Topic Extraction from Microblog Posts Using Conversation Structures

Jing Li[1,2]*, **Ming Liao**[1,2], **Wei Gao**[3], **Yulan He**[4] and **Kam-Fai Wong**[1,2]
[1]The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[2]MoE Key Laboratory of High Confidence Software Technologies, China
[3]Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, Qatar
[4]School of Engineering and Applied Science, Aston University, UK
{lijing,mliao,kfwong}@se.cuhk.edu.hk[1,2]
wgao@qf.org.qa[3], y.he9@aston.ac.uk[4]

Abstract

Conventional topic models are ineffective for topic extraction from microblog messages since the lack of structure and context among the posts renders poor message-level word co-occurrence patterns. In this work, we organize microblog posts as conversation trees based on reposting and replying relations, which enrich context information to alleviate data sparseness. Our model generates words according to topic dependencies derived from the conversation structures. In specific, we differentiate messages as leader messages, which initiate key aspects of previously focused topics or shift the focus to different topics, and follower messages that do not introduce any new information but simply echo topics from the messages that they repost or reply. Our model captures the different extents that leader and follower messages may contain the key topical words, thus further enhances the quality of the induced topics. The results of thorough experiments demonstrate the effectiveness of our proposed model.

1 Introduction

The increasing popularity of microblog platforms results in a huge volume of user-generated short posts. Automatically modeling topics out of such massive microblog posts can uncover the hidden semantic structures of the underlying collection and can be useful to downstream applications such as microblog summarization (Harabagiu and Hickl, 2011), user profiling (Weng et al., 2010), event tracking (Lin et al., 2010) and so on.

Popular topic models, like Probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003b), model the semantic relationships between words based on their co-occurrences in documents. They have demonstrated their success in conventional documents such as news reports and scientific articles, but perform poorly when directly applied to short and colloquial microblog content due to severe sparsity in microblog messages (Wang and McCallum, 2006; Hong and Davison, 2010).

A common way to deal with short text sparsity is to aggregate short messages into long pseudo-documents. Most of the studies heuristically aggregate messages based on authorship (Zhao et al., 2011; Hong and Davison, 2010), shared words (Weng et al., 2010), or hashtags (Ramage et al., 2010; Mehrotra et al., 2013). Some works directly take into account the word relations to alleviate document-level word sparseness (Yan et al., 2013; Sridhar, 2015). More recently, a self-aggregation-based topic model called SATM (Quan et al., 2015) was proposed to aggregate texts jointly with topic inference.

However, we argue that the existing aggregation strategies are suboptimal for modeling topics in short texts. Microblogs allow users to share and comment on messages with friends through reposting or replying, similar to our everyday conversations. Intuitively, the conversation structures can not only enrich context, but also provide useful clues for identifying relevant topics. This is nonetheless ignored in previous approaches. Moreover, the occurrence of non-topic words such as emotional, sentimental, functional and even meaningless words are very common in microblog posts, which may distract the models from recognizing topic-related key words and thus fail to produce coherent and meaningful topics.

We propose a novel topic model by utilizing the structures of conversations in microblogs. We link microblog posts using reposting and replying rela-

[*] Part of this work was conducted when the first author was visiting Aston University.

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 2114–2123,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

tions to build conversation trees. Particularly, the root of a conversation tree refers to the original post and its edges represent the reposting/replying relations.

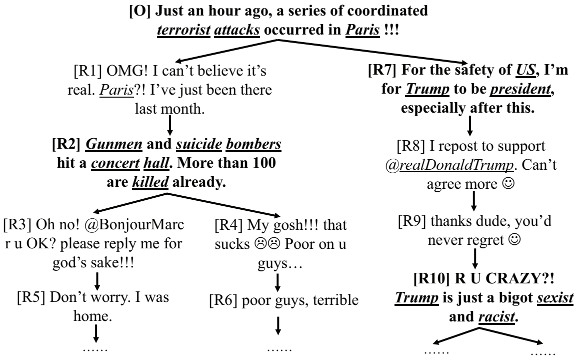
[O] Just an hour ago, a series of coordinated terrorist attacks occurred in Paris !!!
[R1] OMG! I can't believe it's real. Paris?! I've just been there last month.
[R2] Gunmen and suicide bombers hit a concert hall. More than 100 are killed already.
[R3] Oh no! @BonjourMarc r u OK? please reply me for god's sake!!!
[R4] My gosh!!! that sucks ☹☹ Poor on u guys…
[R5] Don't worry. I was home.
[R6] poor guys, terrible
[R7] For the safety of US, I'm for Trump to be president, especially after this.
[R8] I repost to support @realDonaldTrump. Can't agree more ☺
[R9] thanks dude, you'd never regret ☺
[R10] R U CRAZY?! Trump is just a bigot sexist and racist.
……
……
……
……

