# Accounting for Temporal Dynamics in Document Streams

Zhendong Chu
School of Computer Science
Fudan University
Shanghai, China
zdchu15@fudan.edu.cn

Renqin Cai
Department of Computer Science
University of Virginia
Charlottesville, VA, USA
rc7ne@virginia.edu

Hongning Wang
Department of Computer Science
University of Virginia
Charlottesville, VA, USA
hw5x@virginia.edu

## ABSTRACT

Textual information, such as news articles, social media, and online forum discussions, often comes in a form of sequential text streams. Events happening in the real world trigger a set of articles talking about them or related events over a period of time. In the meanwhile, even one event is fading out, another related event could raise public attention. Hence, it is important to leverage the information about how topics influence each other over time to obtain a better understanding and modeling of document streams.

In this paper, we explicitly model mutual influence among topics over time, with the purpose to better understand how events emerge, fade and inherit. We propose a temporal point process model, referred to as Correlated Temporal Topic Model (CoTT), to capture the temporal dynamics in a latent topic space. Our model allows for efficient online inference, scaling to continuous time document streams. Extensive experiments on real-world data reveal the effectiveness of our model in recovering meaningful temporal dependency structure among topics and documents.

## CCS CONCEPTS

• **Computing methodologies** → **Learning in probabilistic graphical models**; • **Information systems** → *Data streams*;

## KEYWORDS

Temporal topic modeling; Hawkes process; Online clustering

**ACM Reference Format:**
Zhendong Chu, Renqin Cai, and Hongning Wang. 2019. Accounting for Temporal Dynamics in Document Streams. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357384.3358022

## 1 INTRODUCTION

Millions of text documents flood on the internet everyday, which makes it impossible for an ordinary user to digest the information buried in these documents effectively. Service providers, such as search engines, social media platforms, and online forums, invest a huge amount of resources in organizing these documents to help their users retrieve and utilize relevant information [2, 3, 8, 17, 20]. How to effectively model and analyze these unstructured document streams become increasingly crucial for service providers to improve users' experience and maximize their service utility.

Document streams are composed of text content generated over time, which, though could be large in volume, are never independent of each other. Both temporal and textual information strongly manifest the underlying dependency structure in document streams. First, it has been verified in various studies [2, 10] that documents appearing in close temporal proximity tend to share the same topic. For example, the occurrence of one event may result in a series of documents discussing about it in a short period. As shown in Figure 1, by manually analyzing a subset of 48,986 news articles from 5 mainstream press (e.g., CNN) published between 01/01/2017 and 07/06/2017, we find that when a shooting incident happened in Chicago on 01/10/2017, a series of news articles about this event were published subsequently in the next 3 hours. Many similar observations are also obtained on other breaking news events in the dataset. Second, the appearance of a topic leads to the emergence of relevant topics in close temporal proximity. As Figure 1 shows, right after the report of *Chicago Shooting*, the discussion about *Gun Control* and *Public Safety & Security* rose simultaneously. Moreover, it is not a surprise to find such a correlation among related topics repeats itself in this dataset. Third, the temporal dynamic patterns are different across topics. The occurrence of certain topics can be transient, while some are periodically popular. For instance, in Figure 1 *President Mourn for Chicago Shooting* raised public's attention to *Chicago Shooting* again, re-triggering a series of news articles about *Gun Control* and *Public Safety & Security*. These two topics which fade out in a period ago regained popularity. These three aspects of the complex temporal dependency in document streams suggest that it is not trivial to capture the embedded temporal dynamics, but necessary when modeling document streams.

Modeling document streams while capturing the aforementioned temporal dependency structures is our focus in this work. Various solutions have been proposed to model document streams. Topics Over Time (TOT) model [20] and Dynamic Topic Model (DTM) [4] are typical solutions. Specifically, TOT samples timestamp for each document based on the document's topic distribution; and DTM assumes at different time periods there are different topic distributions over documents. However, such models treat topics as independent, such that they cannot realize the correlation among topics. In addition, these models assume the popularity of topics is stationary, but words representing topics change over time, which fails to capture the temporal variance of topic popularity.

Some recent developments introduce temporal point process models to capture the dynamics of topics over time. A typical example is the Dirichlet Hawkes Process (DHP) model [10], which


*CIKM '19, November 3–7, 2019, Beijing, China*

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3358022

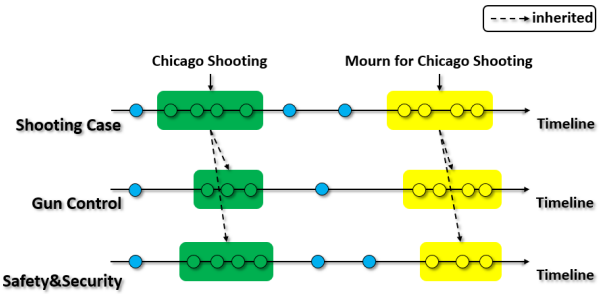



**Figure 1: An illustration of temporal dynamics in CNN newswire. Every circle represents a document published at a particular time on a global timeline. Colored boxes are documents about the same event. Each selected topic is represented by its own timeline. Dotted arrows indicate the influence between different topics within the same event.**

assumes documents in a close temporal proximity share the same topic and topics temporally faraway fade out gradually. However, this simplified assumption ignores the heterogeneity of temporal patterns across topics. In DHP, all topics which remain silent for a period of time are to be abandoned; if a previously fading away topic becomes popular again after a while, it will be treated as a new topic. As a consequence, documents on the same topic but far apart in time will be assigned to different topics, which makes it hard to capture the dependence among documents accurately. In addition, mutual influence among correlated topics is also missing in DHP, as topics are treated independently from each other. Hierarchical Dirichlet Hawkes Process (HDHP) [16] considers the heterogeneity of topics' temporal dynamics by assuming the generative process of documents follow the Hierarchical Dirichlet Process (HDP), instead of the Dirichlet Process (DP). This enables a topic to regain popularity after fading out for a while. But the topics are still assumed to be independent from each other in HDHP.

In this work, we integrate the temporal point process, specifically the Hawkes Process [13], with Hierarichal Dirichlet Process to account for different aspects of temporal dependence in document streams. We name the proposed model as Correlated Temporal Topic Model, or CoTT in short. We assume each document is associated with a topical cluster, such that a document stream is organized into clusters over time; and each document cluster is then associated with a single topic. Different document clusters can share the same topic. This forms a two-level structure of document stream, where we categorize the topical dependence in a document stream into two types: that among documents and that among topics. CoTT regards the dependence among documents as the tendency that documents appearing temporally close to each other tend to be on the same topic. This is achieved by assigning documents temporally close to the same cluster via a uni-dimension Hawkes Process. New clusters are always needed to model the constantly generated documents; and in CoTT, we employ a DP prior distribution over the clusters to adaptively create document clusters on the fly.

On top of the document clusters, CoTT explicitly models the dependence among topics as mutual influence across topics. A multi-dimension Hawkes Process is introduced to model the *mutual-excitement* among topics over the document clusters. Specifically, when deciding the topic assignment for a document cluster, CoTT considers not only the popularity of a topic in existing clusters (i.e., its base intensity), but also the other existing topics' temporal influence (i.e., pairwise mutual influence). The popularity-based base intensity allows a historically popular topic to regain attention, and the temporal mutual influence emphasizes the proximity between the appearances of topics over time. Again, to enable the creation of new topics on the fly, we impose a DP prior on top of this multi-dimension Hawkes Process for modeling the topics.

To investigate the effectiveness of CoTT in modeling temporal dynamics in document streams, we performed extensive experiments on both synthetic data and a large real-world news corpus consisting of news articles from multiple mainstream publishers. Our solution obtained promising performance improvement in modeling unseen documents, predicting future content and appearance time of a target document. In particular, meaningful mutual influence structure and temporal pattern among topics can be automatically recovered, which provide helpful insight in text analytics over document streams.

