4-2014

Recurrent Chinese Restaurant Process with a Duration-based Discount for Event Identification from Twitter

Qiming Diao
Singapore Management University, qiming.diao.2010@smu.edu.sg

Jing JIANG
Singapore Management University, jingjiang@smu.edu.sg

Recurrent Chinese Restaurant Process with a Duration-based Discount for Event Identification from Twitter

Qiming Diao* Jing Jiang*

Abstract

Due to the fast development of social media on the Web, Twitter has become one of the major platforms for people to express themselves. Because of the wide adoption of Twitter, events like breaking news and release of popular videos can easily catch people's attention and spread rapidly on Twitter, and the number of relevant tweets approximately reflects the impact of an event. Event identification and analysis on Twitter has thus become an important task. Recently the Recurrent Chinese Restaurant Process (RCRP) has been successfully used for event identification from news streams and news-centric social media streams. However, these models cannot be directly applied to Twitter based on our preliminary experiments mainly for two reasons: (1) Events emerge and die out fast on Twitter, while existing models ignore this burstiness property. (2) Most Twitter posts are personal interest oriented while only a small fraction is event related. Motivated by these challenges, we propose a new nonparametric model which considers burstiness. We further combine this model with traditional topic models to identify both events and topics simultaneously. Our quantitative evaluation provides sufficient evidence that our model can accurately detect meaningful events. Our qualitative evaluation also shows interesting analysis for events on Twitter.

1 Introduction

With the rapid growth of social media on the Web and the fast adoption of smart mobile devices, the way people consume information has been fundamentally changed. For the younger generation, traditional media such as newspapers, TV and radio have been replaced by new media such as Twitter, Facebook and YouTube. Moreover, social media allow users to actively participate in generating content. In particular, Twitter as a microblog site allows people to publish short, instant textual posts anywhere and anytime, making content generation ever easier. A consequence of the wide adoption of Twitter is that the popularity and importance of a news event can be approximately gauged by the volume of relevant tweets covering the event. In addition, relevant tweets also reflect the public's opinions and reactions to events such as elections and scandals. It is therefore very useful to find popular events and their relevant tweets from Twitter. In this paper, we study event identification from Twitter streams.

The problem we study is similar to but different from *event detection* on Twitter that has been a hot research topic in recent years. Existing work on event detection from Twitter usually focuses on *early*, *online* detection of major events [17, 15, 21, 6, 14]. For example, Sakaki et al. studied realtime detection of earthquake events for Japan [17]. Petrović et al. studied how to detect the first tweet covering a new event [15]. These studies stress the importance of detecting the onset of an event at the moment or shortly after the event happens, which is critical for monitoring social media for unexpected events such as natural disasters, terrorist attacks and outbreaks of contagious diseases.

In contrast, we study identification of events from a given segment of Twitter stream in a *retrospective* manner. We argue that it is also useful to study this retrospective event identification problem because of several reasons: (1) Petrović et al. recently pointed out that Twitter stream does not lead news streams for major news events, but Twitter stream covers a much wider range of events than news streams [16]. It suggests that early detection of events on Twitter may not be as desirable as we thought but retrospective event identification may help recover a wider range of events than what mainstream news media cover. (2) Retrospective event identification allows us to measure the significance of an event based on the amount of user attention reflected in the volume of tweets throughout the entire life cycle of the event. (3) It also enables us to collect different perspectives on the event by different users after it takes place, and to subsequently summarize the event comprehensively. (4) It makes it possible to study the evolution of events if we observe different phases of an event by analyzing all the relevant tweets.

Formally, we define an event to be something nontrivial that happens at a certain time. An event can be either planned or unexpected. Examples of events include plane crashes, concerts, elections, etc. Traditionally the definition of an event also requires a certain place where the event happens [23, 4]. Here we remove this constraint because we also consider events that happen in the online

*School of Information Systems, Singapore Management University. (email: qiming.diao.2010@smu.edu.sg, jingjiang@smu.edu.sg)

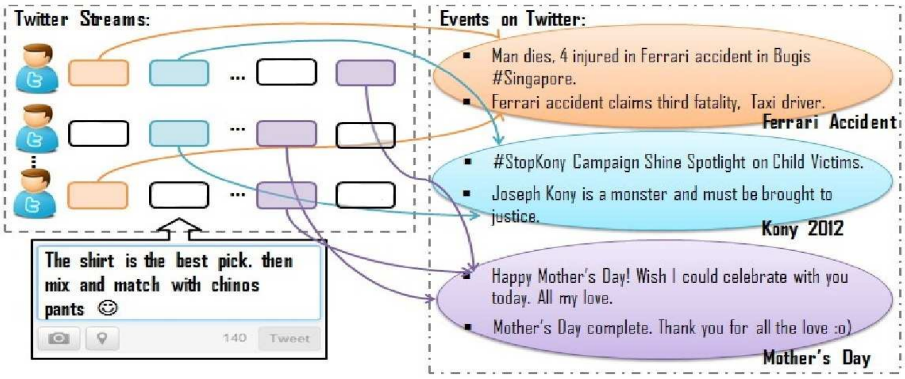


Figure 1: Example events on Twitter and some representative tweets. The Twitter stream is split into multiple ones, each per user. Note that tweets can be both event related (colored) and personal life related (in white).

space, such as the viral spread of a video online. Given this notion of event, our problem is to identify (gapped) subsequences of tweets from a segment of Twitter stream where each subsequence contains tweets discussing the same event. Figure 1 illustrates the problem definition and shows some example events with their representative tweets.

The problem can be regarded as an evolutionary clustering problem, where items are ordered as a stream and clustered depending on not only their similarity but also their closeness in time. For evolutionary clustering of streaming documents, several methods have been proposed, including some from the information retrieval community to address the event detection problem under Topic Detection and Tracking (TDT) (e.g. [23, 5, 24]) and others from the machine learning and data mining communities (e.g. [1, 3]). In particular, Ahmed and Xing proposed a dynamic non-parametric model called the Recurrent Chinese Restaurant Process (RCRP), which performs evolutionary clustering of streaming documents in a principled and elegant way [3]. Being a non-parametric model, it also allows a countably infinite number of clusters and flexibly models the life cycle of each cluster. Because of these appealing characteristics, we choose RCRP as the basis of our solution.