Figure 1: An example of conversation tree. [O]: the original post; [Ri]: the i-th repost/reply; Arrow lines: reposting/replying relations; Dark black posts: leaders to be detected; Underlined italic words: key words representing topics

Figure 1 illustrates an example of a conversation tree, in which messages can initiate a new topic such as [O] and [R7] or raise a new aspect (subtopic) of the previously discussed topics such as [R2] and [R10]. These messages are named as **leaders**, which contain salient content in topic description, e.g., the italic and underlined words in Figure 1. The remaining messages, named as **followers**, do not raise new issues but simply respond to their reposted or replied messages following what has been raised by the leaders and often contain non-topic words, e.g., *OMG*, *OK*, *agree*, etc.

Conversation tree structures from microblogs have been previously shown helpful to microblog summarization (Li et al., 2015), but have never been explored for topic modeling. We follows Li et al. (2015) to detect leaders and followers across paths of conversation trees using Conditional Random Fields (CRF) trained on annotated data. The detected leader/follower information is then incorporated as prior knowledge into our proposed topic model.

Our experimental results show that our model, which captures parent-child topic correlations in conversation trees and generates topics by considering messages being leaders or followers separately, is able to induce high-quality topics and outperforms a number of competitive baselines. In summary, our contributions are three-fold:

- We propose a novel topic model, which explicitly exploits the topic dependencies contained in conversation structures to enhance topic assignments.
- Our model differentiates the generative process of topical and non-topic words, according to the message where a word is drawn from being a leader or a follower. This helps the model distinguish the topic-specific information from background noise.
- Our model outperforms state-of-the-art topic models when evaluated on a large real-world microblog dataset containing over 60K conversation trees, which is publicly available[1].

2 Related Works

Topic models aim to discover the latent semantic information, i.e., topics, from texts and have been extensively studied. One of the most popular and well-known topic models is LDA (Blei et al., 2003b). It utilizes Dirichlet priors to generate document-topic and topic-word distributions, and has been shown effective in extracting topics from conventional documents.

Nevertheless, prior research has demonstrated that standard topic models, essentially focusing on document-level word co-occurrences, are not suitable for short and informal microblog messages due to severe data sparsity exhibited in short texts (Wang and McCallum, 2006; Hong and Davison, 2010). Therefore, how to enrich and exploit context information becomes a main concern. Weng et al. (2010), Hong et al. (2010) and Zhao et al. (2011) first heuristically aggregated messages posted by the same user or sharing the same words before applying classic topic models to extract topics. However, such a simple strategy poses some problems. For example, it is common that a user has various interests and posts messages covering a wide range of topics. Ramage et al. (2010) and Mehrotra et al. (2013) used hashtags as labels to train supervised topic models. But these models depend on large-scale hashtag-labeled data for model training, and their performance is inevitably compromised when facing unseen topics irrelevant to any hashtag in training data due to the rapid change and wide variety of topics in social media.

SATM (Quan et al., 2015) combined short texts aggregation and topic induction into a unified model. But in their work, no prior knowledge

[1] `http://www1.se.cuhk.edu.hk/~lijing/data/microblog-topic-extraction-data.zip`

was given to ensure the quality of text aggregation, which therefore can affect the performance of topic inference. In this work, we organize microblog messages as conversation trees based on reposting/reply relations, which is a more advantageous message aggregation strategy.

Another line of research tackled the word sparseness by modeling word relations instead of word occurrences in documents. For example, Gaussian Mixture Topic Model (GMTM) (Sridhar, 2015) utilized word embeddings to model the distributional similarities of words and then inferred clusters of words represented by word distributions using Gaussian Mixture Model (GMM) that capture the notion of latent topics. However, GMTM heavily relies on meaningful word embeddings that require a large volume of high-quality external resources for training.

