle the branching of clusters, while BSU is used to updating the parameter of a single cluster.

With different conjugate prior pairs, we can describe parameter evolution of variant types of temporal data. In this paper, we employ normal-inverse-Wishart distribution for modeling continuous data with Gaussian assumption, and we employ Dirichlet distribution for modeling temporal text collections. In the sequel, we present the parameter evolutionary scheme for both cases.

For Multivariate Gaussian. For multivariate Gaussian distribution with parameter $\mathcal{N}(\mu, \Sigma)$, where μ is the mean vector and Σ is the covariance matrix, the conjugate prior is normal-inverse-Wishart distribution, i.e., $\mathcal{N}(\mu, \Sigma) \sim NIW(\mu_0, \kappa_0, \nu_0, \Psi_0)$. The parameter of the prior can be interpreted as follows. the mean is estimated from κ_0 observations with sample mean $\boldsymbol{\mu}_0$, and the covariance matrix is estimated from ν_0 observations with sample mean $\boldsymbol{\mu}_0$ and with sum of pairwise deviation products $\boldsymbol{\Psi}_0$ (Note that observations here refer to pseudo observations). In this case, random measure $H = NIW(\mu_0, \kappa_0, \nu_0, \Psi_0)$. We derive the prior and posterior of cluster parameters recursively. We assume that the prior for cluster in epoch t_h is $H_d = \{\mu_{t_h}, \kappa_{t_h}, \nu_{t_h}, \Psi_{t_h}\}$. For the root clusters, the prior is the random measure H.

Suppose $\{\phi_{t_b,k_b}, \cdots, \phi_{t_h,k_h}\}$ is the path from the root cluster to the target cluster in the evolutionary tree. Given the observations $\mathbf{X} = [\mathbf{x}_{t_b}, \ldots, \mathbf{x}_{t_h}]$, the posterior of the cluster is $p(\phi_{t_h,k}|\mathbf{x}_{t_b}, \cdots, \mathbf{x}_{t_h-1}, H, \lambda) = NIW(\mu_{t_h}^{'}, \kappa_{t_h}^{'}, \nu_{t_h}^{'}, \Psi_{t_h}^{'})$, where

$$\begin{aligned}
\mu_{t_h}^{'} =& \frac{\kappa_{t_h}\boldsymbol{\mu}_{t_h} + n\bar{\mathbf{x}}}{\kappa_{t_h} + n}, \\
\kappa_{t_h}^{'} =& \kappa_{t_h} + n, \\
\nu_{t_h}^{'} =& \nu_{t_h} + n, \\
\Psi_{t_h}^{'} =& \boldsymbol{\Psi}_{t_h} + \mathbf{C} + \frac{\kappa_{t_h} n}{\kappa_0 + n}(\bar{\mathbf{x}} - \boldsymbol{\mu}_{t_h})(\bar{\mathbf{x}} - \boldsymbol{\mu}_{t_h})^T,
\end{aligned} \tag{11}$$

where $\bar{\mathbf{x}}_{\mathbf{t_h}}$ is the mean of observations, $\mathbf{C} = \sum_{i=1}^{n}(\mathbf{x}_{\mathbf{t_h},\mathbf{i}} - \bar{\mathbf{x}}_{\mathbf{t_h}})(\mathbf{x}_{\mathbf{t_h},\mathbf{i}} - \bar{\mathbf{x}}_{\mathbf{t_h}})^T$. The posterior parameter in next epoch can be again served as the pseudo observations for the variant clusters with decaying rate η. Then, for inheritor clusters of $\phi_{t_h,k}$, their prior distribution is $p(\phi_{t_h,k}|\mathbf{x}_{t_b}, \cdots, \mathbf{x}_{t_h-1}, H, \lambda) = NIW(\mu_{t_h+1}, \kappa_{t_h+1}, \nu_{t_h+1}, \Psi_{t_h+1})$, where

$$\begin{aligned}
\mu_{t_h+1} =& \frac{\lambda\kappa_{t_h}^{'}\boldsymbol{\mu}_{t_h}^{'} + \lambda n\bar{\mathbf{x}}}{\lambda\kappa_{t_h}^{'} + \lambda n} = \frac{\kappa_{t_h}^{'}\boldsymbol{\mu}_{t_h}^{'} + n\bar{\mathbf{x}}}{\kappa_{t_h}^{'} + n}, \\
\kappa_{t_h+1} =& \lambda(\kappa_{t_h}^{'} + n), \\
\nu_{t_h+1} =& (\nu_{t_h}^{'} + n), \\
\Psi_{t_h+1} =& \lambda\Big(\boldsymbol{\Psi} + \mathbf{C} + \frac{\kappa_{t_h}^{'} n}{\kappa_{t_h}^{'} + n}(\bar{\mathbf{x}} - \boldsymbol{\mu}_{t_h}^{'})(\bar{\mathbf{x}} - \boldsymbol{\mu}_{t_h}^{'})^T\Big).
\end{aligned} \tag{12}$$

It is obvious that all the parameters of a cluster can be recursively inferred from the root of an evolutionary tree.

For discrete data with multinomial distribution. The conjugate prior for multinomial distribution is Dirichlet distribution. Here we take text collections for example. For topics (clusters) in text collections, the parameter is the word distribution. The prior and posterior of the topics take the form of Dirichlet distribution. The parameter of Dirichlet corresponds to the count of words. The prior for orphan topics is set to $H = (H_0, \cdots, H_V)$, where V is the size of the vocabulary. For a inheritor topic in evolutionary tree, its prior is set as the posterior of its parent with a decaying factor λ. So the prior for topic $\phi_{t_h,k}$, i.e., $H_{\phi_{t-1,k}}$, is written as in Eq.(13),

$$p(\phi_{t_h,k}|\mathbf{x}_{t_b}, \cdots, \mathbf{x}_{t_h-1}, H, \lambda) = \mathbf{Dir}(\lambda^{t_h-t_b}H + \sum_{t=t_b}^{t_h-1}\lambda^{t_h-t}\mathbf{x_t}) \tag{13}$$

where $\{\phi_{t_b,k_b}, \cdots, \phi_{t_h,k_h}\}$ is the path from the root topic to the target topic in the evolutionary tree and $\{\mathbf{x}_{t_b}, \cdots, \mathbf{x}_{t_h}\}$ are the corresponding observations. The posterior of $\phi_{t_h,k}$ can be easily inferred by adding observations to its prior,

as in Eq.(14),

$$p(\phi_{t_h,k}|\mathbf{x}_{t_b},\cdots,\mathbf{x}_{t_h-1},\mathbf{x}_{t_h},H,\lambda)=\mathbf{Dir}(\lambda^{t_h-t_b}H+\sum_{t=t_b}^{t_h}\lambda^{t_h-t}\mathbf{x_t}) \tag{14}$$

Note that this parameter evolutionary scheme can *also* be applied to the Evolving Hierarchical Dirichlet Processes in Section 5.

4.4 Model Inference

We employ a Markov Chain Monte Carlo (MCMC) method to estimate the posterior distribution of cluster states by delivering a Gibbs sampler [31]. The states of the sampler contain both the cluster indicator for every data record $\{\theta_{t,i}\}$ and the posterior of cluster parameters $\{\phi_{t,k}\}$. In the inference process, we iteratively sample their states until convergence. Note that $\{\phi_{t,k}\}$ can be calculated as described in Section 4.3. So in this part, we focus on sampling $\{\theta_{t,i}\}$.

