Topic Chronicle Forest for Topic Discovery and Tracking

Noriaki Kawamae
The University of Tokyo
Tokyo
kawamae@gmail.com

ABSTRACT

To ease the comprehension of existing time-stamped corpora, we extend topic models to handle both the specificity and temporality of topics; this is a significant advance over previous models which fail to provide both aspects simultaneously. Our proposed model combines the Topic Chronicle Forest (TCF) and Thematic Dirichlet Processes (TDP). TCF is a set of Topic Chronicle Trees, where each tree is a hierarchy of topics that becomes more specialized toward the leaves. Only one tree is defined in each time interval, a region, and is processed by TDP to generate a document. The advantage of our approach is that it provides more compact topic organization, while preserving both the semantics of the given corpus and the theme of each document. Experiments show that TCF is a useful extension for longitudinal topic discovery and tracking, and helps us to organize and digest data sets.

KEYWORDS

Time Varying Topic Models, Hierarchical Dirichlet Processes, Trend analysis, Hierarchical Bayesian Nonparametrics

ACM Reference Format:

Noriaki Kawamae. 2018. Topic Chronicle Forest for Topic Discovery and Tracking. In *Proceedings of The Eleventh ACM International Conference on Web Search and Data Mining (WSDM 2018).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3159652.3159653

1 INTRODUCTION

Our aim is to discover the topic organization of a given time-stamped corpus, and to track how this organization changes over time. This ability will help us peruse and comprehend time-stamped collections such as news, SNS, or papers, as topical mapping representing how topics evolve over time will be helpful in contextualizing information. While the notion of a "topic" is an "event" in the TDT pilot study [5], it is defined as a probability distribution over words that co-occur frequently in topic models such as Latent Dirichlet Allocation (LDA) [9] and its extensions. These term distributions can help us to understand what each topic represents. We extend these models to form a time-specific generative model that can answer the following questions: What kinds of topics are in the given data?, and How do they change over time?

WSDM 2018, February 5–9, 2018, Marina Del Rey, CA, USA

ACM ISBN 978-1-4503-5581-0/18/02…$15.00
https://doi.org/10.1145/3159652.3159653

To answer these questions, topic models are required to consider the difference between 1) specificity and temporality, 2) between temporality and periodicity of topics. For example, "LSTM" and "GRU" are new and specific terms in machine learning topics, while "LDA" and "HMM" are, in the same topics, specific and constant terms. "Neural network" and "AI" are periodic trends in the same topics, while "Web2.0" is a temporal trend of these topics. As previous models restrict themselves to either the specificity [7, 16, 18, 25, 26] or temporality [1–3, 8, 14, 34, 35] of topics, there is a need for unified models that distinguish between these differences simultaneously, and meet the two abovementioned requirements. This practical need motivates our focus on topic structures that integrate both the topic extraction task and the trend detection task.

To achieve our goal, we propose the combination of Topic Chronicle Forest (TCF) and Thematic Dirichlet Process (TDP) to create a hierarchical organization of given time-stamped corpora at different levels of granularity. TCF constructs time-interval specific topic trees, Topic Chronicle Trees (TCTs), and organizes the given corpus according to its content and trends to assist users in sifting and digesting them. These trees are based on the assumption that the difference between specificity and temporality of topics can be found by identifying topics and capturing temporal hierarchy changes instead of topic evolution. This leads TCF to identifying unique topics and constructing a set of TCTs that exhibit the temporal topic tree in each time interval, a region. In the previous example, "LSTM" and "GRU" are tagged as temporal specific topics and observed in only new trees, while "LDA" and "HMM" are tagged as long-period specific topics and observed in both new and old trees. This allows us to discover long-period specific topics and track their changes as temporal specific topics. TDP uses timestamps in selecting a TCT as a base probability measure from TCF like the hierarchical Dirichlet process (HDP) [31], and constructs a document-specific tree and generates a document via this tree to obtain thematic coherence.

Our experiments confirm the following contributions.

Theoretical contribution: (1) Unlike topic models built on either the specificity or temporality of topics, TCT handles both to create a topic hierarchy, and detects the differences between topic temporality and topic periodicity. (2) While most temporal topic models require manual selection of a fixed-width time frame (a heuristic task), TCF detects the appropriate variable-size time granularity automatically, and is purely data-driven. (3) TCF describes temporal evolution by selecting the TCT instead of topic evolution, and represents the periodicity of topics, where each topic is constant over time.

Practical contributions: (1) As TCF prevents the same topic from appearing (due to the periodicity of topics) as different topics in

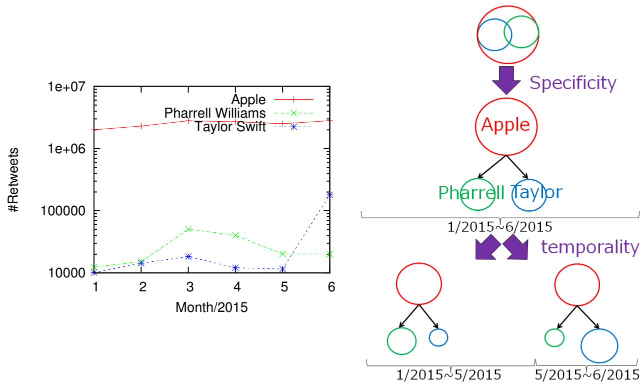

Figure 1: (left) This figure shows the number of retweets covering #apple: "Taylor Swift" was an extra hot topic after she posted a letter to "Apple" in Jun, while "Pharrell Williams" became a hot topic after wearing "Apple watch" in March. (right) The topic hierarchy constructed from the left example.

different trees, it yields more compact hierarchies than other hierarchical models. (2) TCF creates human-understandable hierarchies for topic monitoring without the need for manual intervention.

2 RELATED WORK

Topic detection is defined as generating topics from a document stream in [5], and the task of topic evolution is to discover how and what topics change over time. Topic models have been applied to both tasks. For example, Dynamic Topic Models [8] represents the evolution of topics by estimating the topic distribution at epochs. Recent models describe inter-topic correlations by applying constraints and priors to hyperparameters [27], or using a nonparametric Bayesian approach. The Chinese restaurant process (CRP) [4], a stochastic process that generates partitions of integers and is used in the Dirichlet process (DP) [23], yields the same clustering structures as created by a DP [11]. This is why our approach constructs topic trees to explain the relationship between topics using a nonparametric approach rather than non-negative matrix factorization [13, 17, 29, 32]. A DP can be considered as a distribution of random probability measure G, which we write as $G \sim DP(\gamma, H)$, where γ is a scaling parameter, and H is a base measure. We can view DP in terms of $CRP(\gamma, H)$ and compute the probability of the i-th customer sitting at table k in restaurant j as follows:

$$P(z_i = k|\mathbf{z}_{\setminus i}, \gamma) = \begin{cases} \frac{n_{jk}}{\sum_k n_{jk}+\gamma}, k \text{ is an existing table,} \\ \frac{\gamma}{\sum_k n_{jk}+\gamma}, k \text{ is a new table.} \end{cases} \quad (1)$$

where $\mathbf{z}_{\setminus i}$ is the seating arrangement of the current $i-1$ customers, and n_{jk} is the number of customers sitting at table k in restaurant j.

