A Probabilistic Model for Bursty Topic Discovery in Microblogs

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, Jun Xu, Xueqi Cheng
Institute of Computing Technology, Chinese Academy of Science
No.6 Kexueyuan South Road, Haidian District
Beijing, China 100190

Abstract

Bursty topics discovery in microblogs is important for people to grasp essential and valuable information. However, the task is challenging since microblog posts are particularly short and noisy. This work develops a novel probabilistic model, namely Bursty Biterm Topic Model (BBTM), to deal with the task. BBTM extends the Biterm Topic Model (BTM) by incorporating the burstiness of biterms as prior knowledge for bursty topic modeling, which enjoys the following merits: 1) It can well solve the data sparsity problem in topic modeling over short texts as the same as BTM; 2) It can automatically discover high quality bursty topics in microblogs in a principled and efficient way. Extensive experiments on a standard Twitter dataset show that our approach outperforms the state-of-the-art baselines significantly.

Introduction

Nowadays microblog services have become an important platform for people to share and access information. In Twitter, about 58 million posts are produced every day, involving various topics such as daily chatting, business promotions, and news stories. Among them, there are always many novel topics emerging and attracting wide interest, referred as bursty topics. These topics are often related to some important events or issues in either cyber or physical space. Thus discovering bursty topics can provide people essential and valuable information, and consequently benefit many related applications such as public opinion analysis, business intelligence, and news clues tracking.

However, bursty topic discovery in microblogs is challenging. First, the posts in microblogs are particularly short. How to distill high quality topics from short texts is a non-trivial problem. Second, posts are particularly diverse and noisy, with a large proportion of common and meaningless subjects such as pointless babbles and daily chatting (Analytics 2009). These posts overwhelm in microblogs, making it difficult to distinguish bursty topics from non-bursty content.

In previous studies, a typical way for this task is to detect bursty features (e.g., words or phrases) and then cluster them (Mathioudakis and Koudas 2010; Cataldi, Di Caro, and Schifanella 2010; Li, Sun, and Datta 2012). However, there are two drawbacks within these methods. First, they require cumbersome and complicated heuristic tuning and post-processing, since the bursty features detected are noisy and ambiguous, which are not easy to cluster. Second, representing the topics only by bursty features will lose much information, making them difficult to read and understand.

Another attempt is to discover bursty topics via topic models, a widely used tool for topic discovery in text collections (Hofmann 1999; Blei, Ng, and Jordan 2003). However, conventional topic models are designed to reveal the main topics in a collection (Blei 2012), not directly applicable for bursty topic discovery in microblogs. Although some post-processing techniques can be used to detect bursty topics from the learned topics of conventional topic models (Lau, Collier, and Baldwin 2012), it is not economical since most of the discovered topics might not be bursty. To amend this problem, some researchers tried to introduce the temporal information into topic models (Diao et al. 2012; Yin et al. 2013). Unfortunately, they still rely on post-processing steps or heuristic techniques to distill bursty topics. Furthermore, all of these above methods use topic models designed for normal texts (e.g., LDA), which have been shown not effective for short texts such as microblog posts (Hong and Davison 2010; Yan et al. 2013).

In this paper, we focus on the problem of discovering bursty topics in a microblog stream divided by certain time slices (e.g., day). Formally, a topic is considered to be bursty in a time slice if it is heavily discussed in that time slice, but not in most of other time slices. We propose to discover such bursty topics in a principled way via a novel probabilistic model, namely Bursty Biterm Topic Model (BBTM). Our work is based on a recently introduced Biterm Topic Model (BTM) (Yan et al. 2013), which models biterms (i.e., word pairs) rather than words for effective topic modeling in short texts. The key idea of our approach is to exploit the burstiness of biterms as prior knowledge to incorporate into BTM for bursty topic modeling. BBTM enjoys two substantial merits over the previous methods. First, it well solves the data sparsity problem in topic modeling over short texts, as compared with those methods based on conventional topic models. Second, it can learn bursty topics in a principled and efficient way without any heuristic post-processing steps.

We have conducted extensive experiments over a stan-

dard Twitter dataset. The experimental results demonstrate that our approach achieves substantial improvement over the state-of-the-art methods.

Biterm Topic Model

For completeness, we first briefly review the biterm topic model (i.e., BTM) (Yan et al. 2013; Cheng et al. 2014), a recently proposed probabilistic topic model for short texts. Before BTM, most conventional topic models, such as PLSA (Hofmann 1999) and LDA (Blei, Ng, and Jordan 2003), model each document as a mixture of topics, and thus suffer from the data sparsity problem when documents are extremely short (Hong and Davison 2010; Tang et al. 2014). Instead, BTM learns topics by modeling the generation of biterms (i.e., unordered co-occurring word pairs) in the collection, whose effectiveness are not affected by the length of documents, making it more appropriate for short texts.