Sampling cluster indicator $\theta_{t,i}$. For a data record $x_{t,i}$, conditioned on parameter of clusters and cluster assignments for other data, its cluster indicator $\theta_{t,i}$ is sampled according to its conditional distribution, which can be written as in Eq.(15),

$$p(\theta_{t,i}=k|\theta_{t-1},\theta_{t,-i},x_{t,i},\{\phi_k\}_{t,t-1},G_0,\tau,\gamma,\eta)\propto \begin{cases}\frac{\eta n_{t-1,p} n_{t,k}^{(i)}}{\tau+\sum_{s\in S_{t-1,p}} n_{t,k}^{(i)}}F(x_{t,i}|\phi_{k_{f_{t,i}}}) & \text{if } k\in S_{t-1,p}\\ \frac{\eta n_{t-1,p}\tau}{\tau+\sum_{s\in S_{t-1,p}} n_{t,k}^{(i)}}\int F(x_{t,i}|\phi)d\ p(\phi|\phi_{t-1,p}) & \text{if } k=S_{t-1,p}^{+}\\ n_{t,k}^{(i)}F(x_{t,i}|\phi_{t,o}) & \text{if } k\in O_t\\ \gamma\int F(x_{t,i}|\phi)dG_0(\phi) & \text{if } k=O_t^{+}\end{cases} \tag{15}$$

where $\theta_{t,-i}$ denotes the set of $\theta_{t,\cdot}$ without $\theta_{t,i}$, and $S_{t-1,p}^{+}$ denotes new variant of $\phi_{t-1,p}$.

Time complexity. In the Gibbs sampling procedure for each record, we need to determine whether it belongs to existing clusters, or a new branching cluster of parent clusters, or a brand new clusters. So the computation cost of one iteration of Gibbs sampling for data in one epoch is $O((K_{t-1}+K_t+1)N_t)$, where N_t is the number of records in epoch t, K_{t-1} is the number of clusters in the previous epoch, and K_t is the number of existing clusters. Generally, the expectation of number of topics is proportional to $log(N)$ [38]. So the time complexity of inference procedure is $O(log(N_t)N_t)$. As the number of topics is determined by the data and hyperparameters, so it is only a rough estimation.

5 Evolving Hierarchical Dirichlet Processes

The EDP can model cluster evolutionary traces of *non-hierarchical* temporal data, where each data point entirely belongs to a cluster. However, in real-world applications, a large proportion of data are organized hierarchically. Take text collections as an example, each document contains a group of words, and each word can be assigned to a cluster/topic. So a document is treated as a mixture of topics. To model hierarchical temporal data, we propose the Evolving Hierarchical Dirichlet Processes (EHDP). In this paper we illustrate EHDP *w.r.t.* temporal text collections. However, EHDP can handle variant types of hierarchical data similarly.

As a generative model, the document collections are generated hierarchically. Given a global level measure G_t in epoch t, For a time-varying global level measure, G_t is distributed conditioned on G_{t-1} and H, which is similar with EDP. The document level measure G_{td} is generated with a Dirichlet process from G_t. In summary, EHDP can be formally defined as follows (the graph model of EHDP is given in Figure 5),

$$\begin{aligned} G_t|\tau, G_{t-1}, \gamma, H \sim & \sum_{k\in G_{t-1}} \beta_{k,t-1} DP(\tau, H_{\phi_{t-1,k}}) + \beta_{tO} DP(\gamma, H), \\ G_{dt}|\alpha, G_t \sim & DP(\alpha, G_t) \\ \theta_{dit}|G_{dt} \sim & G_{dt}, \\ w_{dit}|\theta_{dit} \sim & F(\theta_{dit}) \end{aligned} \tag{16}$$

5.1 Construct EHDP with CRP

We focus on deriving the conditional distribution for a word's table/topic selection θ_{tdi} and the conditional distribution of a table's topic selection $\psi_{tdb^{new}}$.

For the global random measure G_t, its distribution is the same as G_t in EDP, which can be expressed as in Eq.(7). Given G_t, the conditional distribution for a word's table selection θ_{tdi} can be computed by Eq.(17),

$$\theta_{tdi}|\theta_{td-i}, \alpha, G_t \sim \sum_{b=1}^{m_{td\cdot}} \frac{n_{td\cdot}}{i-1+\alpha}\delta_{\psi_{tdb}} + \frac{\alpha}{i-1+\alpha} G_t \tag{17}$$

If a term in the first part is chosen, we set $\theta_{dit} = \psi_{dbt}$. If the second term is chosen, we increase $m_{d\cdot t}$ by one, draw a dish for the new table b^{new} as $\psi_{tdb^{new}t} \sim G_t$ and set $\theta_{dit} = \psi_{db^{new}t}$.

The conditional distribution of a table's topic selection $\psi_{tdb^{new}}$ can be computed with G_t integrated out. As G_t is composed of a two-level hierarchical

structure. In the first level, $\psi_{tdb^{new}}$ chooses a DP as follows.

$$\begin{aligned}\psi_{tdb^{new}}|\psi_{t-1:t\cdot\cdot},\gamma,\tau,\eta \propto (\sum_{o'\in O_t} m_{t\cdot o'}+\gamma)\times DP(\gamma,H)+\\ \sum_{k\in K_{t-1}}(\eta m_{t-1\cdot k}+\sum_{s'\in S_{t-1,k}} m_{t\cdot s'})\times DP(\tau,H_{\phi_{t-1,k}})\end{aligned} \tag{18}$$

where O_t is the set of brand new topics, and $S_{t-1,k}$ is the set of inheritors of topic $\phi_{k,t-1}$. $DP(\gamma,H)$ and $DP(\tau,H_{\phi_{t-1,k}})$ can be constructed as in Eq.(8) and Eq.(9). Compared with EDP, Eq.(8) and Eq.(9) in EHDP, the number of data records with cluster k, i.e., n_{tk}, is replaced by the the number tables with cluster k, i.e., $m_{t\cdot k}$.

At the second level, $\psi_{tdb^{new}}$ chooses a specific topic within the picked DP. Combining Eq.(8) and Eq.(9), Eq.(18) can be rewritten as follows,

$$\begin{aligned}&\psi_{tdb^{new}}|\psi_{t-1:t},\gamma,\tau,\eta \propto\\ &\sum_{k\in K_{t-1}}(\eta m_{t-1,\cdot k}+\sum_{s'\in S_{t-1,k}} m_{t\cdot s'})\sum_{s'\in S_{t-1,k}}\frac{m_{t\cdot s'}}{\sum_{s\in S_{t-1,k}} m_{t\cdot s'}+\tau}\delta_{\phi_{ts'}}\\ &+\sum_{k\in K_{t-1}}(\eta m_{t-1,\cdot k}+\sum_{k\in S_{t-1,k}} m_{t\cdot s'})\frac{\tau}{\sum_{s'\in S_{t-1,k}} m_{t\cdot s'}+\tau}H_{\phi_{t-1,k}}\\ &+\sum_{o'\in O_t} m_{t\cdot o'}\delta_{\phi_{to'}}+\gamma H\end{aligned} \tag{19}$$

Eq.(17) and Eq.(19) complete the construction procedure of the EHDP.

