Dynamic Topic Model for Topic Split and Merge

Jing Chen JINGC1@ANDREW.CMU.EDU
Mengtian Li MENGTIAL@ANDREW.CMU.EDU
Lanxiao Xu LANXIAOX@ANDREW.CMU.EDU
Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213 USA

1. Introduction

People have witnessed an exponential boom of textual materials in the past few decades. These textual materials usually contain various topics that are evolving over the time. In the topic evolution process, the topics might remain the same, splits to new topics and merge with other existing ones to form a new topic. For example, in a series of blogs on financial crisis, topic in this text corpora might start with stocks and then shifts to government, bailout, etc.. A visualization of the topic split and merge is shown in Figure 1 for VisWeek publication data over the course of 2001 to 2010.

This overwhelming amount of textual materials make it impossible to manually scanning and organizing the texts in an efficient way. To enable automatic textual materials management, we need to first uncover the latent topics in the documents and then track how the topics evolve within the text corpora.

The Dirichlet process (DP) mixture models (Escobar & West, 1995) have been widely adopted in the research community to cluster the documents based on their topics without specifying the the number of clusters. This nonparameteric method is a generalization from finite mixture models to infinite mixture models. To capture more interaction of the topics in text, the hierarchical version of DP is proposed (Teh et al., 2012). The hierachical Dirichlet process (HDP) is able to model the information exchange between different text corporas.

While many methods have been proposed and shown to be effective in revealing the latent topics in static textual materials, it remains a challenging task to capture properties of and track the dynamic evolution of latent topics over certain time period. Evolutionary hierarchical Dirichlet process (EvoHDP) (Zhang et al., 2010) is proposed to address this problem. Their model is formulated as a series of HDPs by adding time dependencies to the adjacent epochs. The model is inferred by a cascaded Gibbs sampler based on an extended metaphor of the Chinese restaurant process. The benefit of having these additional dependencies is the ability to discover different evolving patterns of clusters, including emergence, disappearance, evolution. However, EvoHDP is not focused on modeling the split and merge of topics and assumes a simple weighted average form for the topic transition between different epochs. It is important in real-world application to understand how various topics gradually disintegrate (topic split) and dissolve into other topics (topic merge).

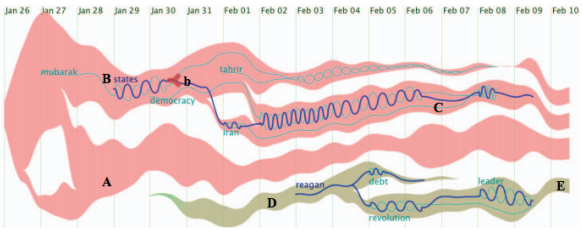

Figure 1. A Visualization of Dynamic Topic Evolution. Picture from (Cui et al.)

Appreciating this importance, we aim to construct an online dynamic topic model that can automatically recognize and visualize the dynamic evolutionary behaviors of multiple topics within large text collections. The split and merge will be modeled by a set of transition vectors from the previous epoch to the current epoch. These vectors can form a matrix all together which can be visualized to reveal the split and merge between topics.

The challenges of this work lie in that the space of split and merge is too large too model directly, that the number of topics varies at different time steps, and that introducing an transition matrix makes the inference task harder. All these challenges will be addressed by our model presented in Section 3.

2. Related Work

In this section, we review two categories of related work, including learning dynamic topic evolution patterns from a time-varying corpus and related sampling methods.

2.1. Dynamic Topic Evolution

It has long been a challenging task to fully understand topic evolution in large text corpora. One challenge is to appropriately define different types of dynamic behaviors of topic evolution and design a model that can effectively encode such evolutionary patterns. Figure 1 is a visualization of the dynamic topic evolution along the time frame produced by (Cui et al.). From this visualization we can observe that topics evolve over time in various ways, such as splitting into new topics or or emerging into other topics, both can generate new topics. In our project we aim to design a probabilistic graphical model, where the present topic distribution is dependent on the previous state in a way that the probability of topic splitting and merging are properly defined.

Research work on text mining has seen many models and algorithms proposed over recent years to automatically detect new topics and find new stories on known topics in temporally-ordered streams. While traditional topic detection and tracking methods generally use heuristic rules, modern machine learning approaches have brought new chances, such as the dynamic Latent Dirichlet Allocation (LDA) topic model (Wang & McCallum, 2006) and its extensions. Evolutionary clustering has later been proposed to produce a sequence of clusters (Xu et al., 2008), where the objective is to preserve the smoothness of clustering results over time. As explained previouly the EvoHDP is a model for handling time-varing contents from multiple text sources, where it is formulated as a series of hierarchical Dirichlet processes (HDP) by adding time dependencies to the adjacent time epoch, but the model can not characterize more complicated behaviors like the split and merge among different topics.

Intense insight into the complicated interaction among multiple topics is taken in the work of (Cui et al.), which proposes TextFlow system to allow coherent visualization based on topic evolution models. More specifically, the topic split/merge activities are detected as critical event, where the potential split/merge activity with highest ranking score is selected as the critical event. Then a visualization is designed accordingly to offer visual aid for users to conveniently see how topics interact with each other. However, the detection of critical events in this work is based on the post-processing the output of (Zhang et al., 2010). As a result, directly cascading the previous two work may lead to suboptimal results with respect to different steps of our detecting and tracking system.