The intuition of BTM is that if two words co-occur more frequently, they are more likely to belong to a same topic. Based on this idea, BTM models each biterm as two words draw from a same topic, while a topic is drawn from a mixture of topics over the whole collection. Specifically, given a short text collection, suppose it contains N_B biterms $\mathbb{B} = \{b_1, ..., b_{N_B}\}$ where $b_i = (w_{i,1}, w_{i,2})$, and K topics expressed over W unique words, the generative process described by BTM is as follows:

1. For the collection,
   - draw a topic distribution $\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha})$
2. For each topic z,
   - draw a word distribution $\boldsymbol{\phi}_z \sim \mathrm{Dir}(\boldsymbol{\beta})$
3. For each biterm $b_i \in \mathbb{B}$,
   - draw a topic assignment $z \sim \mathrm{Multi}(\boldsymbol{\theta})$
   - draw two words $w_{i,1}, w_{i,2} \sim \mathrm{Mulit}(\boldsymbol{\phi}_z)$

where $\boldsymbol{\theta}$ defines a K-dimensional multinomial distribution over topics, and $\boldsymbol{\phi}_z$ defines a W-dimensional multinomial distribution over words. The graphical representation of BTM is illustrated in Figure 1 (a).

Bursty Topic Modeling

BTM is an effective topic model over short texts but not designed for bursty topic discovery. In other words, each biterm occurrence contributes equally in BTM, but in microblogs a large proportion of biterms in microblogs are about common topics such as daily life and chatting. Consequently, BTM tends to discover common topics in microblogs.

To discover bursty topics through BTM, it is important to emphasize those biterms relevant to bursty topics and make the model focus on these observations. Therefore, we introduce and quantify the burstiness of biterms and incorporate it as prior knowledge into BTM for bursty topic discovery.

Bursty Probability of Biterm

Intuitively, when a bursty topic breaks out, relevant biterms can be observed more frequently than usual. For instance, the biterms such as "world cup", "football brazil" became much more popular than usual in Twitter when World Cup 2014 took place. Such biterms provide us crucial clues for bursty topic discovery. Based on the above observation, we introduce a probability measurement, called *bursty probability*, to quantify the burstiness of biterms, which can be estimated from the temporal frequencies of the biterms.

Suppose a biterm b occurred $n_b^{(t)}$ times in the posts published in time slice t. Since a biterm might be observed either in normal usage (e.g., daily chatting) or in some bursty topic, we decompose $n_b^{(t)}$ into two parts: $n_{b,0}^{(t)}$ is the count of biterm b occurred in normal usage, while $n_{b,1}^{(t)}$ is the count of b occurred in bursty topics, with $n_{b,0}^{(t)} + n_{b,1}^{(t)} = n_b^{(t)}$. Note that both $n_{b,0}^{(t)}$ and $n_{b,1}^{(t)}$ are not observed, however, we can determine their value approximately based on the temporal frequencies of b.

Specifically, for a large collection it is reasonable to assume that the normal usage of a biterm is stable during a period of time. In other words, $n_{b,0}^{(t)}$ is supposed to almost be constant over time. Conversely, $n_{b,1}^{(t)}$ may change significantly across different time slices. When some bursty topic relevant to b_i breaks out, $n_{b,1}^{(t)}$ might rise steeply. while in most other time slices, there is no such bursty topic taking place, $n_{b,1}^{(t)}$ will be close to 0. Based on the above analysis, we estimate $n_{b,0}^{(t)}$ by the mean of $n_b^{(t)}$ in the last S time slices, i.e., $\bar{n}_b^{(t)} = \frac{1}{S}\sum_{s=1}^{S} n_b^{(t-s)}$. Consequently, we can obtain $\hat{n}_{b,1}^{(t)} = (n_b^{(t)} - \bar{n}_b^{(t)})_+$, where $(x)_+ = \max(x, \epsilon)$, and ϵ is a small positive number to avoid zero probability. In our experiments, we set $S = 10$, $\epsilon = 0.01$ after some preliminary experiments.

With $n_b^{(t)}$ and $\hat{n}_{b,1}^{(t)}$ in hand, it is straightforward to measure the possibility of b generated from a bursty topic in time slice t as:

$$\eta_b^{(t)} = \frac{(n_b^{(t)} - \bar{n}_b^{(t)})_+}{n_b^{(t)}}. \quad (1)$$

We refer $\eta_b^{(t)}$ as the bursty probability of biterm b in time slice t. The calculation of $\eta_b^{(t)}$ implies that a biterm occurred much more frequently in a time slice than other time slices will be more likely to be generated from bursty topics [1].