5.2 Gibbs Sampling for Model Inference

In this section, we introduce the inference algorithm for EHDP based on Gibbs sampling[31]. Variables of interest are the topic indicator θ_{tdi} for each word w_{tdi}, and the topic-word distribution for a topic ϕ_{tk}. The learning process iterates the following two steps until convergence:

- Step I, given the parameter of topics, sample the topic indicator for each word;
- Step II, given the topic indicator for each word, calculate posterior parameter of topics.

The second step can be performed as described in Section 4.3. In this section we focus on sampling θ_{tdi}.

We can design a Gibbs sampler by following the generation procedure of EHDP as described in Eq.(17) and Eq.(19). Teh et al. in [38] pointed out that the straightforward sampling scheme converges slowly, because changing the topic of a table will change the membership of multiple words at the same time, which potentially lowers the possibility for starting a new topic. To overcome

this problem, we instantiate G_t with β_t, so we can directly assign a topic to a word.

Gibbs sampler can be designed as follows. We add a superscript $-i$ to a variable to indicate the quantity without the contribution of object i.

Sampling β_t. Based on Eq.(18), the weight of DPs in the first level partition can be sampled as follows,

$$\begin{aligned}\beta_t &= (\overbrace{\beta_{t-1,k},\cdots}^{k\in K_{t-1}},\beta_{tO}) \\ &= \mathbf{Dir}(\overbrace{\eta m_{t-1\cdot k}+\sum_{s'\in S_{t-1,k}} m_{t.s'}^{-tdb},\cdots}^{k\in K_{t-1}},\sum_{o'\in O_t} m_{t.o'}^{-tdb}+\gamma),\end{aligned} \tag{20}$$

Based on Eq.(8) and Eq.(9), $\beta_{t-1,k}$ and β_{to} can be sampled as.

$$\begin{aligned}\beta_{t-1,k}(\overbrace{\beta_{ts'},\cdots}^{s'\in S_{t-1,k}},\beta_{S^+_{t-1,k}}) &= \beta_{t-1,k}\mathbf{Dir}(\overbrace{m_{t.s'}^{-tdb},\cdots}^{s'\in S_{t-1,k}},\tau) \\ \beta_{tO}(\overbrace{\beta_{to'},\cdots}^{o'\in O_t},\beta_{K_t^+}) &= \beta_{tO}\mathbf{Dir}(\overbrace{m_{t.o'}^{-tdb},\cdots}^{o'\in O_t},\gamma),\end{aligned} \tag{21}$$

Sampling a topic θ_{tdi}. This can be achieved by computing the likelihood under $\theta_{tdi}=k^{'}$. With an instantiated G_t, the calculation can be realized by grouping together terms associated with topic $k^{'}$ in Eq.(17) and Eq.(19):

$$p(\theta_{tdi}=k^{'}|\mathbf{m}_{t-1:t},\mathbf{k}_t^{-tdi},\beta_t)\propto \begin{cases}(n_{tdk'}^{-tdi}+\alpha\beta_{t-1,k}\beta_{t,k'})f_{\phi_{tk'}}(w_{tdi}) & \text{if } k\in S_{t-1,p}\\ \alpha\beta_{t-1,k}\beta_{S^+_{t-1,k}}\int f(w_{tdi}|\phi_{t,k})dH_{\phi_{t-1,k}}(\phi_{t,k'}) & \text{if } k=S^+_{t-1,p}\\ (n_{tdk}^{-tdi}+\alpha\beta_{tO}\beta_{t,k'})f_{\phi_{tk^+}}(w_{tdi}) & \text{if } k\in O_t\\ \alpha\beta_{tO}\beta_{K_t^+}\int f(w_{tdi}|\phi_{t,k'})dH(\phi_{t,k'}) & \text{if } k=O_t^+\end{cases} \tag{22}$$

where

$$\begin{aligned}\int f(w_{tdi}|\phi_{t,k'})dH_{\phi_{t-1,k}}(\phi_{t,k'}) &= p(w_{tdi}|H_{\phi_{t-1,k}}) \\ &= \frac{p(w_{tdi},\widehat{\phi_{t,k}}|H_{\phi_{t-1,k}})}{p(\widehat{\phi_{t,k}}|w_{tdi},H_{\phi_{t-1,k}})}\end{aligned} \tag{23}$$

As $p(w_{tdi},\widehat{\phi_{t,k}}|H_{\phi_{t-1,k}})$ and $p(\widehat{\phi_{t,k}}|w_{tdi},H_{\phi_{t-1,k}})$ can be directly calculated, so we can choose a particular $\widehat{\phi_{t,k}}$ to calculate the integral. And we can get $\int f(w_{tdi}|\phi_{t,k'})dH(\phi_{t,k'})$ with exactly the same way.

Algorithm 1 The online learning framework for EHDP.

Input: Hyperparameters $\{\alpha, \gamma, \tau, \eta, \lambda, H\}$;

$t \longleftarrow 0$, the evolutionary trace is $NULL$.
while a text stream is not end **do**
  $t \leftarrow t+1$; Get text collections $\mathbf{W_t}$;
  while Not convergence **do**
    For each w_{tdi}, sampling the topic assignment θ_{tdi} by using Eq.(22).
    For each topic ψ_{tdk}, sampling m_{tdk} by using Eq.(24).
    Sampling the β_t by using Eq.(20)and Eq.(21).
    For each topic ϕ_{tk}, inferring its parameter by using Eq.(13)and Eq.(14).
  end while
  Update the evolutionary tree based on the sampling result.
end while

Sampling m_{tdk}. In this sampling scheme, the only effect of table-topic and word-table assignments on other variables is via $\mathbf{m}_t$ in the conditional distribution of β_t as in Eq.(20)and Eq.(21). As a result, it is sufficient to sample $\mathbf{m}_t$ conditioned on word assignments. To obtain the distribution of m_{tdk}, we suppose a Dirichlet process concerning only about n_{tdk} words. According to the work of Antoniak et al. [5] and Teh et al. [38], we can obtain,

$$p(m_{tdk} = m|\mathbf{k}_{td.}, \beta) = \frac{\Gamma(\alpha\beta_{tk})}{\Gamma(\alpha\beta_{tk} + n_{tdk})} s(n_{tdk}, m)(\alpha\beta_{tk})^m \tag{24}$$

where $s(n, m)$ is an unsigned Stirling numbers of the first kind.

Time complexity. In the Gibbs sampling procedure of EHDP. For each word, we need to determine whether it belongs to existing clusters, or a new branching cluster of parent clusters, or a brand new clusters. And for each document, there are two sampling procedures. First, sample the weight of DPs with Eq.(20, 21). Second, after sampling the topic of each words, we needs to sample m_{tdk} for each topic. So the computation cost of one iteration of Gibbs sampling for data in one epoch is $O(K_{t-1}N_t + (K_{t-1} + K_t + 1)N_tL) \approx O((K_{t-1} + K_t + 1)N_tL)$, where N_t is the number of documents in epoch t, L is the average number of words in each document, K_{t-1} is the number of clusters in the previous epoch, and K_t is the number of existing clusters. Here K_t and K_{t-1} is determined by the data. Based on the similar analysis in Section 4.4, the time complexity of sampling procedure of EHDP can be roughly estimated with $O(log(N_tL)N_tL)$.

5.3 An Online Learning Framework.

It is widely admitted that model learning from a large number of documents is computationally demanding. EHDP can be incrementally updated and meet the need of online applications.