Another time-series based hierarchical model is proposed by Li & Li (2013) for generating timeline summarization. The model adopts the similar linear dependency of DPs of consecutive timestamps. Therefore, the model still lacks the capability to model complicated topic interactions.

2.2. Sampling methods

Another challenge in topic evolution is to utilize the designed probabilistic graphical model to conduct inference and learning to detect the split and merge behaviors among topics. Basics techniques include Gibbs sampling for DPs and HDPs, in which a Chinese restaurant process (Aldous, 1985) is used to sample the count or the partition of the topics. As an alternative, a more efficient sampling scheme for the posterior of an HDP is proposed (Teh et al., 2012) with augmented representation for the base measure shared across different child DPs.

More advanced techniques includes Markov chain Monte Carlo (MCMC) methods. Due to the vast literature on this topic, we cannot enumerate them all and therefore we only list a few that is relevant to this work. Split-merge MCMC methods were proposed for Dirichlet process (DP) mixture models with conjugate likelihoods (Jain & Neal, 2004). The main idea is a generic technique for proposing splits consistent with data, called restricted Gibbs (RG) sampling. The split is initially done at random, then, after several passes, the RG sampler encourages a reassigning that agrees with the data. However, this approach only works for static models and is computationally costly for large datasets as it requires many passes.

Building upon (Jain & Neal, 2004), (Hughes et al., 2012) adopts sequential allocation (Dahl, 2005) method, in which multiple sampling sweeps are replaced with randomly choosing two anchoring items and a single sampling sweep. In (Hughes et al., 2012), data-driven (Tu & Zhu, 2002) reversible jump moves are designed to efficiently discover the features unique to a single sequence. This improved method is proposed for Beta process hidden models (BP-HMM), a model designed to identify a concise global library of discriminative features, to annotate each sequence with a subset of features from the library, and to generate a label of active feature for each timestep. The main difference of sampling for DP and for BP-HMM is the way of choosing the anchors and the candidate states to split or merge from.

3. Model

In this section, we first define the mathematical notations adopted in our paper, then propose our graphical model for mining topic evolving patterns, e.g. split and merge, using the transition matrix Λ. We also cover the motivation behind proposed model as well as a simple comparison with the previous research work.

3.1. Preliminaries

Here we present a brief overview of the Dirichlet process and with two perspectives on DPs: the stick-breaking con-

struction and the Chinese restaurant process. Let H be a distribution over Θ and α be a positive real number. Then for any finite measurable partition $A_1, ..., A_r$ of Θ the vector $(G(A_1), ..., G(A_r))$ is random since G is random. We say G is Dirichlet process distributed with base distribution H and concentration parameter α, written $G \sim \text{DP}(\alpha, H)$, if

$$(G(A_1), ..., G(A_r)) \sim \text{Dir}(\alpha H(A_1), ..., \alpha H(A_r)))$$

for every finite measurable partition $A_1, ..., A_r$ of Θ. Samples drawn from DPs are infinite discrete distributions.

The stick-breaking process offers a constructive procedure of obtaining Dirichlet measures. It is based on independent sequences of iid random variables $(w_k)_{k=1}^{\infty}$ and $(\phi_k)_{k=1}^{\infty}$,

$$w_k|\alpha_0, G_0 \sim \text{Beta}(1, \alpha_0), \quad \phi_k|\alpha_0, G_0 \sim G_0 \tag{1}$$

And we define G as

$$\pi_k = w_k \prod_{l=1}^{k-1}(1 - w_l), \quad G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k} \tag{2}$$

where δ_ϕ is a probability measure concentrated at ϕ. According to (Sethuraman, 1994), G is a random probability measure distributed according to $\text{DP}(\alpha_0, G_0)$. And for convenience, we denote this stick-breaking process as $\pi_k \sim \text{GEM}(\alpha_0)$.

The scheme of Chinese restaurant process (Blackwell & MacQueen, 1973) shows that draws from DPs are both discrete and exhibit a clustering property. This scheme does not refer to G directly; rather, it refers to draws from G. Let $\theta_1, \theta_2, ...$ be a sequence of iid random variables distributed according to G. We have:

$$\theta_i|\theta_1, ..., \theta_{i-1}, \alpha_0, G_0 \sim \sum_{l=1}^{i-1} \frac{\delta_{\theta_l}}{i - 1 + \alpha_0} + \frac{\alpha_0 G_0}{i - 1 + \alpha_0} \tag{3}$$

Define $\phi_1, ..., \phi_K$ to be the distinct values taken on by $\theta_1, ..., \theta_{i-1}$, and let m_k be the number of values $\theta_{i'}$ that are equal to ϕ_k for $1 \leq i' < i$. We can re-express (3) as

$$\theta_i|\theta_1, ..., \theta_{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{m_k \delta_{\phi_k}}{i - 1 + \alpha_0} + \frac{\alpha_0 G_0}{i - 1 + \alpha_0} \tag{4}$$