Bursty Biterm Topic Models

We now describe our approach for bursty topic modeling in microblogs, i.e., Bursty Biterm Topic Model (BBTM). In the following, since we focus on data in a single time slice, we

[1] In our experiments, we found $\eta_b^{(t)}$ in Eq. (1) might be overestimated for rare biterms whose $\bar{n}_b^{(t)}$ is small. Since the rare biterms are more likely to be generated by random factors rather than bursty topics, we set $\eta_b^{(t)}$ to ϵ if $\bar{n}_b^{(t)} < 5$.

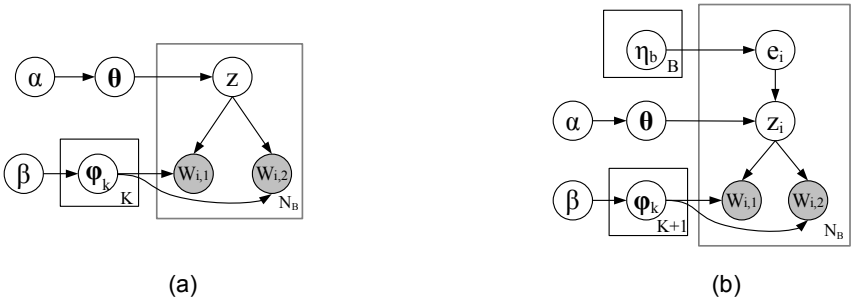

Figure 1: Graphical representation of (a) the biterm topic model, and (b) the bursty biterm topic model.

will not specify the time slice in the notations. For instance, we write $\eta_b^{(t)}$ as η_b for simplification.

As we suppose that a biterm might be observed either in normal usage or in some bursty topic, the basic idea of BBTM is to distinguish the occurrences of biterms from the two parts to learn bursty topics. Specifically, we define a binary variable e_i to denote the source of an occurrence of biterm b_i. $e_i = 0$ indicates b_i is generated by normal usage, while $e_i = 1$ indicates b_i is generated from some bursty topic. Recall that the bursty probability of a biterm encodes our prior knowledge of how likely the biterm is generated from a bursty topic, we thus define a Bernoulli distribution with parameter η_{b_i} as the prior distribution of e_i. Moreover, we introduce K multinomial distributions over the words (i.e., $\{\boldsymbol{\phi}_k | k \in [1, K]\}$) to denote the bursty topics in the collection, and a background word distribution $\boldsymbol{\phi}_0$ to denote the normal usage. The generative process of the biterm set $\mathbb{B}$ in the time slice in BBTM is then defined as follows:

1. For the collection,
   - draw a bursty topic distribution $\boldsymbol{\theta} \sim \mathrm{Dir}(\alpha)$
   - draw a background word distribution $\boldsymbol{\phi}_0 \sim \mathrm{Dir}(\beta)$
2. For each bursty topic $k \in [1, K]$,
   - draw a word distribution $\boldsymbol{\phi}_k \sim \mathrm{Dir}(\beta)$
3. For each biterm $b_i \in \mathbb{B}$
   - draw $e_i \sim \mathrm{Bern}(\eta_{b_i})$
   - If $e_i = 0$,
     - draw two words $w_{i,1}, w_{i,2} \sim \mathrm{Multi}(\boldsymbol{\phi}_0)$
   - If $e_i = 1$,
     - draw a bursty topic $z \sim \mathrm{Multi}(\boldsymbol{\theta})$
     - draw two words $w_{i,1}, w_{i,2} \sim \mathrm{Multi}(\boldsymbol{\phi}_z)$

Its graphical representation is shown in Figure 1 (b), where B denotes the number of distinct biterms in $\mathbb{B}$.

Discussion

In BBTM, the prior distribution η_b indicates how likely a biterm b is generated by bursty topics, which plays a key role in guiding the model to distinguish whether a biterm b is generated from burst topics or not. Previous work measuring word burstiness with statistical testing (Swan and Allan 1999; Lijffijt 2013) or sigmoid mapping of the variance of temporal frequencies of words (Fung et al. 2005; Li, Sun, and Datta 2012) can only determine whether a biterm is bursty or not, rather than the probability of a biterm generated from bursty topics. Therefore, using them as the prior will lead to inferior results, as shown in our preliminary experiments.