In the above Gibbs sampling algorithm, the samplers solely depend on the model states in $t-1$ and t. To satisfy online learning, we omit the information

propagated from epoch $t+1$ [1]. As in the online setting, at epoch t, the data in $t+1$ has not arrived. For $\mathbf{W_{t+1}}$, we can update the model incrementally by branching new topics from leaves or appending new root topics, instead of relearning the model from scratch. The incremental learning framework is presented in Algorithm 1.

Although the number of online text collections can be infinitely large, for each epoch t, we only need to handle the newly arrived data and set the prior based on the previous model state. So this framework is efficient for online applications. For sampling procedure within one epoch, the Metropolis-Hasting-walker [28] can be adopted to accelerate the sampling procedure.

5.4 Discussion

We summarize the merits of EDP and EHDP.

- First, EDP and EHDP are nonparametric Bayesian models, which can learn a proper number of topics in each epoch. In real-world temporal text collections, there are lots of long tail topics and the number of topics varies a lot. So the proposed models are well suited for analyzing topics in real-world data sets.
- Second, EDP and EHDP adopts Dirichlet Processes to model topic branching processes over time, which leads to forest-like topics evolution over time. So existing nonparametric Bayesian models, such as Recurrent Chinese Restaurant Processes can be treated as a special case of EDP and EHDP where all the topics have only one branch. We can also control the scale of topic branches by adjust the concentration factor τ, but still determine the exact number of branches based on data. Compared with temporal Bayesian models such as DP-NetClus and Incremental Gibbs Sampler, our method can discover comprehensible cluster/topic evolutionary trace.
- Third, we adopt the conjugate prior cascade to model the evolution of clusters parameters in EDP and EHDP. So the parameter of topics and the distribution of topics are smoothed over time, which alleviates the possible over-fitting, especially in epochs with rare collections. The scheme also leads to easy posterior inference for the parameter of clusters.
- Forth, we present a online learning framework for EDP and EHDP, which is a desirable feature for handling real-world data sets, especially for online applications.

6 Experiments

In this section, we test EDP and EHDP *w.r.t.* its capability of recovering evolutionary traces from both synthetic and real-world data sets. We conduct the following tests:

- We compare EDP with traditional parametric clustering methods on synthetic data sets, which demonstrates the superiority of EDP.

– We compare EHDP with the state-of-the-art temporal topic models, which shows the superiority of EHDP.
– We use EHDP to analyze the NIPS data set and show that EHDP can recover interesting patterns of topics.
– We explore the sensitivity of the hyperparameters in EHDP.

6.1 Experimental Settings

Measures. We use the metric of *perplexity* for evaluation. Perplexity is widely used in evaluating topic models [12, 24, 44, 41]. We compute perplexity as in [24]. 10% of each document is randomly sampled to create a test set while the remaining is used for training. The training set is used for learning model parameters while the test set is held out to compute the perplexity. Formally, for the text collections, the perplexity is defined as,

$$perplexity(D_{test}) = exp\{-\frac{\sum_{d \in D_{test}} \log p(\mathbf{w}_d)}{\sum_{d \in D_{test}} |\mathbf{w}_d|}\} \tag{25}$$

Benchmarks. We compare the EDP for multivariate Gaussian distribution with the following clustering methods:

– **Kmeans**: Kmeans is an basic clustering method, which requires setting the number of clusters manually.
– **DBscan**: DBscan is a density-based clustering algorithm, which does not require one to specify the number of clusters in the data a priori.

We also compare the EDP and EHDP for Dirichlet distribution with the following state-of-the-art topic models:

– **Dynamic Topic Model (DTM) [11]**: DTM is an extension of LDA for modeling temporal data;
– **Topic over Time model (ToT)[44]**: ToT is a parametric temporal topic model which can discover both global and local topics over time;
– **HDP [38]**: HDP is a hierarchical non-parametric topic model for modeling static data sets;
– **Recurrent CRF (RCRF)[4]**: RCRF is an extension of HDP for temporal data, where the topics over time are organized as linear chains;
– **DP-NetClus[36]**: DP-NetClus is a non-parametric topic model which can infer connection between topics over time.
– **Incremental Gibbs Sampler (IGS)[19]**: IGS infers the connection of topics with the Gibbs sampler.

6.2 Data sets.

Real-world data sets. We conduct experiments on four public data sets which are widely used as benchmark. All data sets can be downloaded online. Table 1 summarizes the statistics of the data sets.

Table 1 Statistics of Data sets

Data sets	# docs	# terms	# time span
NIPS[1]	1740	2000	13
DBLP[2]	20480	4000	10
NSF-Awards[2]	30287	4000	14
Douban comments[3]	120864	8000	60

Table 2 Average Perplexity on Data sets

	EHDP	EDP	DTM	HDP	RCRF	ToT	DP-NetClus	IGS
NIPS	$177.37_{\pm 1.94}$	$191.30_{\pm 3.86}$	$186.97_{\pm 1.95}$	$189.62_{\pm 1.86}$	$182.50_{\pm 1.86}$	$184.41_{\pm 1.90}$	$180.01_{\pm 1.16}$	$185.05_{\pm 1.08}$
DBLP	$335.85_{\pm 2.88}$	$346.04_{\pm 1.89}$	$352.97_{\pm 2.61}$	$359.48_{\pm 2.24}$	$343.51_{\pm 2.90}$	$347.12_{\pm 2.43}$	$340.59_{\pm 1.90}$	$349.48_{\pm 1.35}$
NSF	$318.08_{\pm 2.11}$	$331.42_{\pm 1.58}$	$335.10_{\pm 1.43}$	$340.46_{\pm 1.82}$	$325.39_{\pm 1.97}$	$329.07_{\pm 1.62}$	$323.67_{\pm 2.78}$	$331.54_{\pm 2.11}$
Douban	$556.88_{\pm 2.73}$	$583.52_{\pm 2.37}$	$581.93_{\pm 1.94}$	$587.99_{\pm 3.82}$	$565.53_{\pm 2.46}$	$572.55_{\pm 2.23}$	$564.74_{\pm 2.93}$	$572.51_{\pm 2.44}$

Synthetic data sets. Testing whether our method can *correctly* recover the evolutionary trace of temporal data is difficult because usually the ground true is missing. So we manually design a data set with known evolutionary trace. We adopted 2-D Gaussian to generate the data, as shown in Figure 6.

6.3 Clustering Synthetic Data Sets

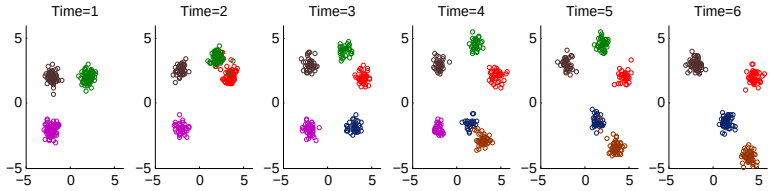
In this part, we test whether EDP can properly recover the evolutionary trace of the synthetic dataset and compare the results with that of clustering methods such as K-means and DBSCAN [46]. We use EDP for modeling continuous Gaussian data. We set EDP parameters to be ($\gamma = 0.05, \alpha = 0.1, \eta = 0.5, \lambda = 0.5$), where H is set to a normal-inverse-Wishart distribution as $NIW(\mu = [0, 0], \kappa_0 = 10, \nu_0 = 10, \Psi = [10, 0; 0, 10])$. We run the Gibbs sampler for 100 iterations. For K-means, we set the number of clusters $K = 5$. And for DBSCAN, we set number of objects in a neighborhood of an object $minPts = 4$ and the neighborhood radius $\epsilon = 0.5$.