## 2 PRELIMINARIES

In this section, we introduce the major building blocks of CoTT, i.e., the Hierarchical Dirichlet process [18] and the Hawkes process [1].

### 2.1 Hierarchical Dirichlet Process

To allow instances to share an unbounded number of clusters, Teh et al. [18] proposed Hierarchical Dirichlet Process (HDP) to structure data points in an unsupervised fashion, which imposes a hierarchy of Dirichlet processes (DPs). It utilizes a DP to model clusters $G_0 \sim DP(\beta_0, H)$ and another layer of DP to model groups of instances which share clusters $G \sim DP(\alpha_0, G_0)$. A corresponding perspective of understanding HDP is the Chinese restaurant franchise process:

(a) Draw table $c_i$ for a new customer $i$,
  1) Sample a new table with probability
$$P(c_i = K+1) = \frac{\alpha_0}{\alpha_0 + \sum_k^K n_k}$$
  2) Sample an existing table with probability
$$P(c_i = k) = \frac{n_k}{\alpha_0 + \sum_k^K n_k}$$
  where $n_k = \sum_{j=1}^{i-1} \mathbb{I}(c_j = k)$ is the number of customers seating at the $k$th table.

(b) If $c_i = K+1$, *i.e.*, the new customer sits at a new table,
  1) Sample a new dish for the table with probability
$$P(z_c = L+1) = \frac{\beta_0}{\beta_0 + \sum_l^L m_l}$$
  2) Sample an existing dish for the table with probability
$$P(z_c = l) = \frac{m_l}{\beta_0 + \sum_l^L m_l}$$
  where $m_l = \sum_{k=1}^K \mathbb{I}(z_k = l)$ is the total number of tables serving dish $l$ in the franchise.

Compared with DP, the data clusters are further grouped in HDP. This enables fine grained modeling of data clustering structure. However, HDP still builds upon the exchangeability assumption, and therefore it is not able to model the temporal dynamics among data points, where time is important to be modeled.

## 2.2 Hawkes Process

Hawkes process, a type of temporal point process, models the dependence of future events on historical events with respect to their temporal distance [14]. Intensity function characterizes a Hawkes process, depicting the occurring rate of an event at time $t$ given historical events. For a uni-dimension Hawkes process that models the dependence within the same type of events, the intensity function $\lambda^*(t)$ is defined as,

$$\lambda^*(t) = \mu + \sum_{t_i \in \mathcal{H}(t)} \alpha\kappa(t, t_i) \tag{1}$$

where $\mu$ is the base intensity, representing the spontaneous occurring rate of an event. The kernel function $\kappa(t, t_i)$ captures the influence of previous event $i$ at time $t_i$ on the generation of current event at time $t$, e.g., a time decay function. Exponential function [7] $\kappa(t, t_i) = \exp(-\omega(t - t_i))$ or RBF function [11] $\kappa(t, t_i) = \exp(-(t - t_i - \tau)^2/2\sigma^2)/\sqrt{2\pi\sigma^2}$ are typically used as the kernel function. The parameter $\alpha$ measures the strength of temporal influence. Because of the time decay effect, previous events occurring temporally closer to the current event have stronger influence on it than those temporally farther away.

To capture dependence among events of different types, multi-dimension Hawkes process [15] is introduced. Its intensity function is defined as,

$$\lambda_u^*(t) = \mu_u + \sum_{t_i \in \mathcal{H}(t)} A_{uu_i}\kappa(t, t_i) \tag{2}$$

where $\mu_u$ is the base intensity of event type $u$. A $K$-by-$K$ matrix $A$ denotes the pairwise mutual influence among $K$ different types of events, and $A_{uu_i}$ represents the mutual influence between event type $u$ and event type $u_i$.

In this work, we consider the generation of a document as an event; and a type of events is summarized as a topic, which is a word distribution over a fixed vocabulary. We model the dependence among documents via a uni-dimension Hawkes process so as to segment a document stream into clusters; and impose a multi-dimension Hawkes process over the document clusters to capture the dependence among topics. Hierarchical Dirichlet process is introduced on top to enable dynamic creation of clusters and topics on the fly.

# 3 CORRELATED TEMPORAL TOPIC MODEL

In this section, we describe our developed Correlated Temporal Topic Model (CoTT) in detail, which is designed to capture temporal dependence both among documents and among topics in a document stream. We assume each document belongs to a latent document cluster, which is linked with a latent topic. As a result, in CoTT, we use a two-level structure to organize documents: documents are organized into clusters and clusters are organized into topics. A high-level illustration of CoTT is shown in Figure 2, and we will describe each component of our design in detail.

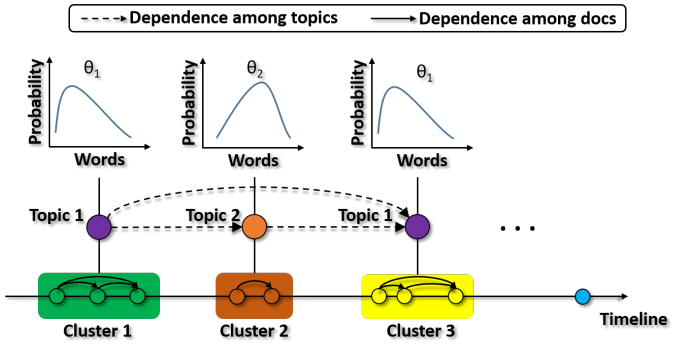



**Figure 2: Illustration of Correlated Temporal Topic Model. Every circle on the timeline represents a newly generated document. Colored boxes are documents belonging to the same cluster. Documents are organized into clusters. Colored circles represent topics, which are shared by different clusters. Documents under the same topic share the same word distribution.**

To facilitate our discussion, we define the following notations. A document stream $D$ is composed of $N$ documents, $D = \{d_i\}_{i=1}^N$, which are ordered chronologically by their timestamps. Each document $d$ is associated with a bag of words and a timestamp when it is published. We use $W_i$ to represent words in document $d_i$ and $t_i$ to represent its timestamp. Accordingly, $c_i^l$ denotes that $d_i$ belongs to document cluster $l$ and $z_{c_i}^k$ denotes that $d_i$ belongs to topic $k$. We use $z_l^k$ to denote that the topic of cluster $l$ is $k$. In addition, we assume a topic $z$ is represented by a unigram language model $\theta$ that specifies the generation of a document. The vocabulary size in the document stream is set to $V$, and we leave it as our future work to model dynamic vocabulary over time. The time span of the document stream is assumed to be $[0, T]$, where $T$ could be infinite.