The background word distribution introduced by BBTM is used to filter out biterms not related to bursty topics. In (Mei and Zhai 2005), a background word distribution is also introduced into a topic model to distill temporal themes from a text stream. However, the background word distribution is simply set to the empirical word distribution that contributes equally on the generation of each word. In BBTM, the background word distribution is learned from the data. Its impact on the generation of biterms is different due to the biterm-wise prior η_b.

Parameter Estimation

In BBTM, the parameter set need to estimate is $\boldsymbol{\Theta} = \{\boldsymbol{\theta}, \boldsymbol{\phi}_0, ..., \boldsymbol{\phi}_K\}$, if given the hyperparameters α and β. It is not hard to write out the likelihood of the biterm set $\mathbb{B}$:

$$P(\mathbb{B}|\boldsymbol{\Theta}) = \prod_{i=1}^{N_B} \Big(\phi_{0,w_{i,1}} \phi_{0,w_{i,2}} (1 - \eta_{b_i}) + \sum_{k=1}^{K} \theta_k \phi_{k,w_{i,1}} \phi_{k,w_{i,2}} \eta_{b_i} \Big). \quad (2)$$

Since the parameters in $\boldsymbol{\Theta}$ are coupled in Eq. (2), it is intractable to determine them exactly. Following (Griffiths and Steyvers 2004), we use the collapsed Gibbs sampling algorithm for approximate estimation.

The basic idea is to estimate the parameters alternatively using samples drawn from the posterior distributions of latent variables sequentially conditioned on the current values of all other variables and the data. In BBTM, there are two types of latent variables, i.e., e_i and z_i. We draw them jointly according to the following conditional distribution

Algorithm 1: Gibbs sampling algorithm for BBTM

Input: K, α, β, $\mathbb{B}$
Output: $\{\phi_k\}_{k=0}^{K}$, $\boldsymbol{\theta}$
Randomly initialize $\mathbf{e}$ and $\mathbf{z}$
for $iter = 1$ *to* N_{iter} **do**
  foreach $b_i = (w_{i,1}, w_{i,2}) \in \mathbb{B}$ **do**
    Draw e_i, k from Eqs.(3-4)
    if $e_i = 0$ **then**
      Update the counts $n_{0,w_{i,1}}, n_{0,w_{i,2}}$
    else
      Update the counts $n_k, n_{k,w_{i,1}}, n_{k,w_{i,2}}$
Compute the parameters by Eqs. (5-6)

(the derivation is provided in the supplemental material):

$$P(e_i = 0|\mathbf{e}^{\neg i}, \mathbf{z}^{\neg i}, \mathbb{B}, \alpha, \beta, \boldsymbol{\eta}) \propto (1 - \eta_{b_i}) \cdot \frac{(n_{0,w_{i,1}}^{\neg i} + \beta)(n_{0,w_{i,2}}^{\neg i} + \beta)}{(n_{0,\cdot}^{\neg i} + W\beta)(n_{0,\cdot}^{\neg i} + 1 + W\beta)}, \quad (3)$$

$$P(e_i = 1, z_i = k|\mathbf{e}^{\neg i}, \mathbf{z}^{\neg i}, \mathbb{B}, \alpha, \beta, \boldsymbol{\eta}) \propto \eta_{b_i} \cdot \frac{(n_k^{\neg i} + \alpha)}{(n_\cdot^{\neg i} + K\alpha)} \cdot \frac{(n_{k,w_{i,1}}^{\neg i} + \beta)(n_{k,w_{i,2}}^{\neg i} + \beta)}{(n_{k,\cdot}^{\neg i} + W\beta)(n_{k,\cdot}^{\neg i} + 1 + W\beta)}, \quad (4)$$

where $\mathbf{e} = \{e_i\}_{i=0}^{N_B}$, $\mathbf{z} = \{z_i\}_{i=0}^{N_B}$, $\boldsymbol{\eta} = \{\eta_b\}_{b=0}^{B}$, $n_{0,w}$ is the number of times that word w is assigned to the background word distribution, $n_{0,\cdot} = \sum_{w=1}^{W} n_{0,w}$ is the total number of words assigned to the background word distribution, n_k is the number of biterms assigned to bursty topic k, $n_\cdot = \sum_{k=1}^{K} n_k$ is the total number of biterms assigned to bursty topics, $n_{k,w}$ is the number of times that word w is assigned to bursty topic k, $n_{k,\cdot} = \sum_{w=1}^{W} n_{k,w}$ is the total number of words assigned to bursty topic k, and $\neg i$ means ignoring biterm b_i.