We show how well the relevant models meet the requirements necessary for our purpose in Table 1. As shown in this table, our proposal meets the requirements in that 1) each tree represents topic specificity, TCT, and TCF represents the temporality of topics with TCT according to continuous time instead of topic evolution, 2) each document retains thematic coherence between topics via TDP and each document specific tree as an internal structure, and 3) TCF identifies unique topics and exhibits the time-varying topic hierarchies by using birth/death/recurrence of topics via TCT.

3 TOPIC CHRONICLE FOREST, AND THEMATIC DIRICHLET PROCESS

3.1 Our motivations and Basic ideas

Why we need to consider both the specificity and temporality of topics: As shown in Figure 1, a set of timestamped data may include specific topics, temporal topics, and parent-child relationships between these topics. Since not only temporal topics but also the parent-child hierarchy could change over time, we need a set of time-specific topic trees rather than one static tree. As you can imagine, the latest IT tweets share more words with previous IT tweets than with the latest entertainment tweets, and form "IT" topics, where hot "IT" topics change over time. It is reasonable that "Apple" is associated with time-invariant "IT" topics, while "Taylor Swift" and "Pharrell Williams" are associated with hot "IT" topics assigned to "Apple", and they peak at different times. This leads us to consider that "Taylor Swift" and "Pharrell Williams" are associated with specificity and temporal topics of "IT" topics with "Apple", and thus differentiate specificity from temporality in organizing topics. Whether a certain word is a specific term or temporal term depends on the context so it can not be uniquely defined, making it difficult to include it in the dictionary beforehand. For example, although "Taylor Swift" is a specific term in a certain period, this word is likely to become a temporal term over the whole period of IT related topics. Furthermore, the period during which this change is observed varies from topic to topic, and then it is not clear how to decide the appropriate size of time window with this change automatically and discretize the timeline into periods. Our challenge is to detect the appropriate variable-size periods, capture these differences, and organize them.

How to define a hierarchical topic tree covering both specificity and temporality: Topic chronicle forest (TCF) constructs topic trees as a set of topic chronicle trees (TCTs). This structure aims to represent each document as a set of topics that are arranged over a tree-like structure whose semantics are contained in parent-child topic hierarchies, the former are more general than the latter. These hierarchies are consistent with our intuition that the root level topic is the background topic, and the child level topics are more specific or temporal than their parent level topic, and meet our requirements more thoroughly than undirected graphical models [12]; previous topic trees are founded on just one side of these aspects. In the previous example, "Taylor Swift" and "Pharrell Williams" appear in different TCTs, because they are tagged as specificity topics that peak at different times. This is why each time period needs its own TCT.

How to obtain thematic coherence with a nonparametric approach: Figure 2 shows how TCF and Thematic Dirichlet Process (TDP) employ a nonparametric approach to determine the appropriate topic tree automatically. To represent documents using the hierarchy of topics, TDP constructs a tree to reveal the thematic coherence, where this tree has a distribution over paths in a globally shared tree. To this end, TDP employs each document specific tree as an internal structure instead of a path for each document in it, so

Table 1: Only TCF/TDP meets all the requirements for topic discovery and tracking in temporal/hierarchical topic models. In this table, S, T, and P denote semantic, thematic, and periodicity of topics, respectively.

	Requirements Reference	DTM [8]	EvoHDP [35]	RCRP+LDA [1]	iDTM [3]	sTOT [14]	nCRP [7]	rCRP [16]	nHDP [26]	TCF/TDP [this paper]
S	specificity	-	-	-	-	-	√	√	√	√
	temporality	√	√	√	√	√	-	-	-	√
T	topic structure	-	-	-	-	-	√	√	√	√
	internal structure	-	-	-	-	-	-	-	√	√
	topic access	node	node	node	node	node	path	path	tree	tree
P	birth/death topics	-	√	√	√	√	-	-	-	√
	recurrent topics	-	√	√	√	-	-	-	-	√
handling continuous time		-	-	-	-	√	-	-	-	√

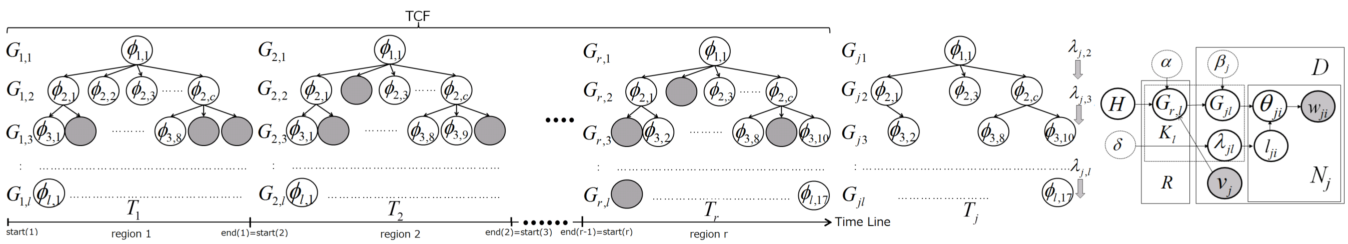

Figure 2: An example of (left) TCF (a set of TCTs, time region r specific T_r), and (center) the j-th document specific topic tree, T_j: Each node of the tree is associated with a topic, ϕ, and shaded notes are inactive in each region. Each TCT is uniquely identified by v_j with the terminating conditions of start(r) $\leq v_j <$ end(r). (right) Graphical model of TDP using TCF: Each latent variable, observed datum, and parameter are shown as nodes. An arrow indicates a conditional dependency between variables and stacked panes indicate a repeated sampling with the iteration number shown. Shaded and unshaded variables indicate observed and latent variables, respectively. Random variables R, L and K_l are finally defined as constants in the experiments/applications.

Table 2: Notations used in this paper

SYMBOL	DESCRIPTION
$D/W/R$	♯documents/♯vocabulary size/♯regions
L/K_l	♯topic levels/♯topics in l-th topic level
N_j/L_j	♯words/♯topic level in j-th document
z_{ji}/l_{ji}	the i-th topic/level variable in j-th document
w_{ji}/θ_{ji}	the i-th word/topic in j-th document
v_j	timestamp in j-th document
λ_{jl}	the l-th level random variable in j-th document
$\phi_{l,c}(\phi_k)$	the c-th topic in l-th level (shorthand of $\phi_{l,c}$)
$\alpha/\delta/\eta/\delta$	parameter: $\alpha/\delta/\eta/\delta \sim Gamma(a_1, a_2)$
β_j	the j specific parameter: $\beta_{jl} \sim Gamma(a_1, a_2)$
κ_r	the r specific parameter: $\kappa_r \sim Gamma(a_1, a_2)$
ϖ_r	the weight of dependency between the current region tree and the previous region tree

that each document can access the entire TCT; nCRP permits each word to follow its path embedded in this tree to a topic. Although rCRP enables a document to have a distribution over the entire topic tree and sample a topic at the word level, they have similar topics in many nodes of the tree, and thus fail to ensure thematic coherency. This leads TDP to share a topic distribution as a base for a second level DP for each document like nHDP, whereas HDP shares a base distribution among documents. Detecting and expressing birth/death/recurrent topics require the ability to identify the same topics as one topic. This is the reason why our approach learns TCF and TDP together; learning them individually would fail to identify topics.