## 3.1 Dependence among Documents

To realize that documents temporally nearby tend to be on the same topic, CoTT encourages documents that appear temporally close to share the same cluster. CoTT achieves this via a uni-dimension Hawkes process for modeling the generation of documents within a given document cluster. As depicted in Eq. (1), a new document is more likely to be generated if there is a nearby document from the same cluster. However, as the cluster assignments on documents are not observable, CoTT imposes a DP prior over the uni-dimension Hawkes process to generate the cluster assignments. Intuitively, an existing cluster with a higher intensity at the current time point is more likely to generate the corresponding document. And this prior distribution also introduces the flexibility of creating new clusters (i.e., a new uni-dimension Hawkes process), when necessary. Hence, CoTT defines the probability of drawing cluster $c_i$ for the $i$th document $d_i$ at time $t_i$ given historical documents $\{d_1, ..., d_{i-1}\}$ and their corresponding cluster assignments $\{c_1, ..., c_{i-1}\}$ as,

$$\begin{cases} P(c_i = l) & = \frac{\lambda_l(t_i)}{\lambda_0 + \sum_{l'=1}^{L} \lambda_{l'}(t_i)}, \\ P(c_i = L+1) & = \frac{\lambda_0}{\lambda_0 + \sum_{l'=1}^{L} \lambda_{l'}(t_i)}. \end{cases} \tag{3}$$

In Eq. (3), we denote the number of unique clusters among $\{c_1, ..., c_{i-1}\}$ as $L$. On the one hand, for an existing cluster $l \in [1, L]$, its calculated intensity $\lambda_l(t_i) = \sum_{j=1}^{i-1} \alpha_l \kappa(t_i, t_j) \mathbb{I}(c_j = l)$ becomes larger if more of its documents are temporally close to the current document; and thus this new document is more likely to belong to it. We employ an exponential kernel function $\kappa(t, t_i) = \exp(-\beta_l(t - t_i))$ in CoTT to reflect the decay of temporal influence, which is controlled by the decay parameter $\beta_l$. We also assign different strength parameter $\alpha_l$ to each cluster to reflect their possibly different temporal influence on subsequent documents. In particular, we assume $\alpha_l$ follows a gamma distribution $Gamma(\tau_1, \tau_2)$. As each cluster is associated with a topic, once a cluster is assigned to document $d_i$, its correspondingly attached topic $z_l^k$ is used to generate the text content of $d_i$, i.e., $W_i \sim Multi(\theta_{z_l^k})$. This process is illustrated in the bottom layer of Figure 2.

On the other hand, the default intensity $\lambda_0$ allows the document to draw a new cluster $L + 1$, i.e., the nature of Dirichlet process. Once a new cluster is created, CoTT needs to assign a topic to it. This calls for the modeling of temporal dependence among topics, and we describe this process in the next section.

## 3.2 Dependence among Topics

The dependence among topics includes their mutual influence (e.g., topic $a$ triggers topic $b$) and temporal dynamics (variance of topic popularity over time). To capture mutual influence among topics, which is suggested by the observation that documents on correlated topics tend to appear in a close temporal proximity, we employ a multi-dimension Hawkes process to model the generation of topic assignments on clusters. As shown in Eq. (2), the mutual influence is modeled via the pairwise influence matrix $A$ between topics. Similar to the design of cluster assignment generation, we impose another DP prior over the multi-dimension Hawkes process to encourage the model to select topics that appear closer to the current timestamp. But the key difference is that although one topic might be temporally far away from the current event, it could still have a high probability to be chosen, if many of its correlated topics appear nearby, i.e., the mutual excitement. As the mutual influence differs among topics, so does the realized temporal dynamics across them. To fulfill the need for possibly different time decay at the cluster level and topic level, we choose the exponential kernel function but with different kernel parameters for this multi-dimension Hawkes process.

Moreover, to balance the tension between the decay effect of topic popularity over time imposed by the Hawkes process and the need of aligning faraway documents of the same topic, we introduce base intensity to the multi-dimension Hawkes process. It is expected capture to the popularity of a topic, which encourages a remote topic to regain attention. And the trade-off is achieved by learning the kernel parameters. This process is illustrated in the top layer of Figure 2. We should note the intensity for a document cluster (defined in Section 3.1) does not contain a base intensity term. The reason for this difference is that the document cluster is designed to capture the burstiness of documents of a certain topic ephemerally and therefore its influence on future documents is measured by temporal proximity; but a topic is supposed to be long-lasting, and thus its base intensity encourages its reappearance even though it fades away for a while.

As a result, denote the unique number of topics among $\{z_1, ..., z_{i-1}\}$ as $K$, the probability of drawing a topic for a new cluster $L + 1$ in CoTT is,

$$\begin{cases} P(z_{L+1} = k) & = \frac{\gamma_k(t_i)}{\gamma_0 + \sum_{k'=1}^{K} \gamma_{k'}(t_i)}, \\ P(z_{L+1} = K + 1) & = \frac{\gamma_0}{\gamma_0 + \sum_{k'=1}^{K} \gamma_{k'}(t_i)}. \end{cases} \tag{4}$$

For an existing topic $k \in [1, K]$, its intensity $\gamma_k(t_i) = \mu_k + \sum_{l=1}^{L} A_{z_l k} \kappa(t_l, t_{L+1})$ considers the mutual influence of previous clusters' topics on topic $k$ through $A_{z_l k}$ and the base intensity of topic $k$ (i.e., $\mu_k$). To calculate time decay, we consider the timestamp of the last document assigned to cluster $l$ as the cluster's timestamp, i.e., $t_l = \max\{t_j | c_j = l\}_{j=1}^{i-1}$. We assume the time decay in this layer is different from the uni-dimension Hawkes process used above, so that we set another decay parameter $\beta_k$ in the kernel function $\kappa(t_l, t_{L+1}) = \exp(-\beta_k(t_l - t_{L+1}))$. We assume the existing topics' mutual influence on a new topic is zero. When a new topic is created, we will sample a new language model $\theta_{K+1}$ from a Dirichlet distribution over the fixed vocabulary.

## 3.3 Modeling Document Streams with CoTT

Putting the aforementioned components together, we obtain a complete generative model of document streams. When generating document $d_i$ in a document stream $D = \{d_i\}_{i=1}^{N}$, CoTT first draws its timestamp $t_i$ from the uni-dimension Hawkes process, and then samples its cluster assignment $c_i$ from the associated Dirichlet process. If a new cluster is generated, CoTT samples its topic assignment $z_{c_i}$ from another layer of Dirichlet process based on a multi-dimension Hawkes process, which factors in the mutual influence and temporal dynamics among topics. The document content $W_i$ is subsequently sampled from the topic-specific word distribution $\theta_{z_{c_i}}$.

To complete our description of CoTT, we illustrate the generation of document stream $D$ as follows:

1. Initialize the *number of clusters* $L = 0$ and the *number of topics* $K = 0$.
2. For document $d_1$, draw $t_1 \sim Hawkes(\lambda_0)$, $\theta_1 \sim H(\theta_0)$, and $\alpha_1 \sim Gamma(\tau_1, \tau_2)$. For the word content $W_1$ of document $d_1$: $W_1^v \sim$ Multi($\theta_1$). Then set $L = L + 1$ and $K = K + 1$.
3. For document $d_i$, $i > 1$:
   (a) Draw $t_i$ from Hawkes($\sum_l \lambda_l(t_{i-1}) + \sum_k \gamma_k(t_{i-1})$).
   (b) Draw cluster $c_i$ for document $d_i$ by Eq. (3). If a new cluster is chosen, draw $\alpha_{L+1} \sim Gamma(\tau_1, \tau_2)$, and increase the number of clusters $L = L + 1$.
   (c) If a new cluster is created, sample topic by Eq. (4). If a new topic is created, draw $\theta_{K+1} \sim G_0(\theta)$, and increase the number of topics $K = K + 1$.
   (d) Draw words in document from $W_i^v \sim$ Multi($\theta_{z_{c_i}}$)

# 4 INFERENCE & PARAMETER ESTIMATION

To apply CoTT for modeling a document stream, we need to infer the latent cluster assignments on documents (i.e., $\{c_i\}_{i=1}^{N}$) and topic assignments on clusters (i.e., $\{z_l\}_{l=1}^{L}$). To perform this posterior inference, we need to first estimate the model parameters of CoTT.

The scaling parameters $\lambda_0$ and $\gamma_0$ in these two layers of DPs are set as hyper-parameters. In this section, we present an efficient algorithm to infer latent variables and estimate parameters of CoTT on the fly.