Although RCRP has been successfully applied to find events from news streams [2] and news-centric social media streams [18], Twitter has some major differences from news streams and therefore these existing models are not directly applicable to our problem. (1) Existing models assume that all documents are event-related and must be assigned to a cluster. On Twitter, however, many tweets are not related to any significant event. According to a Twitter study by PearAnalytics[1], only 3.6% of tweets are news-related and 8.7% have pass-along value. The majority of tweets are about people's personal interests and daily routines. We therefore separate tweets into *topic tweets* and *event tweets*, which capture user's personal life topics and major events on Twitter respectively. We identify the former using a topic model and the latter using a RCRP-based model. Although this assumption is a much simplified view of the wide range of tweets, we find it effective to detect meaningful events and topics. (2) RCRP does not model the phenomenon that events on Twitter are bursty. Because of the nature of microblogs, people usually use Twitter to spread or comment on breaking news rather than old events, which means events on Twitter tend to die out fast. However, RCRP only captures the "rich get richer" phenomenon. We therefore need to introduce some mechanism to favor bursty clusters.

In this paper, we propose a new non-parametric generative model for identifying events from Twitter. Following [2] and [18], our model distinguishes between longstanding topics and bursty events. In our model, only events are modeled by RCRP and allowed to emerge and disappear along the timeline. Different from the previous models, we separate topical tweets from event-related tweets by considering each user's longstanding topical interests. Moreover, we introduce a novel duration-based probability discount into RCRP, which penalizes longstanding events and hence models the burstiness of events on Twitter.

We evaluate our model on a real Twitter dataset that contains the posts of 500 users published during a period of three months from April to June 2012. Our experiments show that our proposed model can more accurately identify meaningful events than two baseline methods. Our model also finds more relevant tweets and generates better temporal profiles of events.

Our work has the following contributions: (1) We propose a principled unified probabilistic model for event identification on Twitter. Each event forms its own cluster inside the model and no post-processing is needed. (2) Event-related tweets can be separated from personal topical tweets automatically within our unified model. (3) We propose a

[1] http://www.pearanalytics.com/wp-content/uploads/2012/12/Twitter-Study-August-2009.pdf

novel duration-based probability discount for RCRP, which allows us to capture the burstiness of events on microblogs.

2 Related Work

The problem we study is related to several lines of work, which we discuss below.

Event Detection on Twitter: There have been quite a few studies on event detection on Twitter [17, 15, 21, 6, 14, 22]. Sakaki et al. trained a classifier to recognize tweets reporting earthquakes in Japan [17]. Weng and Lee proposed a method that first characterizes temporal patterns of individual words using wavelets and then groups them into events [21]. Petrović et al. proposed the first story detection task on Twitter [15]. Becker et al. explored supervised approaches to distinguishing between messages about real-world events and non-event messages for Twitter stream analysis [6]. Xie et al. proposed a sketch-based topic model together with a set of techniques to achieve real-time detection of bursty topics on Twitter [22]. As these studies focus on early event detection, their major concerns are storage of past posts and efficient ways of computing similarities between posts. However, recently, Petrović et al. pointed out that Twitter does not necessarily lead traditional news media on major events, which suggests that early event detection on Twitter may not be as critically important as thought to be.

Topic Detection and Tracking (TDT): TDT is a relatively old research problem in the information retrieval community. A topic is defined as "a seminal event or activity, along with all directly related events and activities [4]." Much work has been done along this direction [5, 24, 23, 12], and these studies focus mostly on news articles, which used to be the best source of information to detect and summarize events. These studies are mainly based on two approaches: *document-pivot* and *feature-pivot*. The former aims to represent documents as vectors and calculate similarities between documents, and then cluster documents into events [5, 24, 23]. The latter aims to identify the features of the hidden events from the stream first, and then detect events by clustering these features [12]. Nevertheless, identifying events on Twitter stream is more challenging and our approach is quite different, due to several reasons: (1) Only a small proportion of tweets is event-related in Twitter streams, while most news articles are event-oriented. (2) Twitter content is user generated, where each user has her specific characteristics. We use a probabilistic approach which detects events and considers users' personal interests at the same time. It is worth mentioning that "topic" means seminal events in TDT here, while "topic" represents longstanding personal interests in our task.

Temporal Topic Models and the Recurrent Chinese Restaurant Process: LDA is a widely-used method to model topics [9], but it is static. There have been many extensions to LDA to capture the temporal aspects of topics [8, 19, 11]. We only mention a few here. Blei and Lafferty considered the evolution of topics based on discretization of time [8]. Wang and McCallum model continuous time using a Beta distribution [19]. The models proposed in [20] and in [10] assume a topic distribution within each time epoch. However, these models need to pre-define the number of topics. Intuitively, the number of events should reach countable infinite over time in text streams. The Recurrent Chinese Restaurant Process overcomes this limitation by allowing topics to emerge and disappear along the timeline [3]. Ahmed et al. proposed a unified model which combines the Recurrent Chinese Restaurant Process with LDA to detect events in news stream [2]. Tang et al. further extended the model by capturing user interests in some news-centric social media streams [18]. These studies are closely related to our approach, and the details will be given in the next section.

3 Method

We first briefly review RCRP and its application for event identification. Because our preliminary experiments show that existing RCRP based models cannot be directly applied in our task, we then introduce our method, which extends RCRP.