3.2 Model Definition

3.2.1 Topic Chronicle Forest and Topic Chronicle Tree. Figure 2 shows an example of a topic chronicle Forest, T, which is a set of topic chronicle trees, $T_r (r \in R)$. Each T_r is constructed for a given time region (time window), r, that has a start timestamp, $start(r)$, and an end timestamp, $end(r)$, where the #TCTs equal #regions, R. We denote the DP of T_r as G_r, and represent the l-th level of G_r as $G_{r,l}$. The first level of the tree consists of only the background topic. The l-th level could have countably infinite number of children nodes, where the probability of transitioning to child node c, topic $\phi_{l,c}$, equals the probability of the c-th break of a stick-breaking construction [30]. As this construction is applied to each of these stick segments at the next level, G_r achieves coherence between topics on a path from the root to a leaf of the tree. Note that each $\phi_{l,c}$ is constant over time and is reused in other TCTs. As shown in Figure 2, TCF tells us which topics occur or disappear in each period by comparing adjacent regions. For example, $\phi_{2,2}$ is the missing topic in region 2, while $\phi_{3,9}$ is the emerging topic in the same region. This leads TCF to identify periodic topics as the same topic, which reduces the total number of topics, as a single topic cannot appear as different topics in different TCTs. To maintain the information about which topics survive or detect new topics, TCF incorporates time dependencies between adjacent topic chronicle trees.

3.2.2 Document-specific Topic Tree. TDP allows each document to have a document-specific tree, T_j, that shares the set of topics and the parent-children topic hierarchy over all documents via TCF. As shown in Figure 2, TDP selects T_r, when v_j is in region r, and uses each DP of $G_{r,l}$ as a base for a second level DP drawn independently for constructing the topic distribution of l-th level of T_j, G_{jl}. This tree contains all thematic content of a document, and allows TDP to select a thematically consistent topic path at the word level rather than the document level. For selecting this word-specific path, TDP introduces a document-specific random variable, λ_{jl}, that acts as stochastic switch; given a word at token i in document j, it decides the probability that determines whether this word uses the current topic level l in a given topic tree, or continues on down the tree. Thus, G_{jl} has the same topic as $G_{r,l}$ on the same topic level, l, in region r, but with different probability weights and topic level.

Note that TDP uses this distribution in constructing the document specific topic tree, while the nested Hierarchical Dirichlet process (nHDP) [26] extends nCRP to select a topic in a given document-specific topic tree, and then uses it on this constructed tree. This difference allows TDP to reflect the parent-child relationships more precisely than nHDP, and prevents similar topics from appearing in the global tree. Even if the transition probability from one parent topic to its child topic is the highest, T_j, nHDP might miss this relationship if the transition probability to this parent topic is lower than the others. TDP can share topics with TCT, where the probability weight on not only each topic, but also their topic level, varies with the document.

This relationship constrains the increment in the number of topics and so prevents the global topic tree from generating many similar topics that will be observed in the nCRP. Therefore, the probability of a topic level will vary with the document, even if different documents share the same topics.

3.3 Generative Process of TCF and TDP

With reference in the graphic model shown in Figure 2 and notations shown in Table 2, we can provide an overview of the generative procedure of TCF/TDP.

Topic Chronicle Forest generation: For each region $r \in R$, define G_1 using nCRP if $r = 1$; otherwise define G_r using both the previous measure G_{r-1} and DP:

$$G_{r,1} = \delta_{\phi_{1,1}}, V_{l,c} \sim GEM(\alpha), \phi_{l,c} \sim H,$$
$$G_{r,l} \begin{cases} = \sum_{c=1}^{\infty} V_{l,c}\delta_{\phi_{l,c}}, r = 1 \\ \sim DP(\kappa_r, \varpi_r G_{r-1,l} + (1-\varpi_r)H), r > 1 \end{cases} \tag{2}$$

where $GEM(\alpha)$ refers to such a process: $\tilde{V}_{l,c} \sim Beta(1,\alpha), V_{l,c} = \tilde{V}_{l,c}\prod_{h=1}^{c-1}(1-\tilde{V}_{l,h})$, $\delta_{\phi_{l,c}}$ is a point measure concentrated at $\phi_{l,c}$, and the base distribution H is a symmetric Dirichlet over the vocabulary, i.e., $\phi_{\phi_{l,c}}$ are drawn independently. The sum of $V_{l,c}$ equals the stick length of their parent, as the length of the stick is divided up into an infinite number of pieces at its children topic level. Both r(start(r) and end(r)), and the set of r, R, are defined to suit the purpose.

Document-specific Topic Tree generation using TDP: For each document j (j = 1 to D),
1) Select T_r on the condition that $start(r) \le v_j < end(r)$.
2) For each level l of DP in T_r, define G_{jl}:

$$G_{jl} \sim DP(\beta_j, G_{r,l}). \tag{3}$$

3) For each level l in T_j, define $\tilde{\lambda}_{jl}$:

$$\tilde{\lambda}_{jl} \sim Beta(\delta_1,\delta_2), \lambda_{jl} = \tilde{\lambda}_{jl}\prod_{h=0}^{l-1}(1-\tilde{\lambda}_{jh}). \tag{4}$$

To avoid bias toward to a specific level, we update these parameters using both the number of tokens assigned l in j and method-of-moments estimates of the parameters. **Document generation from Document-specific Topic Tree:** For the i-th (i = 1 to N_j) token in the j-th document,
1) Draw topic level l_{ji}: $l_{ji} \sim \lambda_{jl}$.
2) Draw topic θ_{ji}: $\theta_{ji} \sim G_{jl}$ if $l_{ji} = l$.
3) Draw word w_{ji}: $w_{ji} \sim Multinomial(\phi_k)$ if $\theta_{ji} = \phi_k$.