Similarly, after observing $\theta_1, ..., \theta_{i-1}$ draws from G, the posterior of G is still a DP

$$G|\theta_1, ..., \theta_{i-1} \sim \text{DP}(\alpha_0 + i - 1, \frac{\sum_{k=1}^{K} m_k \delta_{\phi_k} + \alpha_0 G_0}{\alpha_0 + i - 1}) \tag{5}$$

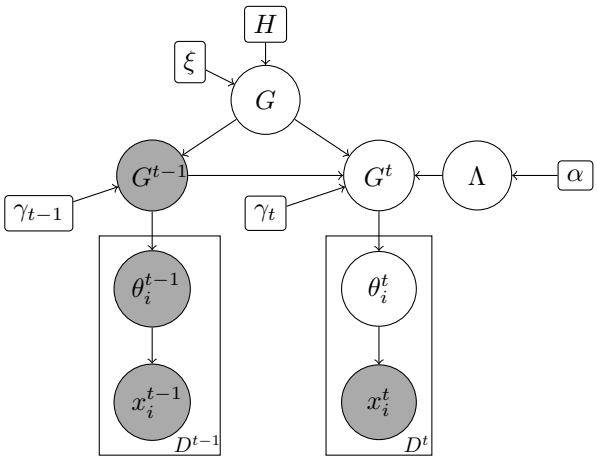

Figure 2. The origianl representation of the graphical model

3.2. Motivation

To uncover topic evolution in text stream corpus, we base our model on dynamic nonparametric topic models, more specifically, the EvoHDP model (Zhang et al., 2010), which can effectively discover different evolving patterns within a corpus and across different corpora. However, one problem with EvoHDP is that it assumes that a topic in one time epoch is either a new topic or derived from one topic in the previous time epoch. This assumption does not make sense in most real-life applications, where topics usually interact with each other rather frequently.

Based on the above intuition, our model aims to extend the EvoHDP model by encoding the dependencies between one topic in current time epoch and all topics in the previous time epoch. Moreover, such dependency factors can be further be interpreted to convey topic evolving split and merge patterns over time.

Therefore, while previous work (Cui et al.) interpret and visualize split and merge patterns of topics by post-processing based on the output of (Zhang et al., 2010), our approach would be capable to provide a unified dynamic nonparametric model that directly captures the evolving topic behaviours.

3.3. Proposed Model

In this section, we first present a detailed introduction of our model in its original representation and then explain the stick-breaking construction version of the model for the convenient of later derivation. The two representations are equivalent.

We first introduce some settings and notations to make further discussions clear. There are T time epochs and within each epoch there are D^t documents observed. We assume

the underlying model to generate document x_i^t at epoch t is an infinite mixture model

$$p^t(x^t|G^t) = \int G^t(\theta^t) f(x^t|\theta^t) d\theta^t$$

where $G^t = \sum_{k=1}^{\infty} \beta_k^t \delta_{\phi_k}$ 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 the topic-word distribution of a topic.

Our goal is to model a time-varying corpus as a series of DPs with time dependencies while maintaining an overall bookkeeping of components for all epochs. We let these DPs share an identical discrete base measure G, and G is drawn from DP(ξ, H) with H as the base measure. We call G the *overall measure*. Moreover, for each time epoch t, we use G^t to denote the measure at this epoch, which we call *snapshot measure*. Since our goal is to conducting online modeling of topic evolution, for each time epoch t, the snapshot measure G^{t-1} and corresponding topic assignment of each document are assumed to be fixed or observed.

The key issue of our problem is the dependency of snapshot measure G^t on G^{t-1}. In this project, the dependency is incorporated by a *dependency factor* Λ. The topic split/merge information is also encoded in the dependency factors.

Figure 2 is a graphical representation of the model we described above. And in accordance with this representation, the generation process of our model is as follows.

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

2. For each time epoch t, draw the snapshot measure G^t according to the overall measure G and the previous snapshot measure G^{t-1}:

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

where Λ is designed such that probablity of each topic $P(\theta = \phi_k)$ of $\Lambda(G^{t-1})$ is the weighted sum of that in the previous time epoch, G^{t-1}. A comparison of our design and (Zhang et al., 2010) is shown in Figure 3. In (Zhang et al., 2010), the $G^t \sim$ DP$(\gamma^t, w^t G^{t-1} + (1-w^t)G)$, which means the prior of G^t is a weighted average of the snapshot measure of the previous time epoch G^{t-1} and the overall measure. Under this condition, the probability of each topic in the G^{t-1} is scaled by the same factor w^t. However, in our design, the probability of each topic in the prior of G^t is dependent on all the topics in G^{t-1}. By learning the Λ, we can obtain the split/merge information in topic evolution.