## 4.1 Inference of Cluster and Topic Assignments

We develop a particle sampling algorithm based on Sequential Monte Carlo (SMC) [9] to infer the posterior of cluster assignment $\{c_i\}_{i=1}^N$ for each document and topic assignment $\{z_l\}_{l=1}^L$ for each cluster on the fly. For simplicity, we define $\psi_n = (c_n, z_n)$ for each document $d_n$. In SMC, we keep a set of particles $\mathcal{F}$; for each particle $f \in \mathcal{F}$, we keep an approximation of the posterior $p(\psi_{1:n-1}|d_{1:n-1}, t_{1:n-1})$, i.e., the latent variable assignments before a new document $d_n$ arrives. At time $t_n$, we update the posterior to $p(\psi_{1:n}|d_{1:n}, t_{1:n})$ by sampling $\psi_n$ in each particle. Each particle is associated with a weight indicating how well the sampled assignments fit the data. If a particle's weight is lower than a threshold (usually set to $\frac{1}{|\mathcal{F}|}$), it will be replaced with a duplicate of a remaining particle, sampled with respect to their weights. This step is referred to as *resampling* in SMC.

Given a true posterior and a proposal distribution, the particle weight is defined as $w_n^f = \frac{p(\psi_{1:n}|t_{1:n}, d_{1:n})}{q(\psi_{1:n}|t_{1:n}, d_{1:n})}$. In SMC, we use $p(\psi_n|t_{1:n}, d_{1:n}, \psi_{1:n-1})$ as the proposal distribution to sample $\psi_n$ sequentially, based on the previously sampled results of $\psi_{1:n-1}$. This can also minimize the variance of $w_n^f$ [2]. At each step of SMC for CoTT, the proposal posterior is computed by $p(\psi_n|t_{1:n}, d_{1:n}, \psi_{1:n-1}) \propto p(\psi_n, d_n|t_n, d_{1:n-1}, t_{1:n-1}, \psi_{1:n-1}) = p(d_n|\psi_n, rest) \times p(\psi_n|t_n, rest)$, based on the conditional independence assumption in CoTT. To simplify the notations, we just use *rest* to denote the other random variables governing the generation of corresponding latent variables. We can further decompose the above probabilities into $p(d_n|z_n, rest) \times p(c_n|t_n, z_n, rest) \times p(z_n|t_n, rest)$.

Now we describe how to compute each component in this posterior probability. First, $p(d_n|z_n, rest)$ can be directly computed by the Dirichlet-Multinomial distribution [12]:

$$p(d_n|z_n, rest) = \frac{\Gamma(C_k^{n-1} + |V|\theta_0) \prod_v^V \Gamma(C_{v,k}^n + \theta_0)}{\Gamma(C_k^n + |V|\theta_0) \prod_v^V \Gamma(C_{v,k}^{n-1} + \theta_0)} \tag{5}$$

where $C_k^{n-1}$ and $C_k^n$ are the total number of words assigned to topic $k$ in the document set $d_{1:n-1}$ and $d_{1:n}$ respectively, $C_{v,k}^{n-1}$ and $C_{v,k}^n$ are the count of word $v$ appearing in document set $d_{1:n-1}$ and $d_{1:n}$, and $V$ is the vocabulary size.

Second, $p(c_n|t_n, z_n, rest)$ is directly defined by CoTT's uni-dimension Hawkes process over document clusters under topic $z_n$,

$$p(c_n|z_n, t_n, rest) = \begin{cases} \frac{\lambda_0}{\lambda_0 + \sum_{l'=1}^L \lambda_{l'}(t_i)\mathbb{I}(z_{l'}=z_n)} \\ \frac{\lambda_l(t_n)}{\lambda_0 + \sum_{l'=1}^L \lambda_{l'}(t_i)\mathbb{I}(z_{l'}=z_n)} \end{cases} \tag{6}$$

Finally, $p(z_n|t_n, rest)$ is defined in CoTT's multi-dimension Hawkes process as in Eq. (4).

After we sample the cluster and topic assignments for document $d_n$ in every particle, the particle weight $w_n^f$ can be updated by,

$$\begin{aligned} w_n^f = w_{n-1}^f &\times p(d_n|z_n, rest) \times p(z_n|t_n, rest) \\ &\times p(c_n|t_n, z_n, rest) \times p(t_n|t_{1:n-1}, rest) \end{aligned} \tag{7}$$

The calculation of $p(t_n|t_{1:n-1}, rest)$ is directly related to our parameter estimation procedure for CoTT; and therefore we will leave the discussion of it to the next section.

## 4.2 Online Parameter Estimation

In CoTT, the timestamps of documents associated with a particular topic $k$ can be considered as being sampled from a stochastic process consisting of a uni- and a multi-dimension Hawkes process described in Section 3, whose overall intensity function can be written as,

$$\lambda_k^*(t_n) = \mu_k + \sum_i^{n-1} \alpha_{c_i} \kappa_{\beta_l}(t_n, t)\mathbb{I}(z_{c_i} = k) + \sum_l^L A_{kz_l} \kappa_{\beta_k}(t_n, t_l) \tag{8}$$

There are three components in the above intensity function. First, the base intensity vector $\mu \in \mathbb{R}_{>0}^K$, which captures the instantaneous generation of different topics. Second, the cluster strength $\alpha \in \mathbb{R}_{>0}^L$ controls the degree of dependence among documents within a close temporal proximity. Third, the mutual influence matrix $A \in \mathbb{R}_{\geq 0}^{K \times K}$ captures the correlation among topics. These are the model parameters in CoTT. As the parameters are organized in a two layer structure, i.e., document clusters and topics, we develop a Hierarchical Alternating Direction Method of Multipliers [1, 6, 7] (H-ADMM) to estimate them accordingly.

Given a sequence of documents $\{d_1, ..., d_N\}$, assuming the time horizon of this sequence is $T$. The log-likelihood on the timestamps can be computed as,

$$\mathcal{L}(\mu, \alpha, A) = \sum_{i=1}^N \log \sum_{k=1}^K p(t_i|z_i = k, t_{1:i-1}, rest) \tag{9}$$

Since $p(t_n|t_{1:n-1}, rest) = \sum_{k=1}^K p(t_n|z_n = k, t_{1:n-1}, rest)$, Eq. (9) can be directly used in Eq. (7) to compute particle weight. To ensure meaningful correlation among topics can be identified, we impose $L1$ regularization on the learnt mutual influence matrix $A$ [21]. Consequently the optimization objective function becomes $-\mathcal{L}(\mu, \alpha, A) + \eta_A||A||_1$, where $\eta_A$ is a trade-off coefficient.

To optimize this non-differentiable objective function, we rewrite it into the following, by introducing auxiliary variable $Z$ and dual variable $U$, where $\rho > 0$ is a hyper-parameter controlling the regularization,

$$F(\mu, \alpha, A) = \min_{\mu \geq 0, \alpha \geq 0, A \geq 0} -\mathcal{L}(\mu, \alpha, A) + \eta_A||Z||_1 \tag{10}$$

$$+ \rho Tr(U^T(A - Z)) + \frac{\rho}{2}||A - Z||^2 \tag{11}$$

Then we take the following steps to iteratively optimize the model parameters.

**Step 1: Update $\mu$, $\alpha$ and A.** To solve the optimization problem defined in Eq. (10), we resort to the majorization-minimization algorithm, which optimizes the upper bound of $F(\mu, \alpha, A)$ by introducing a set of branching parameters $p_{ii}$, $p_{ji}$ and $p_{li}$.

The branching parameter $p_{ii} = \frac{\mu_{z_i}}{\lambda_{z_i}^*(t_i)}$ can be seen as the probability that the $i$-th document is generated from the base intensity.