3.4 Posterior distribution of TCTs

We propose a Gibbs sampler for the posterior sampling of table and dish assignments in TCF/TDP. By employing a Pólya urn scheme [6] based on the marginalization of unknown infinite-dimensions [20], we gain conditional distributions by marginalizing out factors ϕ.
Topic Chronicle Forest: We use $m^r_{l,c}$ to count the #customers sitting at the table serving dish $\phi_{l,c}$ in region r, see Figure 2. Likewise, $M^r_{l,c}$ is the number of customers that are descendants of $\phi_{\acute{l},c}$ ($l < \acute{l}$) including $\phi_{l+1,c,*}$ itself. After observing draws in $T_r (r = 1)$, the customer follows one path down T_r, selects the table serving ϕ according to the following distributions using Eq (1):

$$P_r(\phi|\mathbf{S}^r) = \begin{cases} \frac{m^r_{1,1}}{M^r+\alpha}, \phi \text{ is a background topic } \phi_{1,1} \\ \frac{m^r_{l,c}}{m^r_{l,\cdot}+\alpha}, \phi \text{ is an existing topic } \phi_{l,c} \\ \frac{\alpha}{m^r_{l,\cdot}+\alpha}, \phi \text{ is a new topic } \phi_{l,c} \\ \frac{M^r_{l,c}}{m^r_{l,\cdot}+\alpha}, \phi \text{ is an existing topic } \phi_{\acute{l},c,*} \end{cases} \tag{5}$$

where $\mathbf{S}^r$ is the previous assignment in T_r, $m^r_{l,\cdot} = \sum_{c=1}^{C_l} m^r_{l,c}$, $M^r = \sum_{l=1}^{L}\sum_{c=1}^{C_{l-1}} M^r_{l,c}$.

Document specific topic tree: As illustrated in Figure 2, the i-th customer chooses an existing table with a probability proportional to the number of customers sitting at the table and shares the same topic in each document j, or selects a new table with probability β and orders a topic from G_r. This process continues until a full path is defined by descending tree T_r, which yields the construction of T_j. Combining Eq (5) and λ_{jl}, we denote $m^r_{l,c}$, ($m^r_{l,\cdot}$) in T_j by $m^j_{l,c}$ ($m^j_{l,\cdot}$) as document j-specific quantities, and gain the conditional distribution of ϕ in T_j: $P_j(\phi_k|\mathbf{S}_j) =$

$$\begin{cases} \lambda_{j1}, \text{if } \phi_k \text{ is background topic } \phi_{1,1}, \\ \lambda_{jl}\sum_{h=1}^{l-1}(1-\lambda_{jh})(\frac{m^j_{l,c}}{m^j_{l,\cdot}+\beta_j} + \frac{\beta_j}{m^j_{l,\cdot}+\beta_j}P_r(\phi|\mathbf{S}^r)), \text{otherwise}, \end{cases} \tag{6}$$

where $\mathbf{S}_j$ is the previous assignment in T_j. G_{jl} inherits the topics from $G_{r,l}$, but assigns to them document-specific weight $\lambda_{jl}\sum_{h=1}^{l-1}$ $(1-\lambda_{jh})$ at each topic level.

As w_{ji} is drawn from ϕ_k, the conditional density of w_{ji}, $f(w_{ji}|\mathbf{w}^{-ji}, z_{ji} = k)$, is gained by marginalizing out ϕ_k as

$$f(w_{ji}|\mathbf{w}^{-ji}, z_{ji} = k) = \frac{n_{kw}^{-ji} + \eta_{kw}}{\sum_w^W (n_{kw}^{-ji} + \eta_{kw})}, \tag{7}$$

where $\mathbf{w}^{-ji}$ indicates $(w_{11}, \cdots, w_{ji-1}, w_{ji+1}, \cdots, w_{DN_D})$,$\eta_k$ (η_{kw}) is the ϕ_k (word w in k) specific parameter, n_{kw}^{-ji} represents the number of w assigned to ϕ_k, except token w_{ji}.

Algorithm 1 Inference for TCF, and TDP

1: // initialization
2: zero all count variables, $m^r_{l,c}, m^r_{l,\cdot}, m^j_{l,c}, m^j_{l,\cdot}$, and set K_l
3: **for** $j = 1$ to D **do**
4: **for** $i = 1$ to N_j **do**
5: define region r_j according to v_j
6: sample topic index $z_{ji} = k \sim Multinomial(1/K)$
7: **if** $r_j = r$ **then**
8: $m^r_{l,c}$ += 1; $m^r_{l,\cdot}$ += 1; $m^j_{l,c}$ += 1; $m^j_{l,\cdot}$ += 1
9: **end if**
10: **end for**
11: **end for**
12: // Gibbs sampling over burn-in and sampling period
13: **for** iteration=1 to $N_{iteration}$ **do**
14: **for** $j = 1$ to D **do**
15: **for** $i = 1$ to N_j **do**
16: **if** $r_j = r$ **then**
17: $m^r_{l,c}$ -= 1; $m^r_{l,\cdot}$ -= 1; $m^j_{l,c}$ -= 1; $m^j_{j,p,\cdot}$ -= 1
18: **end if**
19: draw ϕ_k using Eq (8).
20: **if** $r_j = r$ **then**
21: $m^r_{l,c}$ += 1; $m^r_{l,\cdot}$ += 1; $m^j_{l,c}$ += 1; $m^j_{l,\cdot}$ += 1
22: **end if**
23: **end for**
24: update $\beta_{j*}, \tilde{\lambda}_{j*}$
25: **end for**
26: update $\alpha, \eta_{1,2}, \delta_{1,2}$
27: **for** $k = 1$ to K **do**
28: // Merge topic phase
29: **end for**
30: Call Region() (Alg 2)
31: **end for**

4 INFERENCE IN TCF/TDP

4.1 Inference algorithm of TDP

Since multinomials and distributions are marginalized out, we obtain the conditional distributions which means that a Gibbs sampler can train both TCF and TDP. Details of the inference for TCF and TDP are shown in Algorithm (1). In practice, we define $m^s_{l,c}$, $m^s_{l,\cdot}$, $M^s_{l,\cdot}$, and M^s to count the number of customers observed in each segment, s, and time-zone of s, t_s, where $m^r_{l,c} = \sum_{start(r) \le t_s < end(r)} m^s_{l,c}$. This introduction enables us to reduce

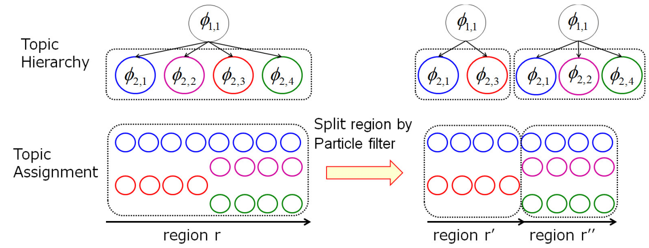

Figure 3: In this example, a l-th topic assignment, under the same parent topic located on $l-1$-th level of TCT, is sliced into two assignments on the same topic level.

the cost of calculation, when the boundaries of regions happen in this inference.