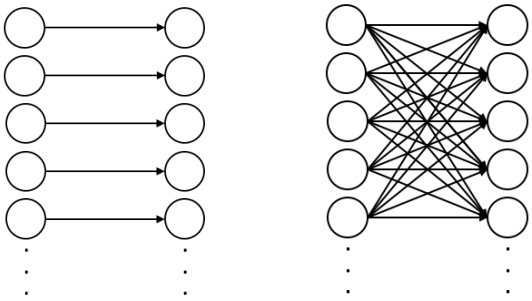

Figure 3. Comparison of the EvoHDP (Zhang et al., 2010) and our model

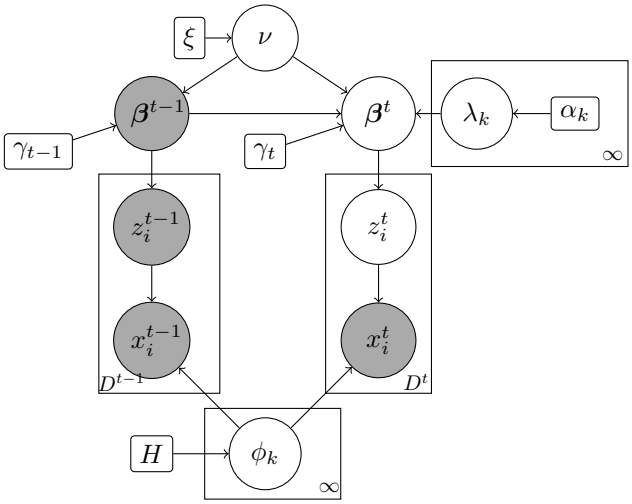

Figure 4. The stick-breaking representation of the graphical model

3. For data samples $\{x_i\}_{i=1}^{D^t}$ in each time epoch t, draw the parameters of the component densities and generate the data samples:

$$\theta_i^{t-1} \sim G^t, \quad x_i^t \sim F(x|\theta_i^t) \quad (7)$$

3.4. The Stick-Breaking Construction

According to the stick-breaking construction of a DP, we can write the explicit form of G:

$$G = \sum_{k=1} \nu_k \delta_{\phi_k}, \quad \boldsymbol{\nu} \sim \text{GEM}(\xi) \quad (8)$$

Consequently, according to Eqs. (6), G^t has the form

$$G^t = \sum_{k=1}^{\infty} \beta_k^t \delta_{\phi_k}, \quad \boldsymbol{\beta}^t \sim \text{DP}(\gamma^t, \hat{\boldsymbol{\beta}}^t) \quad (9)$$

where for $k = 1, 2, ...,$ $\hat{\beta}_k^t = w^t \dfrac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + (1 - w^t)\nu_k$,

and $\dfrac{\boldsymbol{\lambda}_k}{\sum_j \lambda_{kj}} \sim \text{Dir}(\alpha_k)$

In this way, we obtain the stick-breaking construction for our model, which is shown in Figure 4.

3.5. A Restaurant Franchise Metaphor

Based on the stick-breaking construction described in the last section, we continue to offer a more intuitive way of understanding this model by using a metaphor following the Chinese restaurant franchise (CRF) (Teh et al., 2012). The corpus in our problem 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_i^t. Having z_i^t, the left generation process of x_i^t is straightforward, which is drawn from $F(x|\phi_{z_i^t})$.

The generation mechanism of each snapshot measure $G^t = \sum_{k=1}^{\infty} \beta_k^t \delta_{\phi_k}$,i.e., the behavior of the restaurant in day t. Since $\boldsymbol{\beta}^t$ is drawn from DP$(\gamma^t, \hat{\boldsymbol{\beta}}^t)$ as shown in Eq. (9), we can represent this DP using the stick-breaking construction as

$$\boldsymbol{\beta}^t = \sum_{m=1}^{\infty} u_m^t \delta_{k_m^t}, \{k_m^t\}_m \sim \hat{\boldsymbol{\beta}}^t, \boldsymbol{u}^t \sim \text{GEM}(\gamma^t) \quad (10)$$

We call each m a table, then $\boldsymbol{u}^t$ is a distribution on tables. The manager has infinite number of empty tables beforehand and he selects a dish k_m^t for each table m from the *table-dish menu* $\hat{\boldsymbol{\beta}}^t$.Then a customer i enters the restaurant, he select table m_i^t from the table distribution $\boldsymbol{u}^t$ and enjoys the dish $k_{m_i^t}$ in this table. We denote M_k^t by the number of tables with dish k in day t and N_k^t by the number of customers enjoy dish k in day t. Thus $M_k^t = \sum_m I(k_m^t = k)$ and $N_k^t = \sum_i I(z_i^t = k)$. Since $\hat{\beta}_k^t = w^t \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + (1-w^t)\nu_k$, with probability w^t a dish k_m^t is selected from the *local combined menu* $\frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}}$ and with probability $(1-w^t)$, a dish k_m^t is selected from the *global menu* $\boldsymbol{\nu}$. We denote the number of tables selected from the global menu in day t as $M_k^{0\to t}$, and the number of tables selected from the local combined menu as $M_k^{t-1\to t}$. Thus $M_k^t = M_k^{0\to t} + M_k^{t-1\to t}$.