3.1 Recurrent Chinese Restaurant Process The Recurrent Chinese Restaurant Process (RCRP) is a non-parametric model for evolutionary clustering proposed by Ahmed and Xing [3], which basically chains up the Chinese Restaurant Process (CRP) [7] based on the timeline. To model streaming data, RCRP models a restaurant with infinite number of tables and customers coming on different days. When the i-th customer on the t-th day comes in, she can choose a table that either is serving some customers on day t or served some customers on day $t-1$ (or both) with probability $\frac{n_{k,t-1}+n_{k,t}^{(i)}}{N_{t-1}+i-1+\alpha}$, where $n_{k,t-1}$ is the number of customers sitting at table k at the end of day $t-1$, $n_{k,t}^{(i)}$ is the number of customers sitting at table k on day t before customer i comes, and N_{t-1} is the total number of customers served by the restaurant on day $t-1$. This customer can also choose to sit at a new empty table that did not even serve any customer on the previous day with probability $\frac{\alpha}{N_{t-1}+i-1+\alpha}$. With the RCRP metaphor, we can cluster a sequence of items that are divided into epochs. Each resulting cluster not only contains a set of items but also has a duration with a start time and an end time. The RCRP model encourages popular clusters in epoch $t-1$ to remain alive in epoch t. Under RCRP, items from different epochs are no longer exchangeable. When RCRP is applied for document clustering, we further assume that each cluster is associated with a multinomial word distribution. We can then model the generation of documents

from a cluster where each document is a bag of words. Such a model allows us to prefer documents with similar word usage to be clustered together.

The RCRP model can be used to cluster news articles into storylines where each story is a series of news articles about the same event [2, 18]. Ahmed et al. [2] proposed a RCRP-LDA model which assumes that there is a fixed number of topics that exist at all times and an infinite number of events that emerge and disappear over time. Each document is assumed to belong to an event, but words inside a document can be either topical or event related. Tang et al. further extended the RCRP-LDA model to incorporate user interests by assuming that each event has a user distribution [18]. They applied their model to some news-centric social media streams such as Digg and online discussion forums.

3.2 Our Motivation However, as we have stated earlier, a major difference between Twitter and news article streams is that the majority of tweets is about trivial events and personal interests, while only a small fraction of them is event related. Therefore, we cannot directly apply existing RCRP based models or TDT methods, which assume each document is related to an event. To illustrate this difference, we apply two representative existing methods for event identification from news articles to a subset of our Twitter data from September to November 2011. The first one is a TDT method from [24], which aims to detect events from news streams retrospectively using hierarchical group average clustering. The second method is from [18] for identifying events from news-centric social media streams using RCRP and LDA. We show some top-ranked events identified by the two methods in terms of top keywords in Table 1. For each event, we also plot a temporal profile that shows the volume of identified relevant tweets over time. We can see that the top keywords and the temporal profiles do not clearly indicate any important event. Besides the special property that not every tweet is event-related, another characteristic of events on Twitter is that they tend to be bursty. Standard RCRP only models the "rich get richer" phenomenon, which can lead to events with long durations.

To address the two problems above, we propose a different RCRP-based generative model for identifying events from Twitter. The proposed model assumes that a tweet is either topical or event-related. It further introduces a duration-based probability discount to favor bursty events.

3.3 Preliminaries We first formally formulate our problem before we go to the detail of our method. We assume that we have a stream of tweets that are divided into T epochs. (In our experiments, we use one day as an epoch.) Let $t \in \{1, 2, \ldots, T\}$ denote the index of an epoch. Each epoch contains a sequence of tweets ordered by their exact time

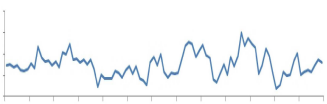
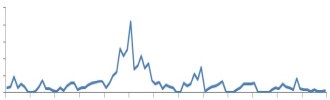
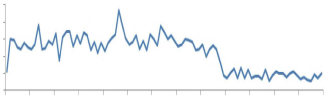
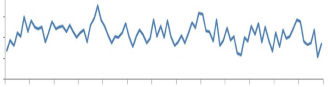
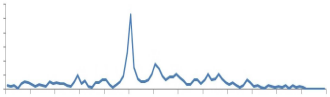
Top words	Temporal profile
time, love, good, people, la, sleep, today, im, day, work	
day, #nowplaying, gonna, video, song, there's, yeah, wait, find, love	
steve, jobs, love, time, good, feel, rip, happy, damn, miss	
people, life, make, love, moment, person, awkward, hate, smile, things	
sleep, tired, im, home, bed, early, wake, gonna, rain, feel	
iphone, steve, jobs, apple, 4s, ios, app, phone, siri, rip	

Table 1: Three of the top-ranked events identified by the models in [24] (top three) and [18] (bottom three) from our data.

stamps, and each tweet is a bag of words. Let V be the size of our vocabulary and let $w_{t,i,j} \in \{1, 2, \ldots, V\}$ denote the j-th word (represented by its index in the vocabulary) from the i-th tweet in the t-th epoch. We also take note of the authors of these tweets. Let $u_{t,i} \in \{1, 2, \ldots, U\}$ denote the user who published the i-th tweet in the t-th epoch, where U is the total number of users. Our goal is to identify a set of events from these tweets, where each event is a set of tweets. Note that not every tweet has to belong to an event.

Our general idea is to cluster these tweets such that each cluster represents an event. But since not all tweets are event-related, we assume that each tweet is either about a general longstanding topic (a *topical* tweet) or related to an event (an *event-related* tweet). Only the event-related tweets will be clustered using the Recurrent Chinese Restaurant Process. For the topical tweets, we assume that they are closely related to each user's topical interests.

3.4 A Duration-based Discount for RCRP Recall that one problem we have identified with RCRP for Twitter is that RCRP only models the "rich get richer" phenomenon. In other words, popular events tend to attract even more users to tweet about them. However, on microblogs users also tend to

follow the newest trends. Once an event becomes old, it may no longer attract much attention. In fact, [13] identified these two factors on both mainstream and social media and termed them *imitation* and *recency*. They argued that any model of the news cycle needs to incorporate some version of these two ingredients. RCRP already captures the imitation factor. What is missing is the recency factor.