Biterm Topic Model (BTM) (Yan et al., 2013) directly explores unordered word-pair co-occurrence patterns in each individual message. Our model learns topics from aggregated messages based on conversation trees, which naturally provide richer context since word co-occurrence patterns can be captured from multiple relevant messages.

3 LeadLDA Topic Model

In this section, we describe how to extract topics from a microblog collection utilizing conversation tree structures, where the trees are organized based on reposting and replying relations among the messages[2].

To identify key topic-related content from colloquial texts, we differentiate the messages as *leaders* and *followers*. Following Li et al. (2015), we extract all root-to-leaf paths on conversation trees and utilize the state-of-the-art sequence learning model CRF (Lafferty et al., 2001) to detect the leaders[3]. As a result, the posterior probability of each node being a leader or follower is obtained by averaging the different marginal probabilities of the same node over all the tree paths that contain the node. Then, the obtained probability distribution is considered as the observed prior variable input into our model.

3.1 Topics and Conversation Trees

Previous works (Zhao et al., 2011; Yan et al., 2013; Quan et al., 2015) have proven that assuming each short post contains a single topic is useful to alleviate the data sparsity problem. Thus, given a corpus of microblog posts organized as conversation trees and the estimated leader probabilities of tree nodes, we assume that each message only contains a single topic and a tree covers a mixture of multiple topics. Since leader messages subsume the content of their followers, the topic of a leader can be generated from the topic distribution of the entire tree. Consequently, the topic mixture of a conversation tree is determined by the topic assignments to the leader messages on it. The topics of followers, however, exhibit strong and explicit dependencies on the topics of their ancestors. So, their topics need to be generated in consideration of local constraints. Here, we mainly address how to model the topic dependencies of followers.

Enlighten by the general Structural Topic Model (strTM) (Wang et al., 2011), which incorporates document structures into topic model by explicitly modeling topic dependencies between adjacent sentences, we exploit the topical transitions between parents and children in the trees for guiding topic assignments.

Intuitively, the emergence of a leader results in potential topic shift. It tends to weaken the topic similarities between the emerging leaders and their predecessors. For example, [R7] in Figure 1 transfers the topic to a new focus, thus weakens the tie with its parent. We can simplify our case by assuming that followers are topically responsive just up to (hence not further than) their nearest ancestor leaders. Thus, we can dismantle each conversation tree into forest by removing the links between leaders and their parents hence producing a set of subgraphs like [R2]–[R6] and [R7]–[R9] in Figure 1. Then, we model the internal topic dependencies within each subgraph by inferring the parent-child topic transition probabilities satisfying the first-order Markov properties in a similar way as estimating the transition distribution of adjacent sentences in strTM (Wang et al., 2011). At topic assignment stage, the topic of a follower will be assigned by referring to its parent's topic and the transition distribution that captures topic similarities of followers to their parents (see Section 3.2).

In addition, every word in the corpus is either

[2] Reposting/replying relations are straightforward to obtain by using microblog APIs from Twitter and Sina Weibo.

[3] The CRF model for leader detection was trained on a public corpus with all the messages annotated on the tree paths. Details are described in Section 4.

a topical or non-topic (i.e., background) word, which highly depends on whether it occurs in a leader or a follower message.

Figure 2 illustrates the graphical model of our generative process, which is named as LeadLDA.