According to Eq. (9), for table m, a waiter select a dish k_m. For each dish k the probability of been chosen is $\hat{\beta}_k^t = w^t \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + (1-w^t)\nu_k$, which means that the probability of each dish to be chosen equals to a linear combination between the probability to be chosen in day $t-1$ for each dish and the dependency factor w^t. In our problem, $\{\lambda_k\}_k$ encodes the topic evolutionary behavior. For example, for topic k, if all entries in $\boldsymbol{\lambda}_k$ equals to zero expect for λ_{kj} and λ_{kh}, then it means that the topic j and topic h in time epoch $t-1$ are merged into topic k. Thus to uncover meaningful split/merge evolutionary behavior, we want the matrix $\Lambda = [\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, ...]^T$ to be sparse.

4. Gibbs Sampling

Based on the infinite mixture model and restaurant franchise metaphor in previous sections, we can further define a Gibbs sampling scheme, where the following variables are sequentially sampled.

4.1. Sampling $\boldsymbol{\nu}$

From the initial time $T = 0$ to the time epoch $T = t$. We denote the total number of tables with dish k as $\mathcal{M}_k^t = \sum_{\tau=1}^{t} M_k^{0\to\tau} = \mathcal{M}_k^{t-1} + M_k^{0\to t}$, then the total number of tables is $\mathcal{M}_\cdot^t = \sum_k \mathcal{M}_k^t$. Given that $G \sim DP(\xi, H)$, the posterior of G is a DP :

$$G|\xi, H, \{\mathcal{M}_k^t\}_{k=1}^K \sim \text{DP}(\xi + \mathcal{M}_\cdot^t, \frac{H + \sum_{k=1}^K \mathcal{M}_k^t \delta_{\phi_k}}{\xi + \mathcal{M}_\cdot^t}) \quad (11)$$

where K is the number of distinct dishes. Theoretically there are infinite number of dishes and the number of tables. But in actual implementation only a finitely number of tables and dishes are associated with the data and represented explicitly. Similar to Sec. 5.2 of (Zhang et al., 2010), G can be represented as,

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

$$\nu = (\nu_1, ..., \nu_K, \nu_u) \sim \text{Dir}(\mathcal{M}_1^t, ..., \mathcal{M}_K^t, \xi) \quad (13)$$

This *augmented* representation of G can reformulates the original infinite vector ν to an equivalent finite one with length $K+1$, thus ν is then sampled from the Dirichlet distribution given in Eq. 13. Then G^t can be rewritten using the augmented representation as,

$$G^t = \sum_{k=1}^{K} \beta_k^t \delta_{\phi_k} + \beta_u^t G_u, G_u \sim \text{DP}(\xi, H) \quad (14)$$

And correspondingly β^t can be represented as,

$$\beta^t = (\beta_1^t, ..., \beta_K^t, \beta_u^t) \quad (15)$$

4.2. Sampling $\boldsymbol{\beta}^t$

We see from Figure 4, z_i^t are samples drawn from $\boldsymbol{\beta}^t$, and $\boldsymbol{\beta}^t$ is dependent on $\hat{\boldsymbol{beta}}^t$, $\boldsymbol{\nu}$ and γ^t. Since $\beta^t \sim$ DP$(\gamma^t, \hat{\beta}^t)$ and given N_k^t, the posterior of β^t is also a DP can can also be sampled from a Dirichlet distribution as,

$$(\beta_1^t, ..., \beta_K^t, \beta_u^t) \sim \text{Dir}(\tilde{\gamma}^t(\tilde{\beta}_1^t, ..., \tilde{\beta}_K^t, \tilde{\beta}_u^t)) \quad (16)$$

where $\tilde{\gamma}^t = \gamma^t + \sum_{k=1}^K N_k^t$ and

$$\tilde{\beta}_k^t = \frac{1}{\tilde{\gamma}^t}(N_k^t + w^t\gamma^t\frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + \gamma^t(1-w^t)\nu_k) \quad (17)$$

$$\tilde{\beta}_u^t = \frac{1}{\tilde{\gamma}^t}(w^t\gamma^t\frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + \gamma^t(1-w^t)\nu_u) \quad (18)$$

4.3. Sampling $\boldsymbol{\lambda}_k$

Given $\boldsymbol{\beta}^t$ and α_k, it is straightforward to sample $\boldsymbol{\lambda}_k$ as,

$$p(\boldsymbol{\lambda}_k|\text{rest}) \propto p(\lambda_k|\alpha_k)p(\beta^t|\lambda_k, \boldsymbol{\beta}^{t-1}, \gamma^t) \quad (19)$$

where $k \in \{1, ..., K, u\}$ with u denoting the index of the new component. Theoretically there should be infinite number of $\boldsymbol{\lambda}_k$s since there are infinite number of topics. But similar to the augmented representation in section 4.1 and 4.2, we only consider the topics that are associated with the data in time epoch t and augment the "unknown topics" into one single dimension. Let K' denote the number of topics associated with time epoch $t-1$. We have

$$p(\lambda_k) \propto \prod_{j=1}^{K'} \lambda_{kj}^{\alpha_k - 1} \quad (20)$$

And the likelihood $p(\beta^t|\boldsymbol{\lambda}, \boldsymbol{\beta}^{t-1}, \gamma^t)$ is:

$$p(\beta^t|\boldsymbol{\lambda}, \boldsymbol{\beta}^{t-1}, \gamma^t) \quad (21)$$

$$\propto \Big(\prod_{k=1}^{K} {\beta_k^t}^{\gamma^t \sum_{j=1}^{K'} \beta_j^{t-1}\lambda_{kj}/\sum_j \lambda_{kj}}\Big) {\beta_u^t}^{\gamma^t \sum_{j=1}^{K'} \beta_j^{t-1}\lambda_{uj}/\sum_j \lambda_{uj}} \quad (22)$$

$$\propto \prod_{j=1}^{K'} \Big((\prod_{k=1}^{K} {\beta_k^t}^{\gamma^t \beta_j^{t-1}\lambda_{kj}/\sum_j \lambda_{kj}}) {\beta_u^t}^{\gamma^t \beta_j^{t-1}\lambda_{uj}/\sum_j \lambda_{uj}}\Big) \quad (23)$$

By combining Eq. 20 and 21, we obtain

$$p(\lambda_{kj}|\text{rest}) \propto \lambda_{kj}^{\alpha_k - 1} {\beta_k^t}^{\gamma^t \beta_j^{t-1}\lambda_{kj}/\sum_j \lambda_{kj}} \quad (24)$$

where $k \in \{1, ..., K, u\}$. From this result we can see that $p(\lambda_{kj}|\text{rest})$ is not a standard distribution. Thus in practice we can not sample λ_{kj} directly. In this project, we use slice sampling to obtain samples of λ_{kj}.

4.4. Sampling z_i^t

Based on the augmented formulation of $\boldsymbol{\beta}$, according to (Teh et al., 2012), the posterior of z_i^t can be expressed as

$$p(z_i^t|\text{rest}) \quad (25)$$

$$= \begin{cases} (N_k^{t,-i} + \gamma^t\hat{\beta}_k^t) f_k^{-x_i^t}(x_i^t) & \text{if } k \text{ previously used} \\ \gamma^t\hat{\beta}_u^t f_{\text{new}}^{-x_i^t}(x_i^t) & \text{if } k = k^{\text{new}} \end{cases} \quad (26)$$

where $N_k^{t,-i}$ is the number of customers except for x_i^t who enjoy dish k, $f_k^{-x_i^t}(x_i^t)$ is the conditional density of x_i^t under mixture component k given all data items except x_i^t. Let $F(\theta)$ have density $f(\cdot|\theta)$ and let H have density $h(\cdot)$. Since in most cases F is conjugate to H, we integrate out the mixture component $\boldsymbol{\phi}$ and obtain

$$f_k^{-x_i^t}(x_i^t) = \frac{\int f(x_i^t|\phi_k) \prod_{i't' \neq it, z_{i'}^{t'}=k} f(x_{i'}^{t'}|\phi_k)h(\phi_k)d\phi_k}{\int \prod_{i't' \neq it, z_{i'}^{t'}=k} f(x_{i'}^{t'}|\phi_k)h(\phi_k)d\phi_k} \quad (27)$$

And $f_{\text{new}}^{-x_i^t}(x_i^t)$ can be expressed as

$$f_{\text{new}}^{-x_i^t}(x_i^t) = \int f(x_i^t|\phi)h(\phi)d\phi \quad (28)$$

which is simply the prior density of x_i^t.

4.5. Sampling $M_k^{0\to t}$ and M_k^t

As described in section 4.1 and section 4.2, the sampling of $\boldsymbol{\nu}$ and $\boldsymbol{\beta}^t$ is dependent on $M_k^{0\to t}$, N_k^t, where $M_k^{0\to t}$ is dependent on M_k^t. Since in section 4.4 we have obtain samples of z_i^t and according to section 3.4 we know that $N_k^t = \sum_i I(z_i^t = k)$, we can compute N_k^t given samples of z_i^t. In this section, we will focus on the sampling of $M_k^{0\to t}$ and M_k^t in detail.

Recall that M_k^t is the number of tables with dish k in the Chinese Restaurant Metaphor. According to (Teh et al., 2012), we can sample $M_k^{0\to t}$ and M_k^t in the following process:

$$(M_k^{0\to t}, M_k^{t-1\to t}) \sim \text{Multinomial}(M_k^t, [p, 1-p]) \quad (29)$$

where $p = \dfrac{(1-w^t)\nu_k}{w^t\dfrac{\boldsymbol{\lambda}_k \cdot \boldsymbol{\beta}^{t-1}}{\sum_j \lambda_{kj}} + (1-w^t)\nu_k}$

According to (Teh et al., 2012), samples of M_k^t can be obtained from this distribution:

$$p(M_k^t = m|\text{rest}) \quad (30)$$

$$= \frac{\Gamma(\gamma^t\hat{\beta}_k^t)}{\Gamma(\gamma^t\hat{\beta}_k^t + N_k^t)} s(N_k^t, m)(\gamma^t\hat{\beta}_k^t)^m \quad (31)$$

where $s(n, m)$ are unsigned Stirling numbers of the first kind. We have by definition that $s(0,0) = s(1,1) = 1$, $s(n,0) = 0$ for $n > 0$ and $s(m,n) = 0$ for $m > n$. Other entries can be computed by $s(n+1, m) = s(n, m-1) + ns(n, m)$.