The clustering result is shown in Figure 6.(a). Compared with the results given by k-means (as depicted in Figure 6.(b)), EDP can learn proper cluster number for each epoch. Although DBSCAN also does not require setting the number of clusters (as depicted in Figure 6.(c)), DBSCAN cannot handle temporal data sets and cannot modeling the evolutionary trace of clusters. What's more, DBSCAN tend to find lots of noise data records which are not actually noise. The corresponding evolutionary trace recovered by EDP is plotted in Figure 6.(d). The results demonstrate that EDP not only recovers the birth and death of clusters, but also detects the branches for clusters.

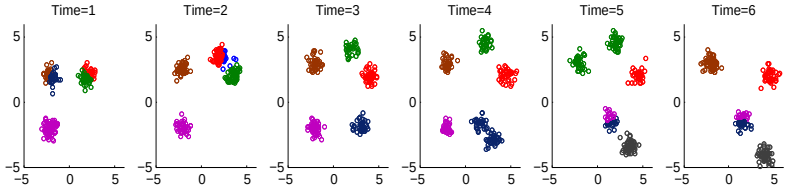
[1] `http://www.cs.nyu.edu/~roweis/data.html`

[2] `http://www.cs.uiuc.edu/~hbdeng/data/kdd2011.htm`

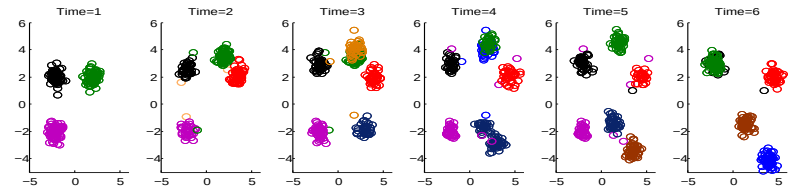
[3] `http://movie.douban.com/`



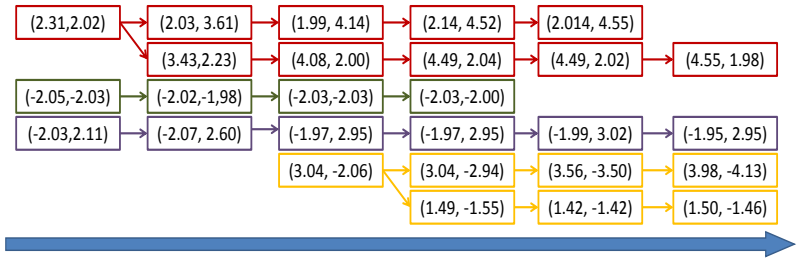
(a) Clustering result by EDP



(b) Clustering result by K-means.



(c) Clustering result by DBSCAN.



(d) Evolutionary tree recovered by EDP

Fig. 6 Test results on synthetic data sets. (a) The clustering result given by EDP. It shows that EDP can learn proper number of clusters. (b) The clustering result given by K-means, with $K = 5$. We can observer that the records are not clustered correctly. (c) The clustering result given by DBScan. We can observe that some some records are treated as outliers and some clusters are divided into multiple clusters mistakenly. (d) the evolutionary trace recovered by EDP; the rectangles represent clusters, the value in rectangle represents mean of Gaussian. We can observe that EDP can recover the tree-like evolutionary trace of clusters.

6.4 Comparison results.

We compare EDP and EHDP with benchmarks for in task of mining the topic evolution *w.r.t.* perplexity. We set the EDP parameters to be ($\gamma = 0.05$, $\alpha = 0.1$, $\eta = 0.5$, $\lambda = 0.5$), and H is set to a Dirichlet prior with all the

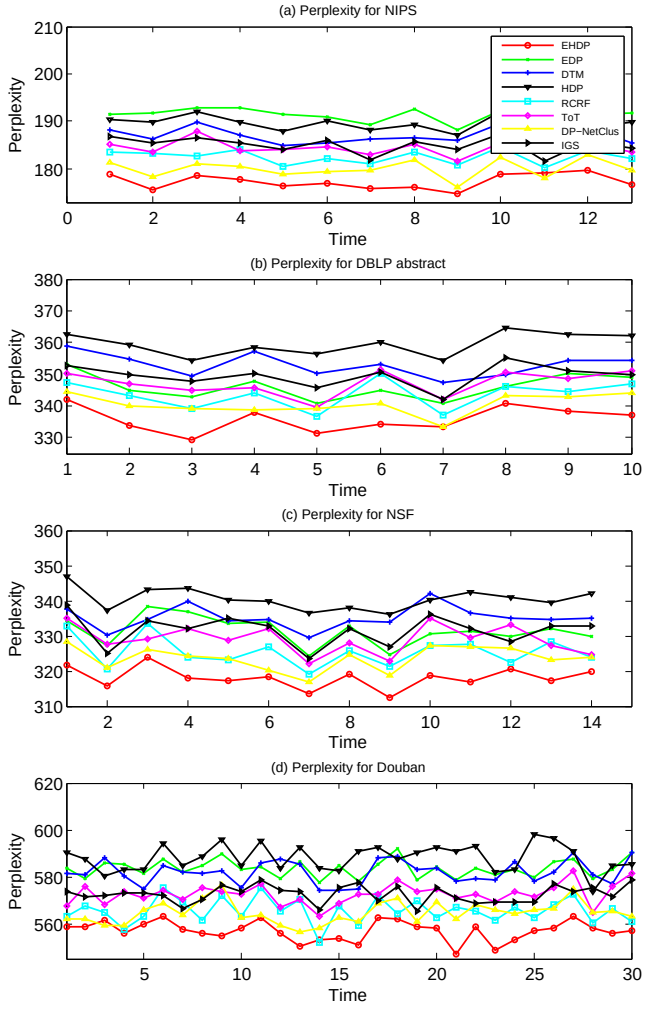


Fig. 7 Model comparison *w.r.t.* perplexity. (a) NIPS data set; (b) DBLP data set; (c) NSF data set; (d) Douban data set. We can observe that on average EHDP outperforms others.

entries equal to 0.1. We set the EHDP parameters the same as those in EDP, except the extra hyper-parameters $\tau = 0.1$. For the hyper-parameters of the benchmarks, we mainly adopts the settings in the original papers without parameter selection procedure. Specially, for DTM, for simplicity, we fixed the number of topics $K = 20$. And for ToT, we fixed the number of topics $K = 50$, which is larger than that of DTM because ToT would recover both local and global topics. All the other methods are nonparametric, so we do not need to set the number of topics. For all the data sets, the sampling procedure of one iteration can be finished within 10 minutes in EDP and EHDP.