The Gibbs sampling algorithm of BBTM is outlined in Algorithm 1. First, we randomly initialize the latent variables. Then, we iteratively draw samples of the latent variables for each biterm according to Eqs. (3-4). After a sufficient number of iterations, we collect the counts n_k and $n_{k,w}$ to estimate the parameters by:

$$\hat{\phi}_{k,w} = \frac{n_{k,w} + \beta}{n_{k,\cdot} + W\beta}, \quad (5)$$

$$\hat{\theta}_k = \frac{n_k + \alpha}{n_\cdot + K\alpha}. \quad (6)$$

Runtime Analysis. Recall that the time complexity of BTM is $O(N_{iter}KN_B)$ and the memory complexity is $K(1+W)+N_B$ (Yan et al. 2013), where N_B is the number of biterms in $\mathbb{B}$. Compared with BTM, BBTM introduces an additional topic, namely the background word distribution, thus its time complexity is $O(N_{iter}(K+1)N_B)$. Moreover, it need to maintain $\boldsymbol{\eta}$ in memory, so its memory complexity is $(K+1)(1+W)+N_B+B$.

Experiments

In this section, we empirically verify the performance of BBTM on bursty topic discovery in microblogs both quantitatively and qualitatively.

Experimental Settings

Dataset. We use a standard microblog dataset, i.e., the Tweets2011 collection published in TREC 2011 microblog track[2]. The dataset contains approximately 16 million tweets sampled in 17 days from Jan. 23 to Feb. 8, 2011. We preprocessed the raw data in the same way as (Yan et al. 2013). Furthermore, we filtered biterms occurred only one time in the collection to save computational cost since most of them are meaningless.

Baseline Methods. We compare our approach against the following baseline methods: 1) **Twevent** (Li, Sun, and Datta 2012) first detects bursty tweet segments and then clustering them to obtain bursty topics. To make a fair comparison, we simply used individual words as segments, and did not exploit Wikipedia to filter the final clusters. 2) **OLDA** (Lau, Collier, and Baldwin 2012) uses online LDA (AlSumait, Barbará, and Domeniconi 2008) to learn topics in each time slice, and then detects bursty topics by measuring the Jensen-Shannon divergence between the words distribution before and after an update of the topics. 3) **UTM** (User-Temporal Mixture model) (Yin et al. 2013) supposes the temporal topics follow a time-dependent topic distribution, and the non-bursty topics follow a user-dependent topic distribution. To ensure the temporal topics discovered to be bursty, the authors heuristically boosted the probability of bursty words in the temporal topics. 4) **IBTM** trains individual BTM (Yan et al. 2013) for each time slice. To distinguish between bursty topics and non-bursty topics, we first greedily matched the topics in two adjacent time slices according their cosine similarity, and then used the post-processing step in OLDA to detect bursty topics. 5) **BBTM-S** is a simplified version BBTM. In BBTM-S, for each occurrence of biterm b_i, we directly draw e_i from a Bernoulli distribution with parameter η_{b_i}, rather than take it as a latent variable. If $e_i = 1$, b_i is selected into the training set, otherwise it is discarded. Finally, we simply train a BTM over the selected biterms to learn the bursty topics.

Parameter Setting. In our experiments, the length of a time slice is set to a day, a typical setting in the literature (Lau, Collier, and Baldwin 2012; Li, Sun, and Datta 2012). Following the convention in BTM (Yan et al. 2013), we set $\alpha = 50/K$ and $\beta = 0.01$ in BBTM. The number of bursty topics K are varied from 10 to 50. The other parameters of the baseline methods are set by their default values in their papers.

Accuracy of Bursty Topics Discovered

First of all, we evaluate the accuracy of the bursty topics discovered by different methods. We asked 5 volunteers to manually label the bursty topics discovered by all of these methods. To ensure unbiased judgment, all the topics generated are randomly mixed before labeling. For each bursty topic, we provided the volunteers its 50 most probable words and time slice information, and external tools, such as Google and Twitter search, to help their judgement.

[2]http://trec.nist.gov/data/tweets/

Method	P@10	P@30	P@50
Twevent	0.592	0.681	0.636
UTM	0.565	0.488	0.453
OLDA	0.231	0.217	0.185
IBTM	0.300	0.325	0.297
BBTM-S	0.785	0.832	0.790
BBTM	**0.810**	**0.865**	**0.842**

Table 1: Accuracy of the bursty topics discovered (measured by Precision@K).

If the bursty topic presented is both meaningful and bursty in its time slice, it gets 1 point; Otherwise, it gets 0 point. A bursty topic is correctly detected if more than half of judges assigned 1 point to it. The comparison of different methods are then based on the average precision at K ($P@K$), i.e., the proportion of correctly detected busty topics among the learned K bursty topics.