4.6. Sampling Hyper-parameter

Hyper-parameters like ξ, γ^t and α_k can be sampled via a vague gamma prior as is shown in (Teh et al., 2012).

$$\xi \sim Ga(a_\xi, b_\xi) \tag{32}$$
$$\gamma^t \sim Ga(a_\gamma, b_\gamma) \tag{33}$$
$$\alpha_k \sim Ga(a_{\alpha_k}, b_{\alpha_k}) \tag{34}$$

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 The sampling procedure at each time epoch t

for $i = 1, ..., D^t$ **do**
  Sampling z^t according to Sec. 4.4
end for
for $k = 1, \cdots, K$ **do**
  Sampling count variables M_k^t and $M_k^{0\to t}$ according to Sec. 4.5
  Sampling concentration parameter α_k
end for
Sampling concentration parameters ξ, γ^t
Sampling $\boldsymbol{\nu}$ according to Sec. 4.1
for $k = 1, \cdots, K$ **do**
  Sampling $\boldsymbol{\lambda}_k$ according to Sec. 4.3
end for
Sampling $\boldsymbol{\beta}^t$ according to Sec. 4.2

5. Implementation

We implement our model in a joint MATLAB and C++ framework, allowing flexible interface and efficient implementation at the same time. Our implementation is based on the HDP implementation by Yee Whye Teh (`http://www.cs.berkeley.edu/~jordan/hdp`). We modified the code to take input a time series of data instead of corpora of documents at a single time step. Also, we implement a new sampling scheme according to Section 4.

6. Experiments

In this section, we experiment with our proposed model on a real-world dataset. We start from the dataset description and adopted pre-processing, then give important experimental detail, including parameter settings, etc. An elaborative analysis on the experimental results are provided.

6.1. Dataset

For the experiment, we use the DBLP Citations dataset version 2 (Tang et al., 2008). This dataset contains the papers from the DBLP. For each paper, we have the year, the publication venue, the abstract and the citations. We take a small subset of the papers with year ranging from 1996 to 2010, with the total number of papers as 7,500. We treat each year as a timestamp and the we are trying to model the topic evolution across the years.

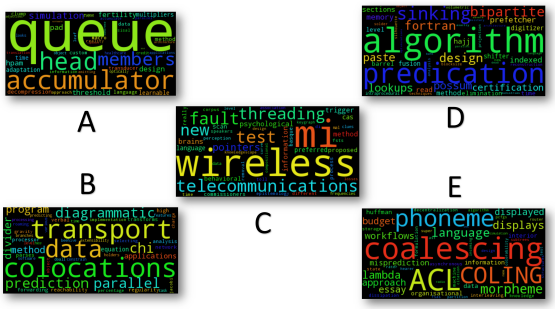

Figure 5. Topic Modeling (1997 - 1998)

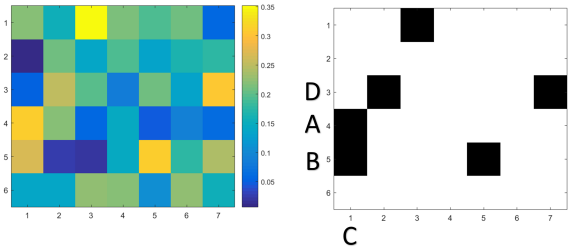

Figure 6. The Lambda Matrix (1997 - 1998)

6.2. Experimental Setup

For simplicity, we perform some simple pre-processing steps on the DBLP corpus, including (1) remove the stop words, which otherwise would bias the result due to extremely high corpus term frequency; (2) only consider English words and turn all letters into lower case for more effective training, etc. Moreover, since the publication venue might be is a perfect indicator of the field of study, which further contributes to the clustering of topics, we append the publication venue to the abstract as an additional word. After performing pre-processing steps, we arrive at a final vocabulary size of 25,356.

Based on the properties of dynamic topic modeling, we set the application dependent distribution F to a multinomial one, representing the probability of generating the abstract of the paper for a particular topic.

For each timestamp, the algorithm takes a sample snapshot before running 1000 burn-in iterations. The base measure H is set as an uniform distribution and the concentration hyper parameters are set to 1.

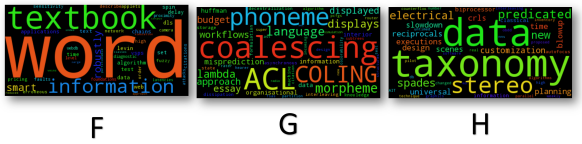

Figure 7. Topic Modeling (1998 - 1999)

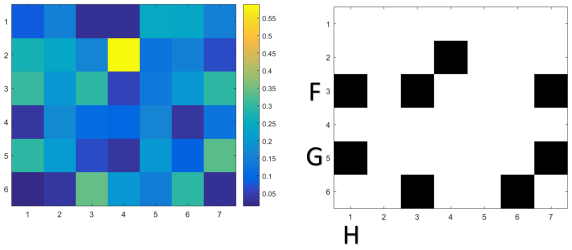

Figure 8. The Lambda Matrix (1998 - 1999)

6.3. Results and Analysis

From year 1997 to 1998. First we visualize the basic topic modeling capability of our model in Figure 5. Each box in the figure represents a word cloud for an identified topic by our model. For example, topic A is about the basic networking algorithms with keywords like queue and accumulator; topic C is about telecommunications; and topic E is about NLP, with ACL, COLING, the name of the top conferences, as key words.