Figure 7 and Table 2 shows the results on difference data sets. We can observe that on average EHDP outperforms others. The reasons are five folders. First, compared with the parametric Bayesian models (e.g., DTM and ToT), EHDP can learn a proper number of topics in each epoch. As there are lots of long tail topics in the real-world text collections, it is almost impossible to assign a proper number of topics in DTM and ToT. So EHDP can achieve a better perplexity. Second, compared with static nonparametric topic models (e.g., HDP), EHDP can capture temporal topic features over time. For the real-world data sets, the word distribution of topics varies a lot, and topics emerge and die constantly. So catching the temporal features will lead to better performance in perplexity. Third, compared with nonparametric temporal topic models which can recover topic chains (e.g., RCRF), evolutionary traces recovered by EHDP better matches the actual pattern in text collections. All the topics in EHDP model are assigned with a proper prior, because a topic are allowed to branch into several topics. While for RCRF, each topic is forced to have at most one branch, so other branches are either treated as an brand new topic, or mistakenly merge with other topics. As a result, the topic parameters can be properly smoothed along the tree, which leads to better performance in perplexity. Fourth, compared with nonparametric topic models which can recover complex evolutionary traces (e.g., DP-NetClus and IGS), EHDP can control the propensity of creating branches. So EHDP can avoid overfitting the training data, and avoid generating topics links which are false-positive. Fifth, the performance of EDP is not good, as it is not a hierarchical model, the entire document is assigned to one topic. In conclusion, EHDP tends to recover proper sparse branches in all data sets, which better performances of modeling topics evolution in temporal collections.

6.5 Case Study on the NIPS Dataset.

We use EHDP to analyze the NIPS collection. In the test, we run the Gibbs sampler for 2,000 iterations. The results are shown in Figure 8. As the entire tree of topics is too large, we only plot partial results in Figure 8. (a). From the results, we can observe that EHDP can spot birth, branch and death of topics over time. For example, EHDP can spot topics after year 1987, the topics of *image recognition* branched into two topics: *feature extraction* and *MRF*. Compared with the topic trace recovered by RCRF where topics are organized as a chain, as depicted in Figure 8. (b), EHDP can recover the interesting connections of topics over time.

Based on EHDP, we can also unveil some interesting knowledge about NIPS. Figure 9 (a) shows the number of topics in NIPS over time. We can observe that most topics are inheriting topics and brand new topics are rarely generated. The results also show the number of death topics over time. We can see that the birth rate of topic is higher than the death rate. Note that the death of a branch does not mean the death of the whole tree. Figure 9 (a) affirms our intuition that NIPS originally focused on the topic of *neural net-*

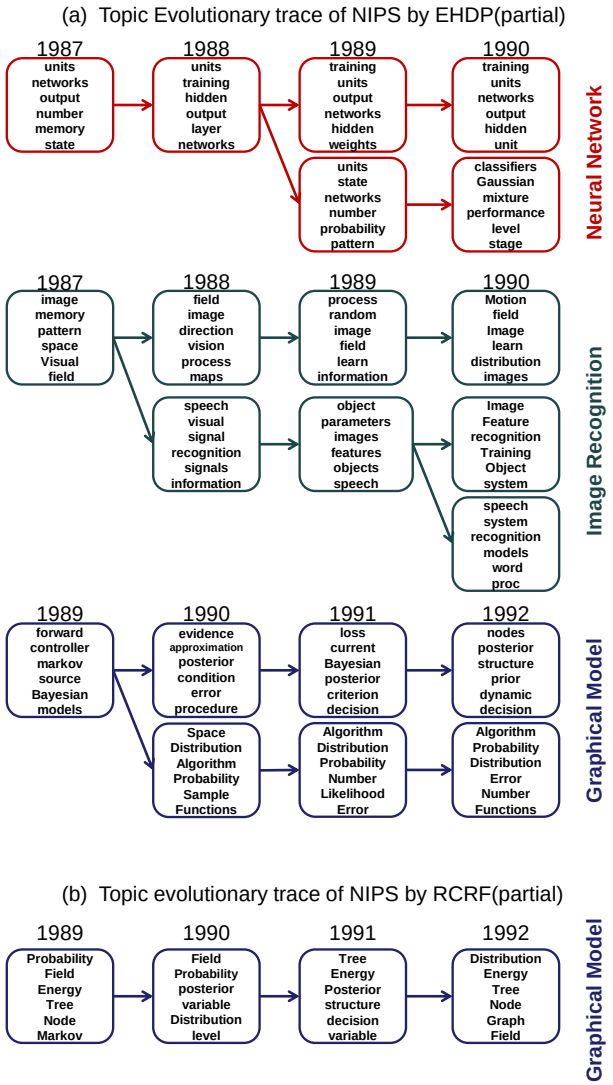


Fig. 8 (a) the recovered topic trace with EHDP (partial), where topics are represented by key words; (b) the recovered topic trace with RCRP. We can observe that EHDP can recover the tree-like evolutionary trace of topics, which provides more informative.

works and then gradually branched into multiple topics and many new topics were merged into NIPS from other fields. Topics both evolved and connected over time.

Figure 9 (b) shows the average KL distance of the word distribution between parents and children. The results above the average indicate that the topics switch sharply. Figure 9 (c) shows the KL distance between the distribution of topics in adjacent epoch. The inheritors of a topic are treated as a

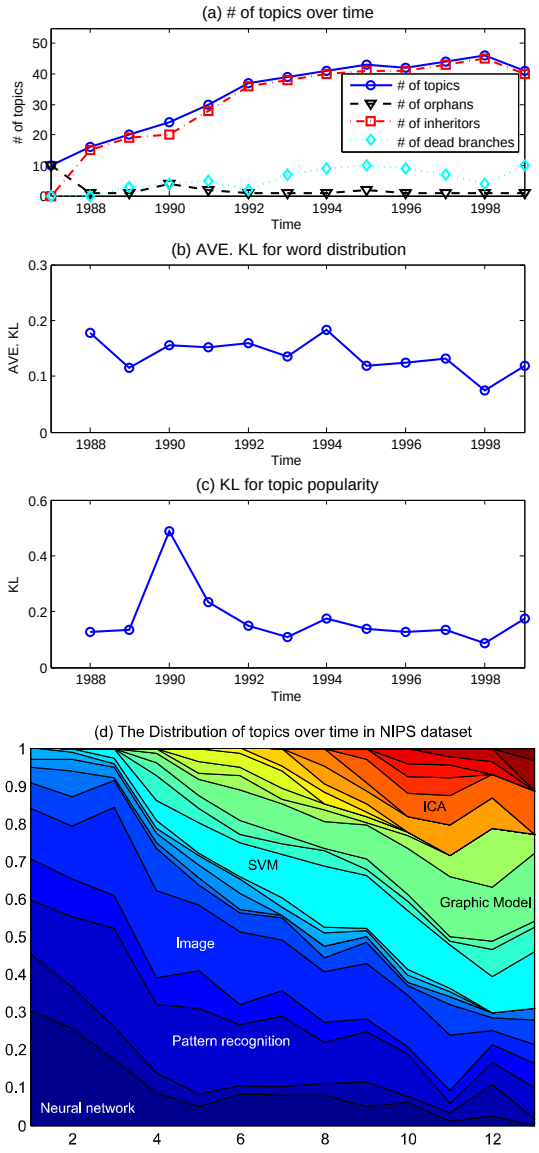


Fig. 9 The results on the NIPS collection. (a) the number of topics over time, (b) the average KL distance for word distributions between parent and children, (c) the KL distance for topic distribution, (d) the evolution of topic distribution over time.

whole. We can observe that in the epoch where KL value is high above the average, the number of orphan topics changes. Figure 9 (d) shows the evolution of topic distribution. From the results, we can spot the hot topics in each