Table 1 lists the precisions of all the methods with different settings of bursty topic number K. We find that 1) BBTM always achieves a high precision over 0.8, which is substantially better than other methods. 2) The simplified version of BBTM, i.e., BBTM-S, falls behind BBTM but works much better than other baselines. We analyze the reason why BBTM-S is worse than BBTM. We find that the biterm sampling process simply using the prior distribution may throw away some potential bursty biterms with moderate busty probability. Meanwhile, from Eq. (4) we can see that the topic assignment of b_i is actually affected by two factors simultaneously. One is the prior knowledge encoded by bursty probability, and the other is co-occurrence patterns with other biterms in the time slice. By preserving all the biterms and modeling the two parts jointly, BBTM thus can better capture the bursty topics. 3) Twevent outperforms other baseline methods that based on topic models (i.e., OLDA, UTM and IBTM). Further examination shows that many topics discovered by these topic model based methods are still about common subjects such as sentiment and life. 4) Moveover, we also find that IBTM outperforms OLDA though they use the same post-processing step, indicating that BTM can better model topics over short texts than LDA.

For qualitative analysis, in Table 2 we show 5 bursty topics (represented by the most probable words) discovered by BBTM along with the corresponding topic probability, i.e., $\hat{\theta}_k$. To help us better understand the results, we also present a news title for each topic obtained by searching its keywords and date in Google. We can see that the 5 bursty topics coincide well with the real-world events reported by the news titles, suggesting the good detection accuracy and potential value of our approach on event detection and summarization.

Coherence of Bursty Topics Discovered

Next, we evaluate the interpretability of the learned bursty topics based on the coherence measure. One popular metric is the PMI-Score (Newman et al. 2010), which calculates the average Pointwise Mutual Information (PMI) between the most probable words in each topic using the large-scale

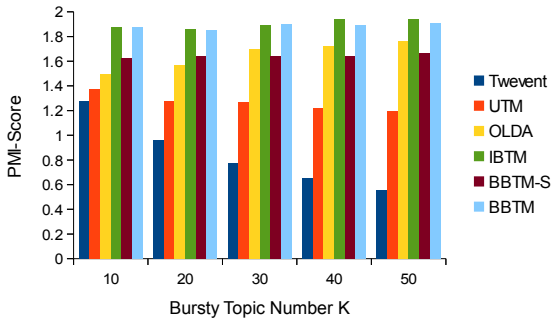

Figure 2: Coherence of the bursty topics discovered (measured by PMI-Score).

Wikipedia data. A larger PMI-Score indicates the topic is more coherent.

The result is shown in Figure 2. We have the following observations. 1) The PMI-Score of BBTM is comparable to IBTM and almost always the highest, indicating good coherence of the learned bursty topics from microblogs. 2) BBTM-S achieves a comparable PMI-Score with OLDA which is substantially higher than UTM and Twevent. However, BBTM-S is lower than BBTM, since it loses a large part of useful word co-occurrence information by filtering the biterms. 3) The PMI-Score of Twevent is always the lowest, indicating that topics obtained by simply clustering the bursty words might be noisy and less coherent. 4) The PMI-Score of UTM is lower than the other topic model based methods. An explanation is that UTM heuristically boosts the probability of bursty words in the temporal topics, which might disturb the topic learning process and degrade the quality of the learned topics.

For qualitative analysis, we chose a hot bursty topic by selecting a hashtag with high bursty probability estimated by Eq.(1). The hashtag is "#ntas" occurred in Jan. 26, 2011, which denotes NTA (National Television Awards), a prominent ceremony in British held on that day. For comparison, we first calculate the empirical word distribution in the tweets containing the hashtag. Then for each method, we select the bursty topic closest to the empirical word distribution of the hashtag under the cosine similarity, and list the results in Table 3.

From Table 3, we can see that: 1) The bursty topics discovered by BBTM is the closest to NTA, and even better explains NTA than the empirical word distribution by ranking some common words such as "morning" and "doctor" in lower position. 2) The bursty discovered BBTM-S is also clearly about NTA, but slightly less readable than BBTM. 3) The bursty topic discovered by Twevent is not well readable, though the words are bursty in the data. UTM seems to have the same problem of Twevent, since it boosts the probability of burst words in each topic in a heuristic way. 4) The topics discovered by OLDA and BTM are relevant to NTA, but they contain many common words such as "year" and "good", suggesting that it is a common issue of the standard topic models.