The novelty of our model lies in the topic transition. We capture this evolutionary relationship with the lambda matrix and that is visualized in Figure 6. Each column of the matrix shows the transition strength of topics in time stamp $(t-1)$ to time stamp t. On the left is the original lambda matrix with value from 0 to 1. On the right is the matrix obtained after thresholding. The first column shows that topic A and B have high influence in the generation of topic C. The values in the lambda matrix are 0.312 and 0.267 respectively. This is consistent with the contents captured by these topics as A is about basic networking algorithm, B is co-location for telecommunications and C is about wireless telecommunications.

The block at the 3rd row, 1st column shows a negative example. The value is as small as 0.053, meaning that topic D has very limited influence on the generation of topic C. Topic D is about algorithms and programming languages and is not very relevant to telecommunications.

From year 1998 to 1999. Similarly, we present the topic modeling in Figure 7 and the lambda matrix in Figure 8. In this transition, we can see that topic F and G have high influence on the generation of topic H. The values in the lambda matrix are 0.303 and 0.294 respectively. Such a result is understandable as topic F is about words in textbooks, topic G is about natural language processing techniques, and topic H is about word taxonomy.

7. Conclusion and Future Work

In this paper, we have proposed a novel non-parametric graphical model to directly characterize the transition of topics in a time-varying corpus. While previous work is only able to model the one-to-one topic interaction across different time stamps, our model is able to capture more complicated interaction defined by our lambda matrix. We derived the formulae for the posterior distributions for parameters involving the lambda matrix and implemented a Gibbs sampler to perform the inference. Experimental results on the DBLP dataset have shown that our model can effectively capture the evolving patterns of research topics across different years.

While our model can effectively analyze dynamic topic behaviours, e.g. split and merge, there is still much remains to be done. Out of the intuition that a newly emerged topic at time epoch $t+1$ is most likely split or merged from only a small number of topics at time epoch t instead of all the topics, it would be interesting to enforce sparsity directly on the Λ transition matrix to allow better interpretability. Possible explorations can be : Treat the transition matrix as a dependency parameter instead of sampling from a prior, and put sparsity constraints on it, e.g. *Lasso*, which can be solved by common optimization algorithms, e.g. gradient descent. On the other hands, it is worth research work to design more effective visualization tools in order to vividly depict the topic evolution patterns over time, for example, using flow graph. Besides, our model can also be generalized to model topic evolution in multiple corpora instead of single-source corpus.

References

Aldous, David J. *Exchangeability and related topics*. Springer, 1985.

Blackwell, David and MacQueen, James B. Ferguson distributions via pólya urn schemes. *The annals of statistics*, pp. 353–355, 1973.

Cui, Weiwei, Tan, Shixai Liu, Shi, Conglei, Song, Yanggiu, Gao, Zekai J, Tong, Xin, and TextFlow, Huamin Qu. Towards better understanding of evolving topics in text ieee transactions on visualization and computer graphics.

Dahl, David B. Sequentially-allocated merge-split sampler for conjugate and nonconjugate dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 11, 2005.

Escobar, Michael D and West, Mike. Bayesian density

estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588, 1995.

Hughes, Michael C, Fox, Emily, and Sudderth, Erik B. Effective split-merge monte carlo methods for nonparametric models of sequential data. In *Advances in Neural Information Processing Systems*, pp. 1295–1303, 2012.

Jain, Sonia and Neal, Radford M. A split-merge markov chain monte carlo procedure for the dirichlet process mixture model. *Journal of Computational and Graphical Statistics*, 2004.

Li, Jiwei and Li, Sujian. Evolutionary hierarchical dirichlet process for timeline summarization. In *ACL*, pp. 556–560. Citeseer, 2013.

Sethuraman, Jayaram. A constructive definition of dirichlet priors. *Statistica sinica*, pp. 639–650, 1994.

Tang, Jie, Zhang, Jing, Yao, Limin, Li, Juanzi, Zhang, Li, and Su, Zhong. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 990–998. ACM, 2008.

Teh, Yee Whye, Jordan, Michael I, Beal, Matthew J, and Blei, David M. Hierarchical dirichlet processes. *Journal of the american statistical association*, 2012.

Tu, Znuowen and Zhu, Song-Chun. Image segmentation by data-driven markov chain monte carlo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5): 657–673, 2002.

Wang, Xuerui and McCallum, Andrew. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 424–433. ACM, 2006.

Xu, Tianbing, Zhang, Zhongfei, Yu, Philip S, and Long, Bo. Dirichlet process based evolutionary clustering. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pp. 648–657. IEEE, 2008.

Zhang, Jianwen, Song, Yangqiu, Zhang, Changshui, and Liu, Shixia. 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.