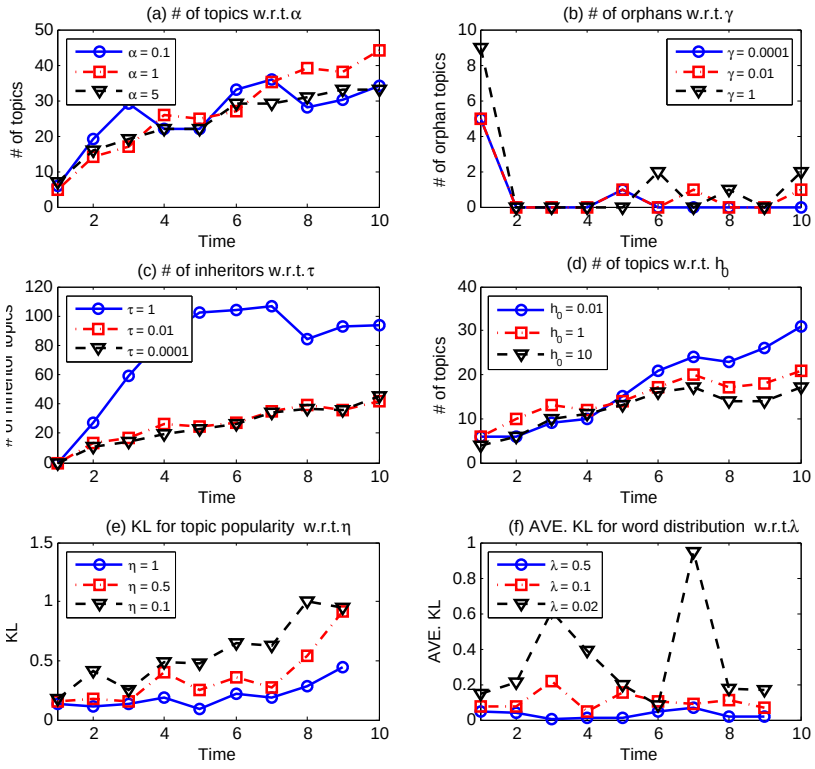


Fig. 10 Performance *w.r.t.* different hyperparameters, (a) # of recovered topics *w.r.t.* α; (b) # of recovered one-child topics *w.r.t.* γ, (c) # of recovered children topics *w.r.t.* τ, (d) # of recovered topics *w.r.t.* measure $H = \mathbf{Dir}(h_0, \cdots)$ with prior h_0, (e) the KL distance for topic distribution *w.r.t.* η, and (f) the KL distance for word distribution *w.r.t.* λ.

epoch and the evolutionary trend over time. Combining these results, we can find important time periods for the development of NIPS.

6.6 Sensitivity of Hyperparameters

To assess the sensitivity of EDP and EHDP, we conduct experiments by holding all hyperparameters fixed with default values and only varying one of them. As EHDP is an extension of EDP, we only access the sensitivity of hyperparameters of EHDP. The hyperparameters are $\{\alpha, \gamma, \tau, \eta, \lambda, H\}$. The test results are depicted in Figure 10. We briefly analyze the result as follows.

– **Concentration factors**(α, γ, τ)**.** For (α, γ, τ), we surprisingly found that they can merely influence the structure of evolutionary trees recovered, as depicted in Figure 10(a)(b)(c). This is because in the inference stage, these

concentration factors can only influence the scale of topics. And the exact number is mainly determined by data itself.
- **Base measure H.** H can slightly influence a topic trace recovered, as depicted in Figure 10 (d). This is because H will influence the probability of $\int p(\mathbf{w_t}|\phi')dp(\phi'|H)$. As a result, small h_0 leads to more orphan topics.
- **Decaying factors (η, λ).** (η, λ) affects the smoothness between topics over time. The KL distances between topic distribution are shown in Figure 10 (e). All inheritors for one topic are treated as a whole. We can observe that a small η results in a larger divergence between topic distribution. Similarly, Figure 10 (f) shows that a small λ results in a bigger divergence of topic-word distribution.

To sum up, we can conclude that although there are six hyperparameters, tuning them would NOT be a difficult task. Because $(\alpha, \gamma, \tau, H)$ have limited influence on clustering results, and (η, λ) fall into the range $[0, 1]$. The parameters can be approximately estimated with model selection techniques such as cross-validation.

7 Conclusions

In this paper, we studied a new challenging problem of modeling nonlinear evolutionary traces of clusters behind temporal data, where we use temporal nonparametric Bayesian method as the solution. We briefly summarize the contributions as follows,

- We proposed a new class of Evolving Dirichlet Processes (EDP) to model nonlinear evolutionary traces of clusters, which can recover the birth, death, and branch of clusters over time.
- We proposed the Evolving Hierarchical Dirichlet Processes (EHDP) which extend EDP for modeling hierarchical data. So we can handle data with hierarchical structures, such as text collections.
- The proposed EDP and EHDP models can capture dynamic features of temporal data, such as the distribution of clusters and evolving parameters.
- We designed an online learning framework for EDP and EHDP to handle unbounded temporal data.
- We conducted experiments on both synthetic and real-world data sets. The results show that our methods can reveal interesting nonlinear evolutionary traces of clusters.

In the future, we plan to develop our model in three directions. Firstly, as the major limitations of the EDP and the EHDP are the computation cost of the inference procedure, because the Gibbs sampler generally takes many rounds to converge. So we will study how to parallelize the inference procedure, in order to apply the proposed methods to handle real-world data set with billions of records. Secondly, we are working on designing a new Bayesian models which can model both branch and merge of clusters over time. Thirdly, we will study how to find the critical data records which causes the branches of clusters, as they generally have specially meaning for the evolution of clusters.