k	The 10 most probable words	$\hat{\theta}_k$
2	police officers shot shooting detroit twitter adam suspect year revenue (Two St. Petersburg police officers were shot and killed)	0.036
11	airport moscow police news killed people dead blast suicide explosion (Deadly suicide bombing hits Moscow's Domodedovo airport)	0.057
15	open #ausopen nadal australian murray mike tomlin cloud #cloud avril (Australian Open Tennis Championships 2011)	0.015
25	jack lalanne fitness 96 dies guru rip died age dead (Jack LaLanne: US fitness guru who last ate dessert in 1929 dies aged 96)	0.044
26	court emanuel rahm chicago ballot mayor mayoral run appellate rules (Court tosses Emanuel off Chicago mayoral ballot)	0.024

Table 2: Bursty topics discovered by BBTM on Jan. 24, 2011. The sentences in parenthesis are news titles corresponding to these topics obtained by querying the most probable words and its date in Google.

Empirical	Twevent	UTM	OLDA	IBTM	BBTM-S	BBTM
#ntas	#thegame	#thegame	amazing	award	award	#ntas
win	malik	malik	vote	shorty	win	award
love	ant	sitting	award	nominate	bell	awards
award	melanie	standing	movie	oscar	taco	win
matt	eastenders	cut	year	awards	high	#nta
morning	derwin	empty	listen	#ntas	shortly	national
watching	#ntas	pres	awesome	film	speed	love
doctor	tosh	reform	film	win	nominate	tv
lacey	nta	remind	king	love	#ntas	television
cardle	corrie	ai	music	good	rail	nta

Table 3: The bursty topic discovered by each method mostly relates to "#ntas" on Jan.26, 2011. The first column list the most frequent words in the tweets with hashtag "#ntas".

Novelty of Bursty Topics Discovered

In microblogs, we know that the content of bursty topics change continually. We would like to compare the sensitivity of these methods on discovering novel bursty topics by evaluating the novelty of the learned bursty topics across different time slice [3]. Specifically, given a topic set sequence $\{\mathbb{Z}^{(0)}, ..., \mathbb{Z}^{(t)}\}$, we collect the T most probable words of each topic in $\mathbb{Z}^{(t)}$ to construct a *topical word set* $\mathbb{W}^{(t)}$ for each time slice. Then, we define the novelty of $\mathbb{Z}^{(t)}$ as the ratio of novel words in the topical word set, compared to the last time slice. Formally:

$$\text{Novelty}(\mathbb{Z}^{(t)}) = \frac{|\mathbb{W}^{(t)}| - |\mathbb{W}^{(t)} \cap \mathbb{W}^{(t-1)}|}{T * K},$$

where $|\cdot|$ denotes the number of elements of a set. In our experiments, we chose $T = 10$.

In Figures 3, we plot the change of the novelty of the bursty topics as a function of the bursty topic number K. We observe that 1) Both BBTM and BBTM-S significantly outperform OLDA and IBTM, especially when K is large, implying these two bursty oriented methods are more sensitive to bursty topics in microblogs than the temporal topic models. 2) Twevent obtains a very high novelty when K is small, since it summarizes burst topics only with bursty words. However, the novelty decreases fast with the increase of topic number K. Further investigation found that the reason is that many small-sized clusters (e.g., with 2 or 3 words) are discovered by Twevent.

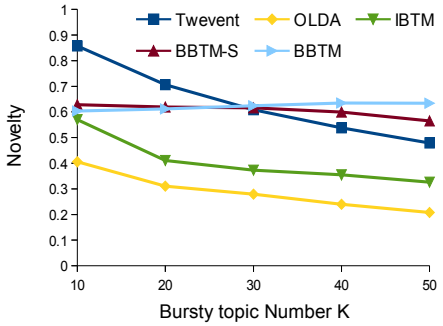

Figure 3: Novelty of the bursty topic discovered.

Efficiency Comparison

Finally, we compare the training time of the methods based on topic models. The experiments are conducted on a personal computer with two Dual-core 2.6GHz Intel processors and 4 GB of RAM, and all the codes are implemented in C++. We summarize the average runtime of per iteration over microblog posts in a single day in Table 4. It is not

[3] Since UTM is a retrospective topic detection model that models topics as static over time, we do not compare it here.

K	UTM	OLDA	IBTM	BBTM-S	BBTM
10	4.24	1.84	4.66	**0.03**	1.57
20	6.02	2.61	5.97	**0.06**	2.89
30	7.84	3.28	7.24	**0.09**	4.40
40	9.79	4.02	8.54	**0.13**	5.71
50	11.63	4.83	9.99	**0.17**	7.24

Table 4: Time cost (second) per iteration.

surprising that BBTM-S costs much less time than other methods, since it only used a subset of biterms for training. We also find that BBTM is much efficient than IBTM and UTM, since it focuses on learning busty topics and spends much less time on learning non-bursty topics. Note that both BBTM and BBTM-S do not require any post-processing steps as OLDA and IBTM, which will cost additional time.