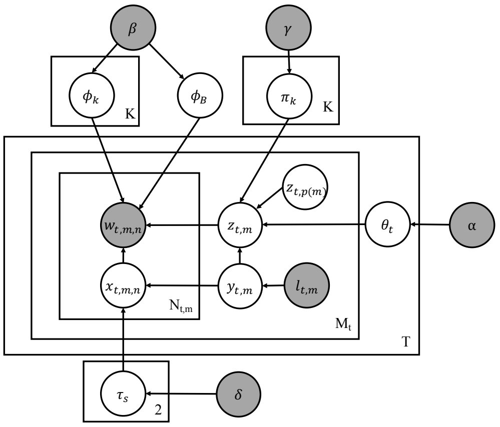

Figure 2: Graphical Model of LeadLDA

3.2 Topic Modeling

Formally, we assume that the microblog posts are organized as T conversation trees. Each tree t contains M_t message nodes and each message m contains $N_{t,m}$ words in the vocabulary. The vocabulary size is V and there are K topics embedded in the corpus represented by word distribution $\phi_k \sim Dir(\beta)$ ($k = 1, 2, ..., K$). Also, a background word distribution $\phi_B \sim Dir(\beta)$ is included to capture the general information, which is not topic specific. ϕ_k and ϕ_B are multinomial distributions over the vocabulary. A tree t is modeled as a mixture of topics $\theta_t \sim Dir(\alpha)$ and any message m on t is assumed to contain a single topic $z_{t,m} \in \{1, 2, ..., K\}$.

(1) Topic assignments: The topic assignments of LeadLDA is inspired by Griffiths et al. (2004) that combines syntactic and semantic dependencies between words. LeadLDA integrates the outcomes of leader detection with a binomial switcher $y_{t,m} \in \{0, 1\}$ indicating whether m is a leader ($y_{t,m} = 1$) or a follower ($y_{t,m} = 0$), given each message m on the tree t. $y_{t,m}$ is parameterized by its leader probability $l_{t,m}$, which is the posterior probability output from the leader detection model and serves as an observed prior variable.

According to the notion of leaders, they initiate key aspects of previously discussed topics or signal a new topic shifting the focus of its descendant followers. So, the topics of leaders on tree t are directly sampled from the topic mixture θ_t.

To model the internal topic correlations within the subgraph of conversation tree consisting of a leader and all its followers, we capture parent-child topic transitions $\pi_k \sim Dir(\gamma)$, which is a distribution over K topics, and use $\pi_{k,j}$ to denote the probability of a follower assigned topic j when the topic of its parent is k. Specifically, if message m is sampled as a follower and the topic assignment to its parent message is $z_{t,p(m)}$, where $p(m)$ indexes the parent of m, then $z_{t,m}$ (i.e., the topic of m) is generated from topic transition distribution $\pi_{z_{t,p(m)}}$. In particular, since the root of a conversation tree has no parent and can only be a leader, we make the leader probability $l_{t,root} = 1$ to force its topic only to be generated from the topic distribution of tree t.

(2) Topical and non-topic words: We separately model the distributions of leader and follower messages emitting topical or non-topic words with τ_0 and τ_1, respectively, both of which are drawn from a symmetric Beta prior parametererized by δ. Specifically, for each word n in message m on tree t, we add a binomial background switcher $x_{t,m,n}$ controlled by whether m is a leader or a follower, i.e., $x_{t,m,n} \sim Bi(\tau_{y_{t,m}})$, which indicates n is a topical word if $x_{t,m,n} = 0$ or a background word if $x_{t,m,n} = 1$, and $x_{t,m,n}$ controls n to be generated from the topic-word distribution $\phi_{z_{t,m}}$, where $z_{t,m}$ is the topic of m, or from background word distribution ϕ_B modeling non-topic information.

(3) Generation process: To sum up, conditioned on the hyper-parameters $\Theta = (\alpha, \beta, \gamma, \delta)$, the generation process of a conversation tree t can be described as follows:

```
• Draw $\theta_t \sim Dir(\alpha)$
• For message $m = 1$ to $M_t$ on tree $t$
    – Draw $y_{t,m} \sim Bi(l_{t,m})$
    – If $y_{t,m} == 1$
        ∗ Draw $z_{t,m} \sim Mult(\theta_t)$
    – If $y_{t,m} == 0$
        ∗ Draw $z_{t,m} \sim Mult(\pi_{z_{t,p(m)}})$
    – For word $n = 1$ to $N_{t,m}$ in $m$
        ∗ Draw $x_{t,m,n} \sim Bi(\tau_{y_{t,m}})$
        ∗ If $x_{t,m,n} == 0$
            · Draw $w_{t,m,n} \sim Mult(\phi_{z_{t,m}})$
        ∗ If $x_{t,m,n} == 1$
            · Draw $w_{t,m,n} \sim Mult(\phi_B)$
```