We therefore propose the following change to the standard RCRP. Recall that in the RCRP metaphor, when the i-th customer of the t-th epoch comes in, the probability to join an existing table k is proportional to $(n_{k,t-1} + n_{k,t}^{(i)})$, i.e. the number of customers sitting at table k on the current and the previous days before customer i comes. Let $\bar{t}_k$ denote the index of the epoch when table k was first occupied. Based on the recency effect, the earlier a table was first occupied, the older the table is and the less likely it will be chosen. We hence want to discount the probability mass to join table k based on $(t - \bar{t}_k)$. Here we propose a discount of $(n_{k,t-1} + n_{k,t}^{(i)})(1 - e^{-\lambda(t-\bar{t}_k)})$, that is, after the discount, the remaining probability mass is $(n_{k,t-1} + n_{k,t}^{(i)})e^{-\lambda(t-\bar{t}_k)}$. Here $\lambda > 0$ is a parameter we can tune. It is obvious that the older table k is, the smaller $\bar{t}_k$ is and the smaller the probability mass for table k is after the discount. On the other hand, the deducted probability mass will be used for starting a new table.

Formally, define $\Delta_{k,t}^{(i)}$ as $(n_{k,t-1}+n_{k,t}^{(i)})(1-e^{-\lambda(t-\bar{t}_k)})$, the duration-based probability discount. Then for the i-th customer of the t-th epoch, she can choose to join an existing table with probability $\frac{n_{k,t-1}+n_{k,t}^{(i)}-\Delta_{k,t}^{(i)}}{N_{t-1}+i-1+\alpha}$ or start a new table with probability $\frac{\alpha+\sum_{k'}\Delta_{k',t}^{(i)}}{N_{t-1}+i-1+\alpha}$.

With the discounted RCRP model, customers prefer not only popular tables but also "fresh" tables. This is the major distinction of our proposed model from the standard RCRP. The discount model also maintains the total probability mass as $(N_{t-1} + i - 1 + \alpha)$, which simplifies the model inference later.

3.5 The Complete Model We are now ready to formally present our complete model for event identification from Twitter. We assume that there are A longstanding topics, each associated with a multinomial word distribution ϕ_a. Each user u has a topic distribution θ_u. Events are formed through the Recurrent Chinese Restaurant Process with the duration-based discount, and each event k also has a multinomial word distribution ψ_k.

During the t-th epoch, for the i-th tweet, a binary variable $y_{t,i}$ is first sampled from a user-specific Bernoulli distribution $\pi_{u_{t,i}}$, which indicates a user's tendency to post topical or event-related tweets. If $y_{t,i}$ equals 0, a topic $z_{t,i}$ is sampled from the user's topic distribution $\theta_{u_{t,i}}$. Then all words in this tweet are sampled from the word distribution

- For each topic $a = 1, \ldots, A$
  - draw $\phi_a \sim \text{Dirichlet}(\beta)$
- For each user $u = 1, \ldots, U$
  - draw $\theta_u \sim \text{Dirichlet}(\gamma), \pi_u \sim \text{Beta}(\tau)$
- For each t and each i
  - draw $y_{t,i} \sim \text{Bernoulli}(\pi_{u_{t,i}})$
  - if $y_{t,i} = 0$
    - draw $z_{t,i} \sim \text{Discrete}(\theta_{u_{t,i}})$
    - for all j, draw $w_{t,i,j} \sim \text{Discrete}(\phi_{z_{t,i}})$
  - if $y_{t,i} = 1$
    - draw $s_{t,i}$ from the RCRP with discount
    - if $s_{t,i}$ is a new event
      - draw $\psi_{s_{t,i}} \sim \text{Dirichlet}(\beta)$
      - set $\bar{t}_{s_{t,i}}$ equal to t
    - for all j, draw $w_{t,i,j} \sim \text{Discrete}(\psi_{s_{t,i}})$

Figure 2: The generative process of our model.

$\phi_{z_{t,i}}$. If $y_{t,i}$ equals 1, then an event $s_{t,i}$ is sampled from a Recurrent Chinese Restaurant Process with the proposed duration-based discount. All words in this tweet are then sampled from the word distribution $\psi_{s_{t,i}}$.

We place uniform Dirichlet priors over all the multinomial distributions. The generative process is also described in Figure 2.

A major difference between the RCRP-LDA models in [2] and [18] and our model is that the RCRP-LDA models differentiate between topics and events at the *word level*, i.e. they allow a document to contain both topical words and event-specific words, whereas in our model the entire content of a tweet is either topical or event-related.Our preliminary experiment shows that when we apply such a setting to Twitter, many tweets end up containing only topical words but are still wrongly assigned to some event which is not related to the content of the tweet. We therefore differentiate between topics and events at the tweet level instead. Also, we do not consider named entities as [2] do because NER on Twitter is less accurate and faces more name variations.

3.6 Model Inference We use collapsed Gibbs sampling to obtain samples of the latent variables based on the conditional distributions derived from our model and finally use these samples to obtain the final hidden label assignment. We find that the conditional probabilities derived from our model are rather complex. This is because unlike the Chinese Restaurant Process, where items are exchangeable, or the Recurrent Chinese Restaurant Process, where items within the same epoch are exchangeable, our model lacks complete exchangeability because of the duration-based discount. While we are able to derive the exact formulas for the conditional probabilities, we find that in terms of efficiency, the exact formulas would incur high computational costs and are not feasible given the large volume of tweets. We then opt for some approximation of the exact sampling formu-