Conclusions & Future Work

We study the problem of bursty topics discovery in microblogs, which is challenging due to the microblog posts are particularly short and noisy. To tackle this problem, we develop a novel bursty biterm topic model (i.e., BBTM) based on the recently introduced short text topic model (i.e., BTM). The key idea is to exploit the burstiness of biterms as the prior knowledge and incorporate it into BTM in a principled way for bursty topic modeling. Our approach can not only well solve the data sparsity problem in topic modeling over short texts, but also automatically learn bursty topics in a efficient way. Experimental results demonstrate the substantial superiority of our approach over the state-of-the-art methods.

For the future work, we would like to further improve the estimation of bursty probability by including more information of the biterms. It would also be interesting to investigate how to model the microblog streams together with other streaming data, like news streams, to better detect and represent bursty topics.

Acknowledgements

This work is funded by the National Basic Research Program of China under Grants No. 2014CB340401, No. 2012CB316303, National High Technology Research and Development Program of China under Grant No. 2012AA011003, No. 2014AA015103, National Natural Science Foundation of China under Grant No. 61232010, 61472401, and National Key Technology R&D Program under Grant No. 2012BAH39B04. We would like to thank the anonymous reviewers for their valuable comments.

References

AlSumait, L.; Barbará, D.; and Domeniconi, C. 2008. Online LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *ICDM*, 3–12. IEEE.

Analytics, P. 2009. Twitter study–august 2009. *San Antonio, TX: Pear Analytics.*

Blei, D.; Ng, A.; and Jordan, M. 2003. Latent Dirichlet allocation. *JMLR* 3:993–1022.

Blei, D. M. 2012. Probabilistic topic models. *Communications of the ACM* 55(4):77–84.

Cataldi, M.; Di Caro, L.; and Schifanella, C. 2010. Emerging topic detection on twitter based on temporal and social terms evaluation. In *MDMKDD*, 4. ACM.

Cheng, X.; Yan, X.; Lan, Y.; and Guo, J. 2014. BTM: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*.

Diao, Q.; Jiang, J.; Zhu, F.; and Lim, E.-P. 2012. Finding bursty topics from microblogs. In *ACL*, 536–544.

Fung, G. P. C.; Yu, J. X.; Yu, P. S.; and Lu, H. 2005. Parameter free bursty events detection in text streams. In *Proceedings of the 31st international conference on Very large data bases*, 181–192. VLDB Endowment.

Griffiths, T. L., and Steyvers, M. 2004. Finding scientific topics. *PNAS* 101(Suppl 1):5228–5235.

Hofmann, T. 1999. Probabilistic latent semantic analysis. *UAI*.

Hong, L., and Davison, B. 2010. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics*, 80–88. ACM.

Lau, J. H.; Collier, N.; and Baldwin, T. 2012. On-line trend analysis with topic models:#twitter trends detection topic model online. In *COLING*, 1519–1534.

Li, C.; Sun, A.; and Datta, A. 2012. Twevent: segment-based event detection from tweets. In *CIKM*, 155–164. ACM.

Lijffijt, J. 2013. A fast and simple method for mining subsequences with surprising event counts. In *ECML PKDD 2013, Prague, Czech Republic, Proceedings, Part I*, 385–400.

Mathioudakis, M., and Koudas, N. 2010. Twittermonitor: Trend detection over the twitter stream. In *SIGMOD*, 1155–1158. New York, NY, USA: ACM.

Mei, Q., and Zhai, C. 2005. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 198–207. ACM.

Newman, D.; Lau, J. H.; Grieser, K.; and Baldwin, T. 2010. Automatic evaluation of topic coherence. In *NAACL HLT*, 100–108.

Swan, R., and Allan, J. 1999. Extracting significant time varying features from text. In *Proceedings of the eighth international conference on Information and knowledge management*, 38–45. ACM.

Tang, J.; Meng, Z.; Nguyen, X.; Mei, Q.; and Zhang, M. 2014. Understanding the limiting factors of topic modeling via posterior contraction analysis. In *ICML*, 190–198.

Yan, X.; Guo, J.; Lan, Y.; and Cheng, X. 2013. A biterm topic model for short texts. In *WWW*, 1445–1456.

Yin, H.; Cui, B.; Lu, H.; Huang, Y.; and Yao, J. 2013. A unified model for stable and temporal topic detection from social media data. In *ICDE*, 661–672. IEEE.