$p_{ji} = \frac{\alpha_{c_j}\kappa_{\beta_l}(t_i,t_j)}{\lambda^*_{z_i}(t_i)}$ represents the probability that the $j$-th document leads to the generation of $i$-th documents. And $p_{li} = \frac{A_{z_l z_i}\kappa_{\beta_k}(t_i,t_l)}{\lambda^*_{z_i}(t_i)}$ indicates the probability that the $l$-th cluster of topic $z_l$ leads to the selection of topic $z_i$.

Setting the gradients of these parameters to zero, we obtain the updating rule for $\mu, \alpha, A$ as follows:

$$\mu_k = \frac{\sum_i^N p_{ii}\mathbb{I}(z_i = k)}{T}$$

$$\alpha_l = \frac{\sum_i^N \sum_{t_j<t_i} p_{ji}\mathbb{I}(c_i = l)}{\sum_{t_i}\mathbb{I}(c_i = l)\int_{t_i}^T \kappa_{\beta_l}(t,t_i)dt}$$

$$A_{kk'} = \frac{1}{4\rho}(-X + \sqrt{X^2 - 8\rho Y})$$

$$X = \rho(U_{kk'} - Z_{kk'}) + \sum_l^L \sum_i^{N_l-1} \int_{t_i}^{t_{i+1}} \kappa_{\beta_k}(t,t_i)dt$$

$$Y = -\sum_i^N \sum_{t_l<t_i} p_{li}\mathbb{I}(z_l = k', z_i = k)$$

**Step 2: Update** $Z$**.** The updating rule of auxiliary variable $Z$ is,

$$Z_{kk'} = \begin{cases} (A_{kk'} + U_{kk'}) - \frac{\eta_A}{\rho}, & A_{kk'} + (U_1)_{kk'} \geq \frac{\eta_A}{\rho} \\ (A_{kk'} + U_{kk'}) + \frac{\eta_A}{\rho}, & A_{kk'} + (U_1)_{kk'} \leq \frac{-\eta_A}{\rho} \\ 0, & |A_{kk'} + (U_1)_{kk'}| \leq \frac{\eta_A}{\rho} \end{cases}$$

**Step 3: Update** $U$**.** Given the updated $A$ and $Z$, we update the dual variable $U$ by $U_{new} = U_{old} + (A_{new} - Z_{new})$.

We adopt a batch update strategy to update the parameters of CoTT for the trade-off between accuracy and efficiency. In every batch of documents, we first sample the cluster and topic assignment for each document in all particles, and then update the particle weight accordingly. If a particle's weight is below the threshold, we will replace it by resampling. We initialize $A$ as a diagonal matrix. Once we get all documents' cluster and topic assignments in a batch, we run H-ADMM until convergence in each particle. The updated model will be applied to documents in the next batch; and this procedure is repeated through the entire document stream.

# 5 EXPERIMENTS

In this section, we verify the effectiveness of CoTT on modeling temporal dynamics in document streams through experiments on both synthetic and real-world datasets. First, on synthetic dataset, we compare CoTT with baselines on recovering the model parameters that govern the generation of data and underlying clustering structure. Second, on real-world dataset, we compare CoTT with baselines on both content and time predictions. We also visualize the temporal dynamics and mutual influence among topics learned by CoTT to illustrate its identified dependency structure.

In our evaluation, we include two state-of-the-art solutions for modeling temporal dynamics of document streams as our baselines.

• **Dirichlet Hawkes Process (DHP)** [10]: It encourages documents in a close time proximity to share the same cluster by integrating Hawkes Process with Dirichlet Process. Although this method considers temporal information to cluster documents, it does not account for the temporal dynamics or the mutual influence among topics. For documents with even identical word distribution, if the time span between them is large, they will be assigned to different topics, i.e., no reusing of historical topics.

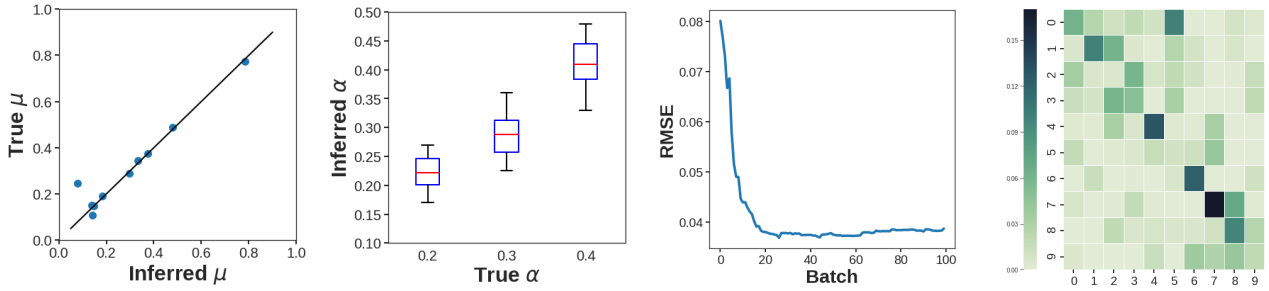
• **Hierarchical Dirichlet Hawkes Process (HDHP)** [16]: It introduces a global topic layer to DHP in order to model the temporal dynamics of topics, which allows a certain topic to be re-used over time based on its global popularity. However, the mutual influence among topics is still missing; and therefore, topics are modeled independently and the observations of one topic do not affect the inference of others.

## 5.1 Experiments on Synthetic Data

To evaluate CoTT on modeling the generation of a document stream, we first compare CoTT with the baselines on synthetic data. First, we generate synthetic data with fixed parameters based on the generative process defined in CoTT. Then, we estimate such parameters via different models on the generated data. We evaluate these models on the quality of their estimated parameters and the learnt clusters.

• **Experimental setup.** We generate document streams according to the generative process described in Section 3.3. In particular, we set the number of topics $K = 10$ and the vocabulary size $|V| = 200$. The hyperparameter of the Dirichlet distribution for generating the word distribution under a topic is set to 0.1. For every topic, we randomly select 150 unique words and set other words' generation probability to zero to control the content overlap across topics (as in total we only have 200 unique words). The length of the documents is sampled from $U(100, 200)$, and accordingly the content is sampled from the word distribution under the chosen topic. The timestamps of documents are generated by the hierarchical Hawkes process, whose parameters are set as follows: each topic's base intensity $\boldsymbol{\mu}$ is uniformly sampled from $U(.01, 1)$. In our evaluation, we vary the cluster strength parameter $\alpha$ to generate different document streams. As for mutual influence matrix $A$, we uniformly sample from $U(0, 1)$ on the $4 \times 4$ blocks on the diagonal of $A$, while sampling other elements in matrix $A$ from $U(0, .3)$. This gives us a more structured dependency relation among the topics. The decay parameters are set to $\beta_l = 1$ and $\beta_k = 0.1$. In this way, we obtain a synthetic data set with 20,000 randomly sampled documents. In the evaluation, we maintain 8 particles in CoTT's posterior sampling procedure. Once scan through the dataset, we report the results from the particle with the largest likelihood.

• **Quality of parameter estimation** To investigate the quality of CoTT on modeling the generation of data, we compare the parameters estimated by CoTT against the ground-truth parameters. Specifically, we look into the base intensity $\boldsymbol{\mu}$, cluster strength $\boldsymbol{\alpha}$ and mutual influence matrix $A$. For an inferred topic, we match its corresponding ground-truth topic by selecting the most similar one, by Kullback-Leibler divergence over word distributions under topics. Results are reported in Figure 3. From Figure 3 (a), we can observe that the inferred $\mu$ is close to the ground-truth, which supports that CoTT can capture the popularity of topics in a document stream. To verify the effectiveness of estimating $\boldsymbol{\alpha}$, we generate three document streams with $\boldsymbol{\alpha} = \{0.2, 0.3, 0.4\}$. Results in Figure 3 (b) show that CoTT is able to learn the parameters with a reasonable variance. As Figure 3 (c) shows, the RMSE between




(a) Estimation of $\boldsymbol{\mu}$.