	$\bar{t}_k < t$		$\bar{t}_k = t$		$\bar{t}_k = t+1$	k is a new event
	$n_{k,t} > 0$	$n_{k,t} = 0$	$i_k < i$	$i_k > i$		
$N_{k,t}$	$(n_{k,t-1} + n_{k,t}) \cdot (n_{k,t} + n_{k,t+1}) \cdot \frac{1}{n_{k,t}} \cdot e^{-\lambda(t-\bar{t}_k)}$	$n_{k,t-1} \cdot e^{-\lambda(t-\bar{t}_k)}$	$n_{k,t} + n_{k,t+1}$	$n_{k,t} + n_{k,t+1}$	$n_{k,t+1}$	$\mathfrak{S}(t,i)$
$O_{k,t}$	1	1	1	$\frac{\mathfrak{S}(t,i)}{\mathfrak{S}(\bar{t}_k,i_k)}$	$\frac{\mathfrak{S}(t,i)}{\mathfrak{S}(\bar{t}_k,i_k)} \cdot e^{-n_{k,(.)}}$	1
$\eta_{k,t}$	$\prod_{\substack{\{k' \mid \bar{t}_{k'}=t+1 \\ \| \bar{t}_{k'}=t, i_k > i\}}} 1 + \frac{1-e^{-\lambda(t_{k'}-t_k)}}{\mathfrak{S}(\bar{t}_{k'},i_{k'})}$		$\prod_{\{k' \mid \bar{t}_{k'}=t+1\}} 1 + \frac{1-e^{-\lambda}}{\mathfrak{S}(\bar{t}_{k'},i_{k'})}$		$\prod_{\{k' \mid \bar{t}_{k'}>t\}} \frac{\mathfrak{S}(\bar{t}_{k'},i_{k'})^{(t,i)}}{\mathfrak{S}(\bar{t}_{k'},i_{k'})}$	1

Table 2: For the formula of sampling events, $N_{k,t}$ $O_{k,t}$ and $\eta_{k,t}$ vary under different conditions.

las. We remove the terms that do not affect the probabilities much and keep the terms that dominate the probability mass. In the content that follows, we first derive the exact formulas for conditional probabilities in detail and then describe the approximation.

For the exact conditional probabilities, we jointly sample $y_{t,i}$, $z_{t,i}$ and $s_{t,i}$. The formulas for $y_{t,i} = 0, z_{t,i} = a$ and $y_{t,i} = 1, s_{t,i} = k$ are different.

Topical: First of all, for $y_{t,i} = 0$ and $z_{t,i} = a$, we have the following formula:

$$p(y_{t,i} = 0, z_{t,i} = a | \mathbf{y}_{\neg(t,i)}, \mathbf{z}_{\neg(t,i)}, \mathbf{w}) \propto \frac{n^{(\pi)}_{u,0} + \tau}{n^{(\pi)}_{u,(.)} + 2\tau} \cdot \frac{n^{(\theta)}_{u,a} + \gamma}{n^{(\theta)}_{u,(.)} + A\gamma} \cdot \frac{\prod_{v=1}^{V} \prod_{l=0}^{E_{(v)}-1} (n^{(\phi)}_{a,v} + l + \beta)}{\prod_{l=0}^{E_{(.)}-1} (n^{(\phi)}_{a,(.)} + l + V\beta)},$$

where we use u to represent author $u_{t,i}$. $n^{(\pi)}_{u,0}$ is the number of topical tweets by user u, and it stems from integrating out user's Bernoulli distribution π_u. $n^{(\pi)}_{u,(.)}$ is the total number of tweets by user u. Similarly, $n^{(\theta)}_{u,a}$ is the number of tweets assigned to topic a for this user, resulting from integrating out user's topic distribution θ_u. $n^{(\theta)}_{u,(.)}$ is the same as $n^{(\pi)}_{u,0}$. $E_{(v)}$ is the number of times word type v appears in the current tweet, and $E_{(.)}$ is the total number of words in the current tweet. $n^{(\phi)}_{a,v}$ is the number of times word type v is assigned to topic a, and $n^{(\phi)}_{a,(.)}$ is the number of words assigned to topic a. Note that we calculate all these counting matrixes without considering the current tweet.

Event-related: Then for $y_{t,i} = 1$ and $s_{t,i} = k$, we use the following formula:

$$p(y_{t,i} = 1, s_{t,i} = k | \mathbf{y}_{\neg(t,i)}, \mathbf{s}_{\neg(t,i)}, \mathbf{w}) \propto \frac{n^{(\pi)}_{u,1} + \tau}{n^{(\pi)}_{u,(.)} + 2\tau} \cdot N_{k,t} \cdot O_{k,t} \cdot \eta_{k,t} \cdot \frac{\prod_{v=1}^{V} \prod_{l=0}^{E_{(v)}-1} (n^{(\psi)}_{k,v} + l + \beta)}{\prod_{l=0}^{E_{(.)}-1} (n^{(\psi)}_{k,(.)} + l + V\beta)}$$

where $n^{(\pi)}_{u,1}$ is the number of event-related tweets by user u, $n^{(\psi)}_{k,v}$ is the number of times word type v is assigned to event k, and $n^{(\psi)}_{k,(.)}$ is the total number of words assigned to event k. These word counters stem form integrating out each event's word distribution, and are set to zero when k is a new born event.

In Table 2, we show the values of $N_{k,t}$ $O_{k,t}$ and $\eta_{k,t}$ under various conditions. These conditions are based on the temporal relation between the current tweet and the candidate event k. Here $n_{k,t}$ is the number of tweets in epoch t assigned to event k, excluding the current tweet. $\Delta^{(i)}_{k,t}$ is as we defined before, i_k is the index of the tweet that started event k in epoch t_k, and $n_{k,(.)} = \sum_{t'=\bar{t}_k}^{T} n_{k,t'}$. To simplify the formula, we use $\mathfrak{S}(t,i)$ to represent $\sum_{k'} \Delta^{(i)}_{k',t} + \alpha$, which reflects the probability to start a new table for the i-th document in epoch t.

Roughly speaking, $N_{k,t}$ contains two factors: (1) The size of event k around epoch t. (2) The time difference between the current time stamp t and the event's start time $\bar{t}_k$. $O_{k,t}$ considers the effect of replacing the cluster starter (the i_k-th tweet in epoch $\bar{t}_k$) with the current tweet. Finally, $\eta_{k,t}$ considers how the current event assignment affects the events which emerge later than the current tweet. In particular, in the condition when $\bar{t}_k$ equals $t+1$, assigning the current tweet to event k will bring the start date $\bar{t}_k$ forward, and $\mathfrak{S}(\bar{t}_{k'}, i_{k'})^{(t,i)}$ is calculated[2] after setting $\bar{t}_k$ to t.