$C^{LB}_{s,(r)}$	# of words with background switchers assigned as r and occurring in messages with leader switchers s.
$C^{LB}_{s,(\cdot)}$	# of words occurring in messages whose leader switchers are s, i.e., $\sum_{r\in\{0,1\}} C^{LB}_{s,(r)}$.
$N^{B}_{(r)}$	# of words occurring in message (t,m) and with background switchers assigned as r.
$N^{B}_{(\cdot)}$	# of words in message (t,m), i.e., $N^{B}_{(\cdot)} = \sum_{r\in\{0,1\}} N^{B}_{(r)}$.
$C^{TW}_{k,(v)}$	# of words indexing v in vocabulary, sampled as topic (non-background) words, and occurring in messages assigned topic k.
$C^{TW}_{k,(\cdot)}$	# of words assigned as topic (non-background) word and occurring in messages assigned topics k, i.e., $C^{TW}_{k,(\cdot)} = \sum_{v=1}^{V} C^{TW}_{k,(v)}$.
$N^{W}_{(v)}$	# of words indexing v in vocabulary that occur in message (t,m) and are assigned as topic (non-background) word.
$N^{W}_{(\cdot)}$	# of words assigned as topic (non-background) words and occurring in message (t,m), i.e., $N^{W}_{(\cdot)} = \sum_{v=1}^{V} N^{W}_{(v)}$.
$C^{TR}_{i,(j)}$	# of messages sampled as followers and assigned topic j, whose parents are assigned topic i.
$C^{TR}_{i,(\cdot)}$	# of messages sampled as followers whose parents are assigned topic i, i.e., $C^{TR}_{i,(\cdot)} = \sum_{j=1}^{K} C^{TR}_{i,(j)}$.
$I(\cdot)$	An indicator function, whose value is 1 when its argument inside () is true, and 0 otherwise.
$N^{CT}_{(j)}$	# of messages that are children of message (t,m), sampled as followers and assigned topic j.
$N^{CT}_{(\cdot)}$	# of message (t,m)'s children sampled as followers, i.e., $N^{CT}_{(\cdot)} = \sum_{j=1}^{K} N^{CT}_{(j)}$
$C^{TT}_{t,(k)}$	# of messages on conversation tree t sampled as leaders and assigned topic k.
$C^{TT}_{t,(\cdot)}$	# of messages on conversation tree t sampled as leaders, i.e., $C^{TT}_{t,(\cdot)} = \sum_{k=1}^{K} C^{TT}_{t,(k)}$
$C^{BW}_{(v)}$	# of words indexing v in vocabulary and assigned as background (non-topic) words
$C^{BW}_{(\cdot)}$	# of words assigned as background (non-topic) words, i.e., $C^{BW}_{(\cdot)} = \sum_{v=1}^{V} C^{BW}_{(v)}$

Table 1: The notations of symbols in the sampling formulas (1) and (2). (t,m): message m on conversation tree t.

3.3 Inference for Parameters

We use collapsed Gibbs Sampling (Griffiths, 2002) to carry out posterior inference for parameter learning. The hidden multinomial variables, i.e., message-level variables (y and z) and word-level variables (x) are sampled in turn, conditioned on a complete assignment of all other hidden variables. Due to the space limitation, we leave out the details of derivation but give the core formulas in the sampling steps.

We first define the notations of all variables needed by the formulation of Gibbs sampling, which are described in Table 1. In particular, the various C variables refer to counts excluding the message m on conversation tree t.

For each message m on a tree t, we sample the leader switcher $y_{t,m}$ and topic assignment $z_{t,m}$ according to the following conditional probability distribution:

$$\begin{aligned}
&p(y_{t,m}=s, z_{t,m}=k \mid \mathbf{y}_{\neg(t,m)}, \mathbf{z}_{\neg(t,m)}, \mathbf{w}, \mathbf{x}, \mathbf{l}, \Theta) \\
\propto&\frac{\Gamma(C^{LB}_{s,(\cdot)}+2\delta)}{\Gamma(C^{LB}_{s,(\cdot)}+N^{B}_{(\cdot)}+2\delta)} \prod_{r\in\{0,1\}} \frac{\Gamma(C^{LB}_{s,(r)}+N^{B}_{(r)}+\delta)}{\Gamma(C^{LB}_{s,(r)}+\delta)} \\
&\cdot\frac{\Gamma(C^{TW}_{k,(\cdot)}+V\beta)}{\Gamma(C^{TW}_{k,(\cdot)}+N^{W}_{(\cdot)}+V\beta)} \prod_{v=1}^{V} \frac{\Gamma(C^{TW}_{k,(v)}+N^{W}_{(v)}+\beta)}{\Gamma(C^{TW}_{k,(v)}+\beta)} \\
&\cdot g(s,k,t,m)
\end{aligned} \tag{1}$$