(b) Estimation of $\boldsymbol{\alpha}$ on three synthetic data sets with different ground-truth $\boldsymbol{\alpha}$.

(c) RMSE of mutual influence matrix over time.

(d) Absolute error between inferred A and true A.

**Figure 3: Parameter estimation performance of CoTT on synthetic data set.**

| Model | Overlap | NMI | ARI |
|---|---|---|---|
| DHP | 0.6 | 0.867 | 0.708 |
| | 0.8 | 0.776 | 0.372 |
| | 0.9 | 0.551 | 0.141 |
| HDHP | 0.6 | **0.998** | **0.997** |
| | 0.8 | 0.919 | 0.862 |
| | 0.9 | 0.652 | 0.274 |
| CoTT | 0.6 | 0.992 | 0.989 |
| | 0.8 | **0.986** | **0.979** |
| | 0.9 | **0.972** | **0.963** |

**Table 1: Clustering performance on synthetic data with different overlap ratio.**

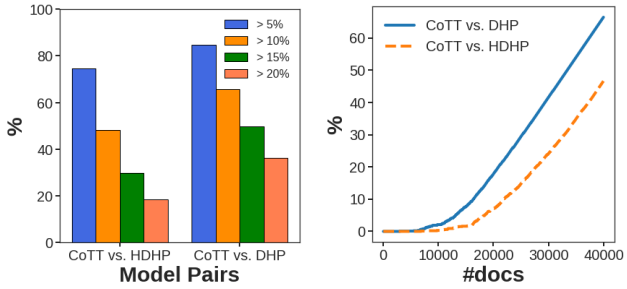
the learnt and ground-truth mutual influence matrices becomes smaller when more documents are available for model update, i.e., improved dependency structure modeling. After convergence, the absolute error in estimating the mutual influence is considerably small, as reported in Figure 3 (d). These observations support the capability of CoTT in modeling complex document streams.

• **Clustering performance.** The key outcome of modeling a document stream is to identify the underlying dependence among documents and topics, which is reflected in the inferred clustering structure of documents. In addition to comparing the models on recovering the underlying parameters for data generation, we further compare CoTT with DHP and HDHP via clustering-based metrics, NMI and ARI [19], on their identified document clusters. We evaluate CoTT and HDHP with the clustering structure at the topic level. As DHP does not have a global topic layer, we treat the clusters learnt by DHP as topics. We create several synthetic data sets with different topic overlap ratio: when the ratio is larger, it is harder to utilize textual information to learn accurate clustering structure, because the topics would look similar to each other. CoTT outperformed the two baselines even with a large topic overlap, which indicates modeling temporal dynamics help CoTT uncover the dependence among documents and topics accurately.

## 5.2 Experiments on Real-World Data

In this section, we quantitatively compare CoTT with the baselines on content and time predictions, and qualitatively analyze the temporal dynamics identified by CoTT.

*5.2.1 Quantitative Analysis.* We used 40,000 news articles from January 1, 2017 to July 15, 2017 of "All the news" data set for our evaluation. "All the news" data set is a public data set from kaggle [1] and consists of news articles from 5 major U.S. news publishers, including CNN, Breitbarg and etc. We performed stopword removal and stemming in our pre-processing. We chose 10,000 words with the highest document frequency as our vocabulary. The timestamps are scaled in hours. In CoTT, we set its scaling parameter $\lambda_0 = 0.05$ and $\gamma_0 = 0.1$; the hyper-parameter of the base distribution used to sample topics was set to 0.1; the decaying parameter $\beta_l$ and $\beta_k$ were set to 3 and 1. We used 8 particles to posterior inference and model estimation. In H-ADMM, the hyper-parameters $\rho$ was set to 1 and $\eta_A$ was set to 0.1. The batch size of training was fixed to 100.




(a) Percentage of documents better than baselines.

(b) Cumulative percentage of documents better than baselines.

**Figure 4: Performance of content prediction on real-world data set.**

• **Content Prediction.** The task of content prediction is that given documents $d_{1:n-1}$ and their corresponding timestamps $t_{1:n-1}$, one needs to predict how likely a particular document $d_n$ will appear at time $t_n$. Presumably a model which can better capture the dependence among documents will better predict the content of a future document. In particular, this likelihood is calculated as $p(d_n|t_n, d_{1:n-1}) = \sum_{k=1}^{K} p(d_n|z_k, d_{1:n-1})p(z_k|t_n)$, where $p(d_n|z_k, d_{1:n-1})$ is the likelihood of document content $d_n$ given historical documents and a specific topic $z_k$. As CoTT, DHP and HDHP make the same assumption of document generation given topic

[1] https://www.kaggle.com/snapcrack/all-the-news/home

assingments, we can compute $p(d_n|z_k, d_{1:n-1})$ via Eq. (5) for CoTT, DHP and HDHP. $p(z_k|t_n)$ corresponds to the intensity functions of topic $z_k$ defined in each model. For example, for CoTT we use Eq. (8) to compute it. After predicting the content of document $d_n$, we treat $d_n$ as observed and move onto the prediction of next document $d_{n+1}$, until the end of this document stream.

Due to various length of documents, the evaluated likelihood varies in a set of documents, which makes the comparison across models difficult. To accurately investigate the performance of different models in predicting document content, we compare models' performance with respect to their predictions on each document one by one. We use the percentage of documents where CoTT produces a higher likelihood than baselines as our evaluation metric (as $d_n$ is what we actually observed at time $t_n$). As Figure 4 (a) shows, in 74.7% documents, CoTT's predicted likelihood are 5% larger than that predicted by HDHP, and in 84.6% documents against DHP. The cumulative percentage of documents where CoTT has a higher likelihood than baselines is reported in Figure 4 (b). The results are expected. Without a global topic layer, DHP cannot refine the old clusters when they reappear, leading to worse content modeling quality. HDHP's modeling of temporal dynamics is limited by not considering the mutual influence among topics, which cannot fully exploit the relatedness among different topics.

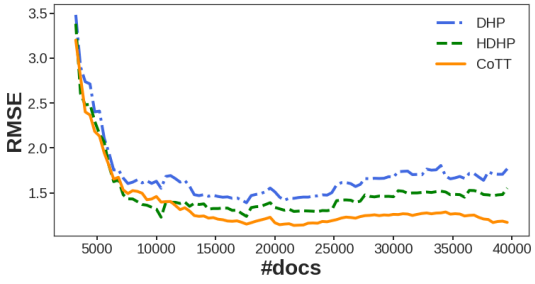
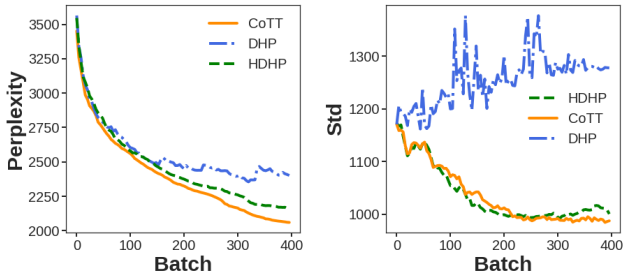



**Figure 5: RMSE of time prediction on real-world data.**

• **Time Prediction.** As a Hawkes process based model, CoTT is able to predict the timestamps of documents. The task of time prediction is given documents $d_{1:n-1}$ and their timestamps $t_{1:n-1}$, predict the timestamp $t_n$ that the given document $d_n$ appears. We believe a model that better captures the temporal dynamics among existing documents can predict the future arrival time of documents more accurately. We calculate the likelihood of different future timestamps in a fix-size predictive time window, and select the one with the highest likelihood as the predicted timestamp for the given document. The likelihood of timestamp $t_n$ is given by $p(t_n|d_n, d_{1:n-1}, t_{1:n-1}) = p(t_n|t_{1:n-1}) \sum_{k=1}^{K} p(d_n|z_k, d_{1:n-1}) p(z_k|t)$, where $p(t_n|t_{1:n-1})$ is the probability of timestamp $t_n$ given historical timestamps. We obtain $p(d_n|z_k, d_{1:n-1}), p(z_k|t)$ in the same way as those for content prediction mentioned above. Again, after predicting $t_n$, we include document $d_n$ into observed historical events and move onto the prediction of $t_{n+1}$ of next document, until the end of the document stream.