Approximation: Given the exact conditional probabilities as the previous formulas show, we opt to approximate the formulas by ignoring the factor $\eta_{k,t}$. We omit this influence factor because we find that it has a minor effect on the probability mass but largely increases the computational complexity. After using such approximation, the complexity of our model is similar to a degenerate variation of our model (one of the baselines in section 4.2), in which d-RCRP is replaced with RCRP. The differences are: (1) when sampling the i-th tweet at epoch t, our model need to record and track the latent event variables of previous tweets in the same epoch to calculate $\mathfrak{S}(t,i)$; (2) when the i-th tweet at

[2] $\mathfrak{S}(\bar{t}_{k'}, i_{k'})^{(t,i)}$ will be affected because of the factor $\Delta^{(i_{k'})}_{k,\bar{t}_{k'}}$. Since the start time $\bar{t}_k$ is changed from $t+1$ to t, the value of $\Delta^{(i_{k'})}_{k,\bar{t}_{k'}}$ should be updated to $(n_{k,\bar{t}_{k'}-1} + n_{k,\bar{t}_{k'}})(1 - e^{-\lambda(\bar{t}_{k'}-t)})$.

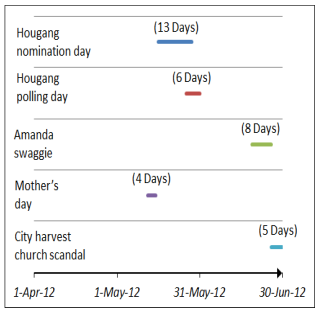
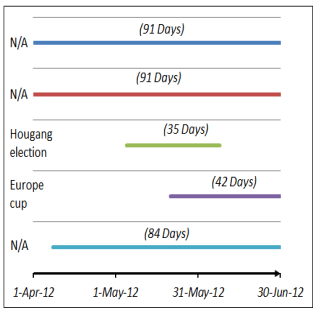
Events	Top words	Life cycle	Events	Top words	Life cycle
Hougang nomination day	#hougangbyelection, hougang, wp, desmond, png		N/A (Malay)	yg, di, yang, aku, dan	
Hougang polling day	#hougangbyelection, hougang, pap, png, desmond		N/A	singapore, prices, oil, asian, stocks	
Amanda swaggie	singapore, amanda, bieber, europe, trending		Hougang election	#hougangbyelection, hougang, wp, desmond, pap	
Mother's day	day, happy, mother's, mothers, love, mom		Europe cup	#euro2012, spain, portugal, euro, germany, italy	
City harvest church scandal	city, harvest, church, kong, founder		N/A	news, home, usa, run, blog	

Table 3: Top five events detected by d-RCRP (left) and RCRP (right). We show each event's name (manually given and N/A indicates a meaningless event), top ranked words, and life cycle (the duration of the event).

epoch t starts an event during the previous iteration, we need to search for the nearest tweet which belongs to the same event to start the event. Although we use this approximated formula for the exact conditional probabilities, we find that in our experiments the formula works fine and generates meaningful results.

4 Experiment

4.1 Dataset We use a Twitter dataset that was previously used in [10] for finding bursty topics. The original dataset contains the tweets published by a large number of Singapore Twitter users. Since the entire dataset is huge, we pick 500 users, including 13 news media users, 2 journalists and 485 random users. We use their tweets between April 1 and June 30, 2012 for our experiments. We use the CMU Twitter POS Tagger[3] to tag these tweets and remove the non-standard words (i.e. words tagged as punctuation marks, emoticons, urls, at-mentions, pronouns, etc.) and stop words. Tweets with less than three words are also discarded. In the end we get 701,878 tweets in total.

4.2 Quantitative Evaluation In the experiments below, we refer to our own model as d-RCRP. We quantitatively evaluate d-RCRP by comparing it with two baseline models:
RCRP: This is a modified version of our own model where we remove the duration-based probability discount, i.e. we use the standard RCRP. Comparison with this model helps us understand the effect of the duration-based discount.
TimeUserLDA: This model is from [10]. Similar to d-RCRP, TimeUserLDA also separates personal topical tweets from event-related tweets. However, it groups the event-related tweets into a *fixed* number of bursty topics and then uses a two-state machine in a postprocessing step to identify events from these bursty topics, i.e. events are not directly modeled within the generative process itself. In contrast, d-RCRP and RCRP directly models events.

It is worth mentioning that both baselines separate topical tweets and event related tweets. We do not compare with the model in [2] or [18] because these methods are designed for news-centric data and treat all documents as event-related. The results of both model are poor as seen from Table 1 in Section 1.

For the parameter settings, we empirically set A to 80, γ to $\frac{50}{A}$, β to 0.01, τ to 1, and α to 1. The duration-based discount parameter λ is set to 1. We run 300 iterations before we collect 10 samples with a gap of 5 iterations to obtain our final latent variable assignment.

4.2.1 Event Quality We first analyze the quality of the detected events. For each method, we rank the detected events based on the number of tweets assigned to them and then pick the top-30 events for each method. We randomly mix these events and ask two human judges to label them. For each event, the judges are provided with 100 randomly selected tweets (or all tweets if an event contains less than 100 tweets) together with their time stamps. The judges are allowed to use external sources to help them. An event is scored 1 if the 100 tweets coherently describe an event or 0 otherwise. The inter-annotator agreement score is 0.639 using Cohen's kappa. The final score of an event is 1 if both judges have scored it 1. Table 4 shows the performance in terms of Precision@K, and Table 3 shows the top five events detected by d-RCRP and RCRP respectively. The results show that our model outperforms the others consistently.

Method	P@5	P@10	P@20	P@30
d-RCRP	**1.000**	**1.000**	**1.000**	**0.800**
RCRP	0.400	0.500	0.600	0.600
TimeUserLDA	**1.000**	0.900	0.800	0.667

Table 4: Precision@K for the various models.