At each iteration, the sampler decreases the topic assignment of the topic tree (line 17), and excludes the current customer eating w_{ji} from a table serving w_{ji} in the restaurant specified by the topic indicator k and previous word sequence u by using AddCustomer. Our sampling strategy for given token i in document j is to propose the topic assignment z_{ji}, and the seating arrangement, $S_{\backslash ji}$. It follows that we need the conditional distribution of all other given topic assignments, which is the set of seating arrangements with a customer corresponding to w_{ji}, as

$$P(z_{ji} = k|\mathbf{w}^{-ji}, S_{j\backslash ji}) \propto P_j(\phi_k|S_{\backslash ji})P(w_{ji}|\mathbf{w}^{-ji}, z_{ji} = k), \tag{8}$$

where $S_{j\backslash ji}$ denotes S_j excluding z_{ji}, $P_j(\phi_k|S_{\backslash ji})$ is given by Eq (6), and $P(w_{ji}|\mathbf{w}^{-ji}, z_{ji} = k)$ is given by Eq (7). After drawing a proposal for z_{ji}, the sampler adds a customer eating w_{ji} to a token serving w_{ji} in the restaurant specified by z_{ji}, and updates the topic assignment of TCF (line 30).

4.2 Inference algorithm of TCF

To detect the boundaries (start(r), end(r)) of region r, TCF judges the topic assignment change that the child topics assignment would persist or change over time in each topic level, where a time line consists of observed timestamps $t_1, t_2, \cdots, t_d$ in a given corpus. Here, we introduce a segment (e.g., one day, one week) as the minimum interval of a given time line. Note that discretizing the time line into bins is not practical as it is difficult to choose the optimal arrangement in advance, and global topic assignments cannot change drastically in a short interval of time. This is why TCF discretizes the time line into segments, and represents each region as a concatenation of segments. Region detection is performed over child topics of the same parent topic, since they came from their parent. As the specific or temporal topics tend to be placed on the lowest level of each tree, we decide the region on only the lowest level of TCF, and synchronize both the start and end of each region with a minimal set of segments among all possible partitions over segmentations of a given time line.

Let $b_{s,l,p}$ be a binary variable that represents whether a change in topic p's child topic assignment occurred on l-th level at s. $P_{s,l,p}(\phi)$ is topic p's child topic distribution on the l-th level from s to $s+1$, and can be calculated by using Eq (5), and $m^s_{l,c}(m^s_{l,\cdot}, M^s_{l,\cdot}, M^s)$ instead of $m^r_{l,c}(m^r_{l,\cdot}, M^r_{l,\cdot}, M^r)$. Here, the number of tables is known and fixed, as total topic assignment is given in this stage.

$$\begin{aligned}
&P_{s,l,p}(\phi_k|S_{s,l,p}, b_{s,l,p}) = \begin{cases} P_{s-1,l,p}(\phi_k|S_{s-1,l,p}), \text{if } b_{s,l,p} = 0, \\ P_{s,l-1,p}(\phi_k|S_{s,l-1,p}), \text{if } b_{s,l,p} = 1, \end{cases} \\
&P_{s-1,l,p}(\phi_k|S_{s-1,l,p}) = P_{r,l,p}(\phi_k|S^r), \text{if start(r)} \le t_{s-1} < end(r), \\
&b_{s,l,p} \sim Bernoulli(\zeta), \zeta \sim Beta(\epsilon_1, \epsilon_2),
\end{aligned} \tag{9}$$

where $S_{s,l,p}$ denotes topic p's child topic arrangement on the l-th level in s, and region r specific arrangement equals to the previous region arrangement, and $P_{r,l,p}(\phi_k|S^r)$ is the l-th level sub-tree of $P_r(\phi_k|S^r)$ over children topics of topic p. That is, $b_{s,l,p} = 0$ means there was no change on the l-th level of p at s ($P_{s,l,p}(\phi_k|S_{s,l,p}) = P_{s-1,l,p}(\phi_k|S_{s-1,l,p})$), and $b_{s,l,p} = 1$ means there was a change on the l-th level of p at s ($P_{s,l,p}(\phi_k|S_{s,l,p})$=$P_{s,l-1,p}(\phi_k|S_{s,l-1,p})$). Figure 3 shows how appropriate TCTs can be obtained by detecting change points b_s over segments, and redefining the boundaries of regions and the corresponding topic assignments. After this synchronization, we recalculate $P_{s,l,p}(\phi)$ and construct the TCT in this region.

As TCF requires detection of the intermittent changes of hidden states, $b_{s,l,p}$, we employ particle filters, that is a family of sequential Monte Carlo sampling algorithms [10], to estimate the posterior distribution of these change points. Each particle cloud is updated recursively for each new observation using importance sampling, and is used for approximating a probability distribution over a latent variable. The particle filter for TCF is outlined in Algorithm 2, where P particles are samples of change points vector $b_s^{(p)}$. At the moment topic assignment is completed, particles form a discrete approximation of the posterior up to the previous segment. Each particle p is propagated forward by drawing change variable $b_{s,l,p}^{(p)}$ from the conditional posterior distribution $P(b_{s,l,p}^{(p)}|\mathbf{S}_{s,l,p}, b_{s-1,l,p}^{(p)})$ and scaling the particle weight by $P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)})$. The conditional posterior used in this propagation step is given by

$$\begin{aligned}
&P(b_{s,l,p}^{(p)}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}) \propto \\
&P(\mathbf{S}_{s,l,p}, b_{s,l,p}^{(p)}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}) = \\
&P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s,l,p}^{(p)}, b_{s-1,l,p}^{(p)})P(b_{s,l,p}^{(p)}|b_{s-1,l,p}^{(p)}) = \\
&\begin{cases} P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}, b_{s,l,p}^{(p)} = 0)P(b_{s,l,p}^{(p)} = 0|b_{s-1,l,p}^{(p)}), \\ P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}, b_{s,l,p}^{(p)} = 1)P(b_{s,l,p}^{(p)} = 1|b_{s-1,l,p}^{(p)}). \end{cases}
\end{aligned} \tag{10}$$

The first term is the likelihood of $\mathbf{S}_{s,l,p}$ when $\mathbf{S}_{s-1,l,p}$ has been fixed, and can be obtained from Eq (9) by

$$\begin{aligned}
&P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}, b_{s,l,p}^{(p)}) \\
&\propto \prod_{c \in C_{l,p}} \begin{cases} \Gamma(m_{l,c}^s + m_{l,c}^{s-1}), \text{if } b_{s,l,p}^{(p)} = 0, \\ \Gamma(m_{l,c}^s + m_{l-1,c}^s), \text{if } b_{s,l,p}^{(p)} = 1, \end{cases}
\end{aligned} \tag{11}$$

where Γ is the gamma function.

As a record of the history of $b_{s-1,l,p}^{(p)}$ over segments, the conditional distribution of $b_{s,l,p}^{(p)}$ is given from Eq (9) by

$$\begin{aligned}
P(b_{s,l,p}^{(p)} = 0|b_{s-1,l,p}^{(p)}) &= \frac{\epsilon_1 + n_{s-1,l,p}^{(p)}(0)}{\epsilon_1 + \epsilon_2 + n_{s-1,l,p}^{(p)}(0) + n_{s-1,l,p}^{(p)}(1)}, \\
P(b_{s,l,p}^{(p)} = 1|b_{s-1,l,p}^{(p)}) &= \frac{\epsilon_2 + n_{s-1,l,p}^{(p)}(1)}{\epsilon_1 + \epsilon_2 + n_{s-1,l,p}^{(p)}(0) + n_{s-1,l,p}^{(p)}(1)},
\end{aligned} \tag{12}$$

where $n_{s-1,l,p}^{(p)}(0)(n_{s-1,l,p}^{(p)}(1))$ denotes the number of topic p's children topics on the l-th level of continue (change) at s.