We report the root mean squared error between the predicted and ground-truth timestamps of the next document. Results in Figure 5 show that all three models could predict the time of next documents better when more training documents are available, but CoTT achieves a more competitive RMSE than DHP and HDHP. DHP only models the dependence among documents in a close time proximity, so that it cannot predict when topics reoccur, leading to a higher time prediction error, even when more documents are available for model estimation. HDHP only models the popularity of topics, which encourages the recurrence of popular topics. This limits its flexibility of temporal dynamics modeling. Mutual influence among topics enables CoTT to better realize the temporal dynamics of topics, which helps it make accurate time prediction in a longer time horizon.




(a) Mean (b) Standard deviation

**Figure 6: Changes in the mean and standard deviation of perplexity when training on real-world data set.**

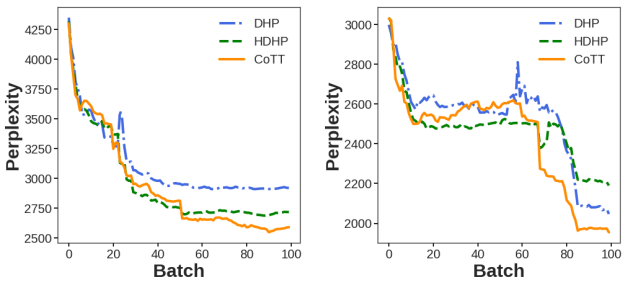
• **Perplexity.** In addition to the evaluation of content prediction above, where we always use the latest model to predict the next document on the fly, we also perform evaluation in a more classical setting, where we preserve documents only for the testing purpose. We evaluate a model's perplexity on those reserved testing documents. A model that extracts the underlying structure of document stream more accurately in the training set can better fit the content of documents in test set and thus achieve a smaller perplexity. We divide the dataset into batches of 100 documents, and randomly select 5 documents in every batch as test set. Because the time span of every batch is small, and selected test documents are evenly distributed in training set. In each batch's test set, we assume there is no new topic created (as the time intervals are all generally very small). The perplexity in a test set is calculated as,

$$perplexity = \exp\left\{ - \frac{\sum_{i:e_i \in \mathcal{D}_{test}} \log p(d_i|\mathcal{D}_{train})}{\sum_{i:e_i \in \mathcal{D}_{test}} |d_i|} \right\}$$

From Figure 6 (a), we could see that all models' perplexity decreases with more batches of documents used for model update. However, since DHP cannot re-use the cluster from temporally far-away documents, it has to create new clusters constantly to fit new data, which is proved by the fluctuation of the standard deviation of perplexity (reported in Figure 6 (b)). Thus DHP eventually had the worst perplexity in this evaluation. Although HDHP allows a certain topic to be re-used, it encourages globally popular topic to re-appear, which ignores the local context of most recently selected topics. Consequently, HDHP captures the underlying structure of document streams less accurately than CoTT does, which is suggested by HDHP's worse perplexity performance.

• **Temporal Refinement of Topics.** As a temporal topic model, with more observed training documents, the topics inferred by

CoTT are expected to be refined on the fly. To verify this, we conduct an in-depth experiment of perplexity using the same setup above. Every time when a training batch is processed, we calculate the perplexity of all the testing batches and report them respectively. We keep track of perplexity change after each training batch.




(a) Change of perplexity in the 50th testing batch.

(b) Change of perplexity in the 80th testing batch.

**Figure 7: Change of perplexity on particular testing batches during online model update.**

In Figure 7, we report the change of perplexity in the 50th and 80th testing batches over all 100 training batches. The perplexity decreases while topics are refined over time. In Figure 7 (a), there is a sharp decrease at the 50th batch in CoTT, because CoTT chose to create new topics which fit those new documents better. Although HDHP allows topics to be re-used, it fail to detect new ones without the help of temporal dynamics of topics. Additionally, CoTT begins to decrease sharply before DHP and HDHP in Figure 7 (b). With the help of mutual influence among topics, CoTT creates topics suitable for future data, which is supported by the early decrease in its perplexity.

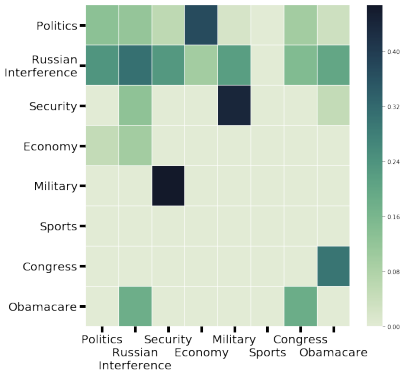



**Figure 8: Mutual influence among topics.**

*5.2.2 Qualitative Analysis.* The learnt topics in CoTT summarize a document stream. In this section, we visualize both the topics and the mutual influence matrix among topics learnt by CoTT.

• **Temporal Dynamics of Topics.** In Figure 9 (a) to (h), we select 8 topics for illustration with word cloud which scales words concerning their probabilities under the topic. We can observe that words which have large probabilities are generally in close meaning and coherent, which indicate suggest meaningful real-world news events in this period of time. In addition, we select two pairs of topics to visualize their intensities over time, which are *Security* and *Military*, *Congress* and *Obamacare*. To fit a pair of topics in a figure, we scale the intensity of each topic. We could see that topic *Security* and topic *Military* appear almost always simultaneously, and topic *Congress* and *Obamacare* follow one another over time. This observation suggests that topics appearing together frequently may be closely related semantically. In addition, we can also find that the same topic may peak and fade over a long period of time, which can still be captured by CoTT.

• **Temporal Dependence among topics.** To have a better understanding of mutual influence among topics, we investigate the mutual influence among 8 representative topics without loss of generality. In Figure 8, we show the mutual influence matrix among these topics, along with their word distributions in Figure 9. We have the following observations: (1) The dependence among topics captured by CoTT is asymmetric. For example, *Economy* has a strong influence on *Politics*, but the reverse does not hold. (2) Mutual influence is sparse. We can observe a lot of zeros in the matrix. This is because some topics do not have transition to other topics in close time proximity. The dependence among these topics is very weak. For example, *Sports* does not have mutual influence on other seven topics.

## 6 CONCLUSION & FUTURE WORKS

In this paper, we proposed Correlated Temporal Topic Model (CoTT), which integrates the Hierarchical Dirichlet process and the Hawkes process, to model the temporal dependency among documents and topics in a document stream. It learns time-sensitive topic distribution and captures mutual influence among topics. Experiments on both synthetic and a large real-world news dataset confirm its effectiveness in discovering meaningful temporal information and topical dependency in a document stream.

In our current solution, we use a single topic to fit a document, which might limit the model's capability in document modeling. As shown in existing works [5], modeling documents as a mixture over a set of topics can better capture the embedded semantics. But this also introduces another dimension of complexity in modeling document streams, as we have to face a dynamic mixture of topics in each document. In addition, our current correlation modeling among topics does not consider the word distribution of specific topics. For example, topics of similar word distribution might have a stronger mutual influence on each other. We plan to model topical correlation as a function of topic content and directly optimize such function for better correlation estimation.