A close examination of the events reveals that RCRP identifies several events that are longstanding general topics

[3] http://www.ark.cs.cmu.edu/TweetNLP/

(as Table 3 shows), which verifies that burstiness is an important factor to consider for identifying events on Twitter. It is interesting to see that TimeUserLDA outperforms RCRP. We believe that it is because TimeUserLDA also considers burstiness. However, TimeUserLDA requires a postprocessing step whereas d-RCRP achieves event identification inside the generative model itself.

4.2.2 Tweet Quality The evaluation above is at event-level. We also want to evaluate the relevance of the tweets assigned to each event. To make fair comparison, we select common events identified by all three methods. We further ask two human judges to score the 100 tweets as either 1 or 0 based on their relevance to the event. We obtain a Cohen's kappa of 0.760, which shows high agreement. Table 5 shows the precision of the tweets for all 5 common events. We find that for 4 of them, our model obtains the highest precision. The false positive tweets by RCRP are mislabeled mainly because the duration of the event tends to be long. For example, several tweets about Labor Day are clustered into the event of Mother's Day. The false positive tweets by TimeUserLDA are the ones with related words. For example, several "happy birthday" tweets are clustered into the event of Father's Day. For April Fools, after we take a close look at the corresponding tweets, we find that our model does not outperform other models mainly because most tweets of this event adopt similar words, such as "aprilfool", "fraud" and "prank", which are quite distinctive and can separate the relevant tweets from other general tweets. Roughly speaking, TimeUserLDA performs well when the event is globally popular (i.e. festivals, or some major events) and the words of the event are distinctive.

Event	d-RCRP	RCRP	TimeUserLDA
Amanda swaggie	**0.91**	0.88	0.79
Mother's day	**0.86**	0.82	0.77
April fools	0.85	0.85	**0.97**
City harvest church scandal	**0.86**	0.85	0.82
Father's day	**0.86**	0.77	0.65

Table 5: Precision of tweets for the 5 common events.

4.2.3 Temporal Profile Quality Besides the quality of the top-ranked events and their tweets, we also evaluate the temporal profiles of the events. Essentially the temporal profile of an event shows how the number of tweets related to an event changes over time. As it is hard for us to obtain the ground truth of the temporal profile of an event through human judgment we use hashtags to help us [11]. Twitter users create specific hashtags when significant events happen. These hashtags are widely used because of the diffusion effect on Twitter's huge network. We rank the hashtags in our data set based on their numbers of tweets. From the top-ranked ones, we pick 7 hashtags that are related to some meaningful events. We obtain a temporal profile of each of these hashtags based on the daily tweet counts. Our hypothesis is that this is close to the real temporal profile of the corresponding event. Then for each of the methods we consider, we pick the corresponding event for each hashtag and also obtain a temporal profile based on the daily tweet counts returned by that method. Finally, we convert the temporal profiles into distributions over time through normalization, and for each hashtag and each method, we compute the JS-divergence between the two distributions, one based on the hashtag and the other based on the method. We believe that the lower the JS-divergence is, the better the temporal profile of an event obtained by the method matches the ground truth. Table 6 shows the results. We can see that d-RCRP consistently gives lower JS-divergence scores than the other two methods except for #aprilfools. It shows that the tweets identified by d-RCRP for an event usually better reflect the real evolution of the event on Twitter.

HashTag/Event	d-RCRP	RCRP	TimeUserLDA
#ss4encore	**0.0282**	0.0448	N/A
#bigbangmonster	**0.0055**	0.1749	N/A
#ss4shanghai	**0.0004**	0.0738	N/A
#3years2ne1	**0.0344**	0.0616	N/A
#chc	**0.0419**	0.0465	0.1443
#aprilfools	0.0797	0.0882	**0.0656**
#getwellsoongaga	**0.1416**	0.2178	N/A

Table 6: The JS-divergence scores of the three methods. N/A means there is no corresponding event.

4.3 Qualitative Evaluation In this section, we show some example results from our experiments that illustrate the advantages of our proposed model. Moreover, we can do various event-centric analyses (i.e. users' tendency to tweet about events, event-topic correlation), because our unified model considers both personal interests and events on Twitter. These analyses help better interpret events in Twitter.

Events	Start date	Top words
candidate announcement	10 May	hougang, choo, desmond, png, candidate
nomination day	16 May	#hougangbyelection, hougang, wp, desmond, png,
polling day	26 May	#hougangbyelection hougang, pap, png, desmond,

Table 7: Case study on Hougang by-election.

4.3.1 Case Study For events that span a relatively long duration, our model tends to identify the most significant sub-events and treat these sub-events as events. For example, our data set covers the Singapore Hougang by-election,

which lasted for around twenty days. There were three major events during this period: Election candidates were announced on May 10, the nomination day was on May 16, and the polling day was on May 26. Table 7 shows that our model correctly finds these major sub-events.

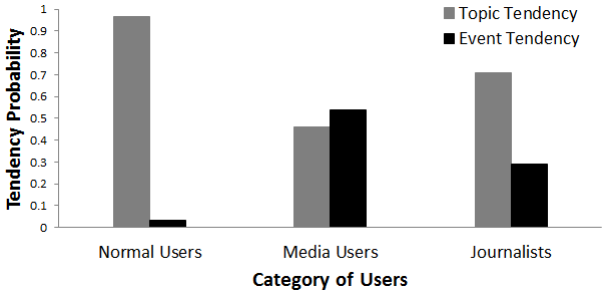


Figure 3: User's tendency to tweet on topics or events.

4.3.2 User Analysis Since our model learns a user's tendency to tweet about topics or events, we can compare such tendency of normal users, media users (e.g. YahooNews) and journalists in our data. For each category of users, we average their Bernoulli parameter π_u and show the results in Figure 3. We can clearly see that media users are more likely to tweet about events compared with normal users. It may appear strange that media users also have a high probability to tweet about topics. This is because many news events tweeted by media users do not attract much attention on Twitter, and therefore these news events are not identified as popular events on Twitter but become general topics by our model. We also find that journalists' tendency to tweet about events lies between normal users and media users, which makes sense because journalists play dual roles as both a normal user and a media user.