The particle weights are scaled by

$$\frac{\omega_{s,l,p}^{(p)}}{\omega_{s-1,l,p}^{(p)}} \propto P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s-1,l,p}^{(p)}). \tag{13}$$

These weights are normalized so that they sum to 1, and approximate the posterior distribution over change points as

$$P(b_{s,l,p}|S_{s,l,p}) \approx \sum_{p=1}^{P} w_{s,l,p}^{(p)} \delta_{b_{s,l,p}}(b_{s,l,p}^{(p)}), \tag{14}$$

where $S_{s,l,p}$ is the seating arrangement in s, and $\delta_y(x)$ is a delta-mass function, which is equal to 1 when $x = y$.

Algorithm 2 Region(): Particle filter for TCF

// initialization
$r = 0$
weights $\omega_0^{(p)} = \frac{1}{P}$ for $p = 1, \cdots, P$
for $s = 1$ to S **do**
  for $p = 1$ to P **do**
    set $\omega_{s,l,p}^{(p)} = \omega_{s-1,l,p}^{(p)} P(\mathbf{S}_{s,l,p}|\mathbf{S}_{s-1,l,p}, b_{s,l,p}^{(p)})$ using Eq (10, 11 and 13)
    sample $b_{s,l,p}^{(p)}$ w.p. $P(b_{s,l,p}^{(p)}|b_{s-1,l,p}^{(p)})$ using Eq (12)
  end for
  normalize $\omega_{s,l,p}^{(p)}$ to sum to 1
  sample $b_{s,l,p}$ w.p. $P(b_{s,l,p}|S_{s,l,p})$ using Eq (14)
  if $b_{s,l,p} = 1$ **then**
    start(r)=t_s, end(r)=t_{s+1}, $r = r + 1$
    $m_{l,c}^r = \Sigma_{t_s \le t(s) < t_{s+1}} m_{l,c}^s$,
  end if
  IF $||w||^{-2} \ll$ESS then resample particles
end for

5 EXPERIMENTS

5.1 Datasets and Experiment design

We conducted both qualitative and quantitative evaluations on the data sets noted in Table 3; we removed the stop words and rare words (appearing less than 10 times in each collection) using NLTK[1]. In experiments, Recurrent CRP [2] + LDA (RCRP+LDA) [1] and

[1]NLTK 3.2.4: http://www.nltk.org

Table 3: Data sets: ACM is a set of papers of 2001-2016 ACM CIKM, SIGIR, KDD, WWW, and WSDM. Twitter is a collection of tweets containing #apple as hashtag from 01/01/2015 to 31/06/2015 via twitter API[3], also shown in Figure 1.

	#documents	#vocabulary	time scale of v_j
ACM	6,255	29,682	publication year
Twitter	85,324	33,477	timestamp

HDP [31]+sTOT [14] are selected as benchmark temporal topic models, and recursive CRP (rCRP) [16], and nested HDP (nHDP) [26] are selected as benchmark hierarchical topic models. As rCRP, nHDP do not include a timestamp variable, we learned them by the year (ACM)/week (Twitter) scale, and compared them with the corresponding region specific TCT of TCF. We follow previous works [33] and set the topic Dirichlet parameter η to 0.1, and made the parameters of DP, α, and the Dirichlet prior, $\boldsymbol{\beta}$, using the Gamma distribution to form sparsity in models, where $\gamma \sim Gamma(0.1, 1)$, $\alpha = \gamma$, and $\boldsymbol{\beta} = \gamma\boldsymbol{\beta}$ [33]. We ran them on Spark[2] on 22 PCs with Dual Core 2.66 GHz Xeon processors.

As TCF is insensitive to the inference setting such as the size of segment (time interval), and #particles, we set #particles to 20 from pre-experiments, and evaluate how they affect the #discovered topics and TCTs extracted from the given data. Since one day is too expensive in terms of computation and not effective as a segment size, we set segment period to 1year (ACM), and 1week (Twitter). From ACM/Twitter, 211/97 topics were gained with 14/22 regions, where the average of time interval, R, was 51/1.1weeks and the average of topic level, L, was 3.2/3.1. These pre-experiments showed that the 1) difference between topics is more sensitive to region interval than the #discovered topics, and 2) the depth of topic level converges at about 3 for all data sets, although it increases in inverse proportion to region interval.

To evaluate whether TCF can discover and monitor topics in a given data and meet the our goals (See Table 1), we designed experiments to answer the following questions:

- Can TCF find the time granularity and handle both the specificity and temporality of topics?: Subsection 5.2.1
- Can TCF maintain thematic coherence, and construct more compact trees than the alternatives? Subsection 5.2.2
- Can TCF discover the periodicity of topics and track trends?: Subsection 5.3

5.2 Quantitative Evaluation

5.2.1 Comparison of specificity and temporality. We use perplexity (test set perplexity), Jensen-Shannon divergence (JS-D) [19] and Normalized Mutual Information (NMI) [21]. Perplexity is widely used to assess the predictive power of models (lower numbers are better) [15, 27, 28] in the NLP community. JS-D measures average word distribution separations between all pairs of different topics. As this allows us to quantify the difference between topics and discuss the effect of the hierarchical topic structure on distinguishing the differences between topics, we use the average JS-D to evaluate the specificity of topics. NMI is always a score between 0 and 1, where 1 means the clustering results exactly match the ground truth class (time scale of v_j) and 0 means the two sets are independent. We use NMI to evaluate the temporality of topics, where the ground truth class is year (ACM)/week (Twitter) scale of each document. This leads us to use the divergence between each document topic distribution and region (TCF)/learn unit(the others) specific topic distribution, and determine the class of each document with the minimum value as the class of cluster. The motivations of this experiment are to confirm that each TCT helps us to infer what happened at the moment in question (e.g "Taylor" in Figure 1) via temporal topics, that time-evolving models can detect trends, and that these words are representative of each trend. In this case, we used 10-fold cross-validation based on partitioning; a single subsample is retained as the validation data for testing the models, and the remaining subsamples are used as training data.

Table 4: Comparison of the specificity and temporality of topics: K(#topics), JS-D, NMI, and TSP (perplexity). JS-D is calculated by comparing 3rd level topics in the adjacent unit or region. As all models use the nonparametric setting, they learn K automatically. K of rCRP, nHDP and TCF is the average for each unit/region. Results that differ significantly, t-test $p < 0.01$, from nHDP are marked with '*'.