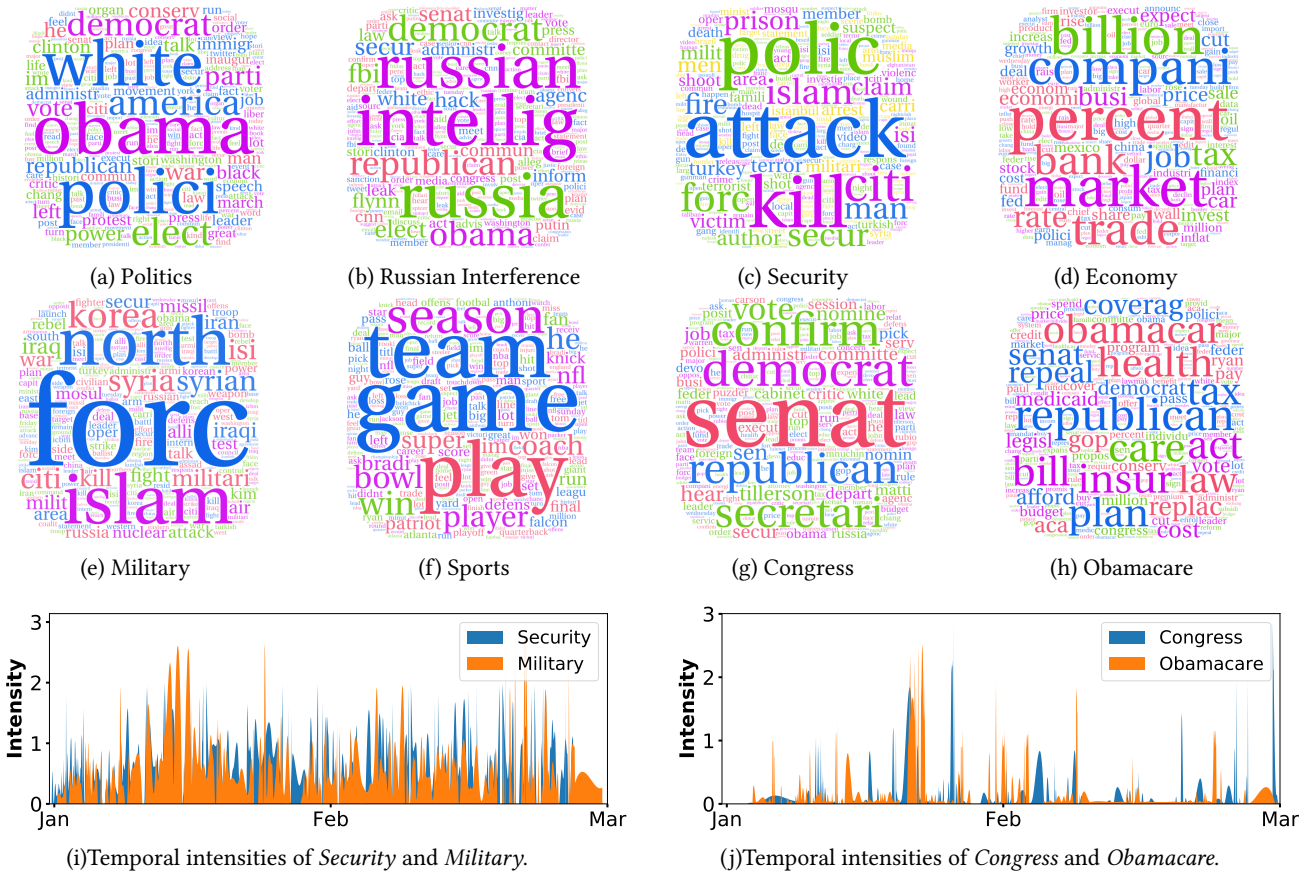
## 7 ACKNOWLEDGEMENT

We thank the anonymous reviewers for their insightful comments. This research was supported by the National Science Foundation under grant IIS-1718216 and IIS-1553568.

## REFERENCES

[1] Odd Aalen, Ornulf Borgan, and Hakon Gjessing. 2008. *Survival and event history analysis: a process point of view.* Springer Science & Business Media.

[2] Amr Ahmed, Qirong Ho, Jacob Eisenstein, Eric Xing, Alexander J Smola, and Choon Hui Teo. 2011. Unified analysis of streaming news. In *Proceedings of the 20th international conference on World wide web.* ACM, 267–276.




(a) Politics (b) Russian Interference (c) Security (d) Economy

(e) Military (f) Sports (g) Congress (h) Obamacare

(i)Temporal intensities of *Security* and *Military*. (j)Temporal intensities of *Congress* and *Obamacare*.

**Figure 9: Visualization of learnt topics in CoTT. The first two rows are word clouds of 8 selected topics. To save space, we present the temporal intensity of two pairs of correlated topics, *Security* and *Military*, *Congress* and *Obamacare*, to illustrate their underlying temporal mutual influence. Gaps on the $x-$axis represent different clusters under a topic. Topics' names are set based on their top ranked words.**


[3] Amr Ahmed, Qirong Ho, Choon Hui Teo, Jacob Eisenstein, Alex Smola, and Eric Xing. 2011. Online inference for the infinite topic-cluster model: Storylines from streaming text. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 101–109.

[4] David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*. ACM, 113–120.

[5] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[6] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning* 3, 1 (2011), 1–122.

[7] Renqin Cai, Xueying Bai, Zhenrui Wang, Yuling Shi, Parikshit Sondhi, and Hongning Wang. 2018. Modeling Sequential Online Interactive Behaviors with Temporal Point Process. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 873–882.

[8] Qiming Diao and Jing Jiang. 2014. Recurrent chinese restaurant process with a duration-based discount for event identification from twitter. In *Proceedings of the 2014 SIAM International Conference on Data Mining*. SIAM, 388–397.

[9] Arnaud Doucet, Nando De Freitas, Kevin Murphy, and Stuart Russell. 2000. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 176–183.

[10] Nan Du, Mehrdad Farajtabar, Amr Ahmed, Alexander J Smola, and Le Song. 2015. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 219–228.

[11] Nan Du, Le Song, Hyenkyun Woo, and Hongyuan Zha. 2013. Uncover topic-sensitive information diffusion networks. In *Artificial Intelligence and Statistics*. 229–237.

[12] Tom Griffiths. 2002. Gibbs sampling in the generative model of latent dirichlet allocation. (2002).

[13] Alan G Hawkes. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58, 1 (1971), 83–90.

[14] Alan G Hawkes and David Oakes. 1974. A cluster process representation of a self-exciting process. *Journal of Applied Probability* 11, 3 (1974), 493–503.

[15] Thomas Josef Liniger. 2009. *Multivariate hawkes processes*. Ph.D. Dissertation. ETH Zurich.

[16] Charalampos Mavroforakis, Isabel Valera, and Manuel Gomez Rodriguez. 2016. Modeling the dynamics of online learning activity. *arXiv preprint arXiv:1610.05775* (2016).

[17] Caroline Suen, Sandy Huang, Chantat Eksombatchai, Rok Sosic, and Jure Leskovec. 2013. Nifty: a system for large scale information flow tracking and clustering. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 1237–1248.

[18] Yee W Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2005. Sharing clusters among related groups: Hierarchical Dirichlet processes. In *Advances in neural information processing systems*. 1385–1392.

[19] Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research* 11, Oct (2010), 2837–2854.

[20] Xuerui Wang and Andrew McCallum. 2006. 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*. ACM, 424–433.

[21] Ke Zhou, Hongyuan Zha, and Le Song. 2013. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *Artificial Intelligence and Statistics*. 641–649.