References

1. Ahmed, A., Ho, Q., Teo, C., Eisenstein, J., Smola, A., Xing, E.: Online inference for the infinite cluster-topic model: Storylines from streaming text. In: Proceedings of the 14th Conference on Artificial Intelligence and Statistics (AISTATS) (2011)
2. Ahmed, A., Hong, L., Smola, A.: Nested chinese restaurant franchise process: Applications to user tracking and document modeling. In: Proceedings of the 30th International Conference on Machine Learning (ICML-13), pp. 1426–1434 (2013)
3. Ahmed, A., Xing, E.: Dynamic non-parametric mixture models and the recurrent chinese restaurant process: with applications to evolutionary clustering. In: SDM, pp. 219–230. SIAM (2008)
4. Ahmed, A., Xing, E.P.: Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. In: Proceedings of the 26th Uncertainty in Artificial Intelligence (UAI), UAI '10 (2010)
5. Antoniak, C.E., et al.: Mixtures of dirichlet processes with applications to bayesian nonparametric problems. The annals of statistics **2**(6), 1152–1174 (1974)
6. Banerjee, A., Basu, S.: Topic models over text streams: A study of batch and online unsupervised learning. In: SDM, vol. 7, pp. 437–442. SIAM (2007)
7. Blei, D.M., Frazier, P.I.: Distance dependent chinese restaurant processes. J. Mach. Learn. Res. **12**, 2461–2488 (2011)
8. Blei, D.M., Frazier, P.I.: Distance dependent chinese restaurant processes. The Journal of Machine Learning Research **12**, 2461–2488 (2011)
9. Blei, D.M., Griffiths, T.L., Jordan, M.I., Tenenbaum, J.B.: Hierarchical topic models and the nested chinese restaurant process. In: NIPS, vol. 16 (2003)
10. Blei, D.M., Jordan, M.I., et al.: Variational inference for dirichlet process mixtures. Bayesian analysis **1**(1), 121–143 (2006)
11. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd international conference on Machine learning, pp. 113–120. ACM (2006)
12. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. the Journal of machine Learning research **3**, 993–1022 (2003)
13. Boyles, L., Welling, M.: The time-marginalized coalescent prior for hierarchical clustering. In: Advances in Neural Information Processing Systems, pp. 2969–2977 (2012)
14. Chakrabarti, D., Kumar, R., Tomkins, A.: Evolutionary clustering. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06, pp. 554–560. ACM, New York, NY, USA (2006)
15. Chen, C., Ding, N., Buntine, W.: Dependent hierarchical normalized random measures for dynamic topic modeling. arXiv preprint arXiv:1206.4671 (2012)
16. Chi, Y., Song, X., Zhou, D., Hino, K., Tseng, B.L.: Evolutionary spectral clustering by incorporating temporal smoothness. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 153–162. ACM (2007)
17. De Smet, W., Moens, M.F.: Representations for multi-document event clustering. Data Mining and Knowledge Discovery **26**(3), 533–558 (2013). DOI 10.1007/s10618-012-0270-1. URL `http://dx.doi.org/10.1007/s10618-012-0270-1`
18. Diao, Q., Jiang, J., Zhu, F., Lim, E.P.: Finding bursty topics from microblogs. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, pp. 536–544. Association for Computational Linguistics (2012)
19. Gao, Z., Song, Y., Liu, S., Wang, H., Wei, H., Chen, Y., Cui, W.: Tracking and connecting topics via incremental hierarchical dirichlet processes. In: Data Mining (ICDM), 2011 IEEE 11th International Conference on, pp. 1056–1061. IEEE (2011)
20. Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., Rubin, D.B.: Bayesian data analysis. CRC press (2013)
21. Griffin, J.E., Steel, M.J.: Order-based dependent dirichlet processes. Journal of the American statistical Association **101**(473), 179–194 (2006)
22. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proceedings of the National Academy of Sciences **101**(suppl 1), 5228–5235 (2004)
23. Kawamae, N.: Trend analysis model: trend consists of temporal words, topics, and timestamps. In: Proceedings of the fourth ACM international conference on Web search and data mining, pp. 317–326. ACM (2011)

24. Kawamae, N.: Theme chronicle model: Chronicle consists of timestamp and topical words over each theme. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pp. 2065–2069. ACM, New York, NY, USA (2012). DOI 10.1145/2396761.2398573
25. Kingman, J.F.: On the genealogy of large populations. Journal of Applied Probability pp. 27–43 (1982)
26. Kingman, J.F.C.: The coalescent. Stochastic processes and their applications **13**(3), 235–248 (1982)
27. Leskovec, J., Backstrom, L., Kleinberg, J.: Meme-tracking and the dynamics of the news cycle. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 497–506. ACM (2009)
28. Li, A.Q., Ahmed, A., Ravi, S., Smola, A.J.: Reducing the sampling complexity of topic models. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 891–900. ACM (2014)
29. Lin, D., Grimson, E., Fisher III, J.W.: Construction of dependent dirichlet processes based on poisson processes (2010)
30. MacEachern, S.N.: Dependent dirichlet processes. Unpublished manuscript, Department of Statistics, The Ohio State University (2000)
31. Neal, R.M.: Markov chain sampling methods for dirichlet process mixture models. Journal of computational and graphical statistics **9**(2), 249–265 (2000)
32. Neal, R.M.: Density modeling and clustering using dirichlet diffusion trees. Bayesian Statistics **7**, 619–629 (2003)
33. Ren, L., Dunson, D.B., Carin, L.: The dynamic hierarchical dirichlet process. In: Proceedings of the 25th international conference on Machine learning, pp. 824–831. ACM (2008)
34. Ristic, B., Arulampalam, S., Gordon, N.: Beyond the Kalman filter: Particle filters for tracking applications, vol. 685. Artech house Boston (2004)
35. Shahaf, D., Yang, J., Suen, C., Jacobs, J., Wang, H., Leskovec, J.: Information cartography: creating zoomable, large-scale maps of information. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1097–1105. ACM (2013)
36. Sun, Y., Tang, J., Han, J., Chen, C., Gupta, M.: Co-evolution of multi-typed objects in dynamic star networks. Knowledge and Data Engineering, IEEE Transactions on **PP**(99), 1–1 (2013). DOI 10.1109/TKDE.2013.103
37. Teh, Y.W.: A hierarchical bayesian language model based on pitman-yor processes. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pp. 985–992. Association for Computational Linguistics (2006)
38. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical Dirichlet processes. Journal of the American Statistical Association **101**(476), 1566–1581 (2006)
39. Teh, Y.W., Kurihara, K., Welling, M.: Collapsed variational inference for HDP. In: Advances in Neural Information Processing Systems, vol. 20 (2008)
40. Thibaux, R., Jordan, M.I.: Hierarchical beta processes and the indian buffet process. In: International conference on artificial intelligence and statistics, pp. 564–571 (2007)
41. Wallach, H.M., Murray, I., Salakhutdinov, R., Mimno, D.: Evaluation methods for topic models. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 1105–1112. ACM (2009)
42. Wang, C., Paisley, J.W., Blei, D.M.: Online variational inference for the hierarchical dirichlet process. In: International Conference on Artificial Intelligence and Statistics, pp. 752–760 (2011)
43. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. Pattern Analysis and Machine Intelligence, IEEE Transactions on **31**(3), 539–555 (2009)
44. Wang, X., McCallum, A.: Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD, pp. 424–433. ACM (2006)
45. Xu, K., Kliger, M., Hero III, A.: Adaptive evolutionary clustering. Data Mining and Knowledge Discovery **28**(2), 304–336 (2014). DOI 10.1007/s10618-012-0302-x. URL `http://dx.doi.org/10.1007/s10618-012-0302-x`

46. Xu, M.E.K.J.: A density-based algorithm for discovering clusters in large spatial databases with noise. pp. 226–231. AAAI (1996)
47. Yang, T., Chi, Y., Zhu, S., Gong, Y., Jin, R.: Detecting communities and their evolutions in dynamic social networksa bayesian approach. Machine learning **82**(2), 157–189 (2011)
48. Yao, L., Mimno, D., McCallum, A.: Efficient methods for topic model inference on streaming document collections. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 937–946. ACM (2009)
49. Zhang, J., Song, Y., Zhang, C., Liu, S.: Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1079–1088. ACM (2010)
50. Zhang, P., Zhou, C., Wang, P., Gao, B., Zhu, X., Guo, L.: E-tree: An efficient indexing structure for ensemble models on data streams. IEEE Trans. Knowl. Data Eng. **27**(2), 461–474 (2015)
51. Zhang, W., Li, R., Feng, D., Chernikov, A., Chrisochoides, N., Osgood, C., Ji, S.: Evolutionary soft co-clustering: formulations, algorithms, and applications. Data Mining and Knowledge Discovery **29**(3), 765–791 (2015). DOI 10.1007/s10618-014-0375-9. URL `http://dx.doi.org/10.1007/s10618-014-0375-9`