4.3.3 Event-Topic Correlation Analysis Our model does not directly model the correlation between events and topics. However, we expect that some events are more related to certain topics than others and therefore more likely to be tweeted by users interested in those topics. E.g. a Korean pop music concert is more related to the general topic on music or entertainment while the event on Eurocup is more related to the topic on sports. We can find such correlations through the following postprocessing. First, we average the topic distributions of all normal users to obtain a background topic distribution of our data. Denote this as θ_{B}. Then for each event, we obtain all users who have tweeted about the event and average these users' topic distributions. We thus obtain a topic distribution θ_k for each event k. By measuring the JS-divergence between θ_{B} and θ_k, we can rank the events.

We show 19 events tweeted by at least 20 users in increasing order of the JS-divergence scores in Table 8. We can see that the top-ranked events (with low JS-divergence) are those that tend to be tweeted by all users, while the low-ranked ones (with high JS-divergence) are those that are more related to certain topics than others and therefore tend to be tweeted by a subgroup of users. E.g. event 19 is about Super Junior, a Korean idol group, and this event is likely to be only interesting to K-pop fans. By analyzing the correlation between events and topics, we can potentially recommend relevant events to a user based on her topic interests.

Rank	Event Name	Score
1	Mother's day	0.0068
2	Father's day	0.0073
3	Indonesia tsunami	0.0106
4	April fool's day	0.0113
5	Tsunami hit Singapore	0.0114
6	Alex push old lady	0.0162
7	Amanda swaggie	0.0170
8	Ferrari accident	0.0198
9	City harvest church scandal	0.0231
10	Staraward(Rui En)	0.0259
11	Hougang election polling day	0.0263
12	Hougang election nomination day	0.0263
13	Bigbang concert ticket sell	0.0320
14	Bigbang album "Monster"	0.0328
15	Euro cup 2012	0.0341
16	Mozambique fashion week	0.0354
17	Staraward(Jay Park)	0.0362
18	LionsXII 9-0 Sabah FA	0.0543
19	Super Junior new album	0.0805

Table 8: Events ranked based on JS-divergence.

5 Conclusions

In this paper, we study the problem of event identification from Twitter stream. The Recurrent Chinese Restaurant Process is appealing for our task because it provides a principled dynamic non-parametric model. However, our preliminary experiment shows that RCRP is not directly applicable in our task for two reasons: (1) events emerge and die out fast on Twitter, (2) most tweets are topical and only a small proportion of them are event-related. Therefore, we propose a novel duration-based probability discount to RCRP to capture the burstiness character of events on Twitter. We then propose a probabilistic model to identify both events and topics simultaneously from Twitter. Our experiments demonstrate that our proposed model can identify events accurately, which shows the effectiveness of duration-based discount. Finally, we qualitatively show some interesting studies on users and event-topic correlations.

6 Acknowledgments

This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office, Media Development Authority (MDA).

References

[1] C. C. Aggarwal and P. S. Yu. A framework for clustering massive text and categorical data streams. In *Proceedings of the SIAM International Conference on Data Mining*, 2006.

[2] A. Ahmed, Q. Ho, J. Eisenstein, E. Xing, A. J. Smola, and C. H. Teo. Unified analysis of streaming news. In *Proceedings of the 20th International Conference on World Wide Web*, pages 267–276, 2011.

[3] A. Ahmed and E. P. Xing. Dynamic non-parametric mixture models and the recurrent Chinese restaurant process: with applications to evolutionary clustering. In *Proceedings of the SIAM International Conference on Data Mining*, 2008.

[4] J. Allan. *Introduction to Topic Detection and Tracking*, pages 1–16. 2002.

[5] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–45, 1998.

[6] H. Becker, M. Naaman, and L. Gravano. Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media*, 2011.

[7] D. Blackwell and J. MacQueen. Ferguson distributions via Polya urn schemes. *The Annals of Statistics*, pages 353–355, 1973.

[8] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 113–120, 2006.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, pages 993–1022, 2003.

[10] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 536–544, 2012.

[11] L. Hong, B. Dom, S. Gurumurthy, and K. Tsioutsiouliklis. A time-dependent topic model for multiple text streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 832–840, 2011.

[12] J. Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 91–101, 2002.

[13] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[14] C. Li, A. Sun, and A. Datta. Twevent: segment-based event detection from tweets. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 155–164, 2012.

[15] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to Twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189, 2010.

[16] S. Petrović, M. Osborne, R. McCreadie, R. Macdonald, R. Ounis, and L. Shrimpton. Can twitter replace newswire for breaking news? In *Proceedings of the 7th International Conference on Weblogs and Social Media*, 2013.

[17] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, pages 851–860, 2010.

[18] X. Tang and C. C. Yang. TUT: a statistical model for detecting trends, topics and user interests in social media. In *Proceedings of 21st ACM Conference on Information and Knowledge Management*, 2012.

[19] X. Wang and A. McCallum. Topics over time: a non-Markov continuous-time model of topical trends. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 972–981, 2006.

[20] X. Wang, C. Zhai, X. Hu, and R. Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 784–793, 2007.

[21] J. Weng and B.-S. Lee. Event detection in Twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media*, 2011.

[22] W. Xie, F. Zhu, J. Jiang, E.-P. Lim, and K. Wang. Topicsketch: Real-time bursty topic detection from twitter. In *Proc. of the 13th IEEE International Conference on Data Mining*, 2013.

[23] Y. Yang, J. Carbonell, R. Brown, T. Pierce, B. Archibald, and X. Liu. Learning approaches for detecting and tracking news events. *Intelligent Systems and their Applications, IEEE*, pages 32–43, 1999.

[24] Y. Yang, T. Pierce, and J. Carbonell. A study of retrospective and on-line event detection. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 28–36, 1998.