		RCRP +LDA	HDP +sTOT	rCRP	nHDP	TCF
ACM	K	227	233	189.3	177.5	125.7
	TSP	1487	1412	1316	1327	1215*
	JS-D	0.22	0.23	0.24	0.27	0.29
	NMI	0.35	0.31	0.45	0.51	0.82*
Twitter	K	137	138	105.7	102.2	72.5
	TSP	1231	1235	1122	1012	905*
	JS-D	0.28	0.27	0.27	0.29	0.44*
	NMI	0.41	0.43	0.52	0.58	0.86*

As shown in Table 4, TDP attains lower perplexity and higher NMI with more distinct topics than the others when identifying low dimensional components. As the trend of ACM changed more slowly than that of Twitter at the given time scale, JS-D over ACM is lower than that over Twitter. These results support our idea that 1) the coherent structure reduces # the number of similar topics and makes each topic clearer than the alternatives do, 2) this structure reduces perplexity, and 3) temporal topic models require a coherent structure. TCF detects the variable-size time granularity as regions, and constructs a time-varying tree that consists of specialized topics as in previous models and maintains the temporality of topics.

5.2.2 Topic coherence of topic trees. This experiment evaluates whether using just three topic levels is enough for distinguishing topics clearly and capturing dynamics, and how well TDP constructs the topic tree using the topic coherence measure $C(k; W^{(k)})$ [22],

$$C(k; W^{(k)}) = \sum_{m=2}^{M} \sum_{n=1}^{m-1} log \frac{D(w_m^{(k)}, w_n^{(k)}) + 1}{D(w_n^{(k)})}, \tag{15}$$

where $D(w)/D(w, w')$ means the document frequency of word w/co-document frequency of word w and w', and $W^{(k)} = (w_1^{(k)}, \cdots, w_M^{(k)})$ is a list of the M most probable words in topic k. as previously proposed metrics [24] are developed on corpora, which are dissimilar to our data sets. We compare TDP with rCRP, and nHDP as they

[2]Apache™Spark®: http://spark.apache.org/
[3]Twitter Developers: https://developer.twitter.com/en/docs

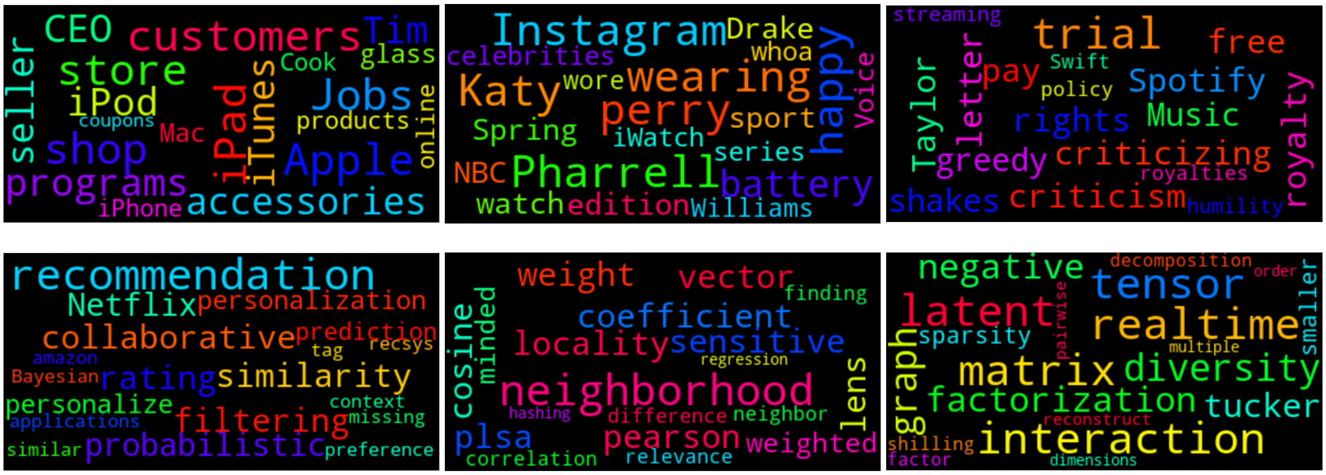

Figure 4: Word Cloud of 2-nd level/the corresponding 3-rd level topic from (upper) Twitter, and (lower) ACM: We list 20 words according to the learned probability from (left column) the 2-nd level topics, and (center and right column) corresponding 3-rd level topics under TCF. 3-rd level topics are sorted in ascending order by the start time of each region. That is, center topic (center column) is the oldest child topic of left topic (left column), and right topic (right column) is the newest child topic of the same left topic.

Table 5: Comparison on #generated topics and topic coherence score in ACM and Twitter: K_l means the average of #topics on l for each unit/region. Co means the topic coherence with M=10 and is the average for each unit/region on the same l. When l=1, this topic is the root node of all models and takes only topic. Results that differ significantly, t-test $p < 0.01$, from nHDP are marked with '*'.

l: topic level		2		3		4	
data		K_l	Co	K_l	Co	K_l	Co
ACM	rCRP	27.9	-146	102.7	-128	57.7	-117
	nHDP	29.3	-142	108.9	-121	38.3	-106
	TCF	35.3	-122*	80.6	-105*	8.8	-88*
Twitter	rCRP	27.3	-122	51.2	-118	26.2	-113
	nHDP	26.1	-112	50.1	-107	25.0	-98
	TCF	21.2	-108	44.2	-99*	6.1	-91*

are hierarchical topic models known to maintain topic coherence; the iterations of Gibbs sampler of all models was set to 2000, where the first 200 iterations were used to burn in the sampler.

Table 5 shows that TCF generated the fewest topics among all models over both data sets, and its topics became more specialized with the depth of topic level, l. As rCRP and nHDP create more topics than TCF does on the same level, their coherence score is lower than that of TDP. Although TCF yields fewer topics than the others on the same level, TCF has higher coherence score. This means that similar topics could appear in rCRP and nHDP, which lowers their scores relative to TCF, and high topic levels are not always necessary for building meaningful topic trees. Indeed, the 4-th level topics of TCF are constrained by the impact of λ_j and are much fewer in number than the other levels. That is, TCF offers a more parsimonious structure than the alternatives while matching their explanatory power.

5.3 Qualitative Evaluation

Figure 4 provides an example of the words associated with the topics learned by TCF/TDP from (upper) Twitter, and (lower) ACM. As we can see, the 2-nd level topic (left) of the first row is related to "Apple products (IT)", see Figure 1, and each 3-rd level topic (center/right) corresponds to temporal events in their parent (2-nd level) topic. For example, the number of retweets covering #apple: "Taylor Swift" indicated she became an extra hot topic after her public criticism of the royalties issue with "Apple Music service" in Jun 2015, while "Pharrell Williams" was a hot topic after wearing "Apple watch" on NBC's 'The Voice' in March 2015. This period matches the region of the 3-rd level topic in this table ($start(r) \leq v_j < end(r)$), and the corresponding words appear more intensively in this topic than the others. In the second row, the left topic is related to papers in "recommendation", center (right) topic corresponds "older topic" ("newer topic") in "recommendation". As this center topic is observed in the other regions and is not a burst topic, TCF can detect the periodicity of topics and trends in "recommendation". As another example, a topic associated with "Deep learning" appears as 2-nd level newer TCTs in TCF. That is, TCF and TDP address both the specificity and temporality of topics, and so are useful for retrospective discovery of topics; they allow trends to be tracked from timestamped corpora.

6 DISCUSSION

The advantage of TCF/TDP lies in its usage of time-varying topic structure to reduce computational cost. Pre-experiments showed that the average topic level of TCF is under 4; the alternatives yielded deeper structures. TCF generates a lot fewer 4-th level topics than the alternatives, although the appropriate level is indirectly influenced by λ_j. This is why the #topics of TCF decrease at the 4-th level in Table 5. Figure 4 shows trends similar to the results of Google trends or Scholar[4]. Since TDP yields simplified structures that constrain the possible topic assignment of each token, it achieves lower perplexity with fewer sampling iterations (more quickly) and fewer topics than the alternatives.

7 CONCLUSION

To monitor dynamics in time-stamped data, this paper presented a topic model that addresses both the specificity and temporality of topics. Our approach tackles three of the major tasks in [5]: (1) segmentation tasks are done by splitting time intervals into regions, (2) detection tasks are achieved by using the periodicity of topics in TCF, and (3) tracking tasks are performed by locating each document in each region and assigning each word a topic via the corresponding TCT. Experiments on various data sets showed that the proposed model yields time-varying topic hierarchies, and helps us comprehend time-stamped corpora. We will extend TCF/TDP for application to streaming data and thus permit the monitoring of trends in real time.

REFERENCES

[1] Amr Ahmed, Qirong Ho, Jacob Eisenstein, Eric Xing, Alexander J. Smola, and Choon Hui Teo. 2011. Unified Analysis of Streaming News. In *WWW*. 267–276.

[2] Amr Ahmed and Eric P. Xing. 2008. Dynamic Non-Parametric Mixture Models and the Recurrent Chinese Restaurant Process: with Applications to Evolutionary Clustering. In *SIAM*. 219–230.

[3] Amr Ahmed and Eric P. Xing. 2010. Timeline: A Dynamic Hierarchical Dirichlet Process Model for Recovering Birth/Death and Evolution of Topics in Text Stream. In *UAI*. 20–29.

[4] D. Aldous. 1985. *Exchangeability and related topics*. Lecture Notes in Math, Vol. 1117. Springer, Berlin. 1–198 pages.

[5] James Allan. 2002. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publishers, Norwell, MA, USA.

[6] David Blackwell and James B. MacQueen. 1973. Ferguson Distributions Via Polya Urn Schemes. *The Annals of Statistics* 1, 2 (Mar 1973), 353–355.

[7] David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. 2004. Hierarchical topic models and the nested Chinese restaurant process. *NIPS* 16 (2004).

[8] David M. Blei and John D. Lafferty. 2006. Dynamic Topic Models. In *ICML*. 113–120.

[9] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *JMLR* 3 (2003), 993–1022.

[10] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. 2000. On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. *STATISTICS AND COMPUTING* (2000), 197–208.

[11] Thomas S. Ferguson. 1973. A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics* 1, 2 (Mar 1973), 209–230.

[12] Zhe Gan, Changyou Chen, Ricardo Henao, David E. Carlson, and Lawrence Carin. 2015. Scalable Deep Poisson Factor Analysis for Topic Modeling. In *ICML*. 1823–1832.

[13] Janani Kalyanam, Amin Mantrach, Diego Saez-Trumper, Hossein Vahabi, and Gert Lanckriet. 2015. Leveraging Social Context for Modeling Topic Evolution. In *KDD*. 517–526.

[14] N. Kawamae. 2011. Trend Analysis Model: Trend Consists of Temporal Words, Topics, and Timestamps. In *WSDM*. 317–326.

[15] N. Kawamae. 2014. Supervised N-gram topic model. In *WSDM*. 473–482.

[16] Joon Hee Kim, Dongwoo Kim, Suin Kim, and Alice Oh. 2012. Modeling topic hierarchies with the recursive chinese restaurant process. In *CIKM*. 783–792.

[17] Daniel D. Lee and H. Sebastian Seung. 1999. Learning the parts of objects by nonnegative matrix factorization. *Nature* 401, 6755 (1999), 788–791.

[18] W. Li, D. Blei, and A. McCallum. 2007. Nonparametric Bayes Pachinko allocation. In *UAI*. 243–250.

[19] J. Lin. 2006. Divergence Measures Based on the Shannon Entropy. *IEEE Trans. Inf. Theor.* 37 (2006), 145–151.

[20] S.N. MacEachern. 1994. Estimating normal means with a conjugate style Dirichlet process prio. *Communications in Statistics-Simulations* 23 (1994), 727–741.

[21] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York.

[22] David Mimno, Hanna M. Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2010. Optimizing Semantic Coherence in Topic Models. In *EMNLP*. 262–272.

[23] Radford M. Neal. 2000. Markov chain sampling methods for Dirichlet process mixture models. *JCGS* 9, 2 (2000), 249–265.

[24] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic Evaluation of Topic Coherence. In *HLT*. 100–108.

[25] Viet-An Nguyen, Jordan Boyd-Graber, and Philip Resnik. 2013. Lexical and Hierarchical Topic Regression. In *NIPS*.

[26] John Paisley, Chong Wang, David M. Blei, and Michael I. Jordan. 2014. Nested hierarchical Dirichlet processes. *TPAMI* (April 2014).

[27] Michael J Paul and Mark Dredze. 2015. SPRITE: Generalizing Topic Models with Structured Priors. *Transactions of the Association for Computational Linguistics (TACL)* 3 (2015), 43–57.

[28] Md Mustafizur Rahman and Hongning Wang. 2016. Hidden Topic Sentiment Model. In *WWW*. 155–165.

[29] Ankan Saha and Vikas Sindhwani. 2012. Learning Evolving and Emerging Topics in Social Media: A Dynamic NMF approach with Temporal Regularization. In *WSDM*. 693–702.

[30] Jayaram Sethuraman. 1994. A constructive definition of Dirichlet priors. *Statistica Sinica* 4 (1994), 639–650.

[31] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. 2006. Hierarchical Dirichlet Processes. *JASA* 101, 476 (2006), 1566–1581.

[32] Carmen K. Vaca, Amin Mantrach, Alejandro Jaimes, and Marco Saerens. 2014. A Time-based Collective Factorization for Topic Discovery and Monitoring in News. In *WWW*. 527–538.

[33] H. Wallach, D. Mimno, and A. McCallum. 2009. Rethinking LDA: Why Priors Matter. In *NIPS*. 859–872.

[34] X. Wang and A. McCallum. 2006. Topics over time: a non-Markov continuous-time model of topical trends.. In *KDD*. 424–433.

[35] Jianwen Zhang, Yangqiu Song, Changshui Zhang, and Shixia Liu. 2010. Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora. In *KDD*. 1079–1088.

[4] https://trends.google.co.jp/trends/, https://scholar.google.com