where $g(s,k,t,m)$ takes different forms depending on the value of s:

$$\begin{aligned}
g(0,k,t,m) =& \frac{\Gamma(C^{TR}_{z_{t,p(m)},(\cdot)}+K\gamma)}{\Gamma(C^{TR}_{z_{t,p(m)},(\cdot)}+I(z_{t,p(m)}\neq k)+K\gamma)} \\
&\cdot\frac{\Gamma(C^{TR}_{k,(\cdot)}+K\gamma)}{\Gamma(C^{TR}_{k,(\cdot)}+I(z_{t,p(m)}=k)+N^{CT}_{(\cdot)}+K\gamma)} \\
&\cdot\prod_{j=1}^{K} \frac{\Gamma(C^{TR}_{k,(j)}+N^{CT}_{(j)}+I(z_{t,p(m)}=j=k)+\gamma)}{\Gamma(C^{TR}_{k,(j)}+\gamma)} \\
&\cdot\frac{\Gamma(C^{TR}_{z_{t,p(m)},(k)}+I(z_{t,p(m)}\neq k)+\gamma)}{\Gamma(C^{TR}_{z_{t,p(m)},(k)}+\gamma)}\cdot(1-l_{t,m})
\end{aligned}$$

and

$$g(1,k,t,m) = \frac{C^{TT}_{t,(k)}+\alpha}{C^{TT}_{t,(\cdot)}+K\alpha}\cdot l_{t,m}$$

For each word n in m on t, the sampling formula of its background switcher is given as the following:

$$\begin{aligned}
&p(x_{t,m,n}=r \mid \mathbf{x}_{\neg(t,m,n)}, \mathbf{y}, \mathbf{z}, \mathbf{w}, \mathbf{l}, \Theta) \\
\propto&\frac{C^{LB}_{y_{t,m},(r)}+\delta}{C^{LB}_{y_{t,m},(\cdot)}+2\delta}\cdot h(r,t,m,n)
\end{aligned} \tag{2}$$

where

$$h(r,t,m,n) = \begin{cases} \frac{C^{TW}_{z_{t,m},(w_{t,m,n})}+\beta}{C^{TW}_{z_{t,m},(\cdot)}+V\beta} & \text{if } r=0 \\ \frac{C^{BW}_{(w_{t,m,n})}+\beta}{C^{BW}_{(\cdot)}+V\beta} & \text{if } r=1 \end{cases}$$

4 Data Collection and Experiment Setup

To evaluate our LeadLDA model, we conducted experiments on real-world microblog dataset collected from Sina Weibo that has the same 140-character limitation and shares the similar market penetration as Twitter (Rapoza, 2011). For the hyper-parameters of LeadLDA, we fixed $\alpha = 50/K$, $\beta = 0.1$, following the common practice in previous works (Griffiths and Steyvers, 2004; Quan et al., 2015). Since there is no analogue of γ and δ in prior works, where γ controls topic dependencies of follower messages to their ancestors and δ controls the different tendencies of

Month	# of trees	# of messages	Vocab size
May	10,812	38,926	6,011
June	29,547	98,001	9,539
July	26,103	102,670	10,121

Table 2: Statistics of our three evaluation datasets

leaders and followers covering topical and non-topic words. We tuned γ and δ by grid search on a large development set containing around 120K posts and obtained $\gamma = 50/K$, $\delta = 0.5$.

Because the content of posts are often incomplete and informal, it is difficult to manually annotate topics in a large scale. Therefore, we follow Yan et al. (2013) to utilize hashtags led by '#', which are manual topic labels provided by users, as ground-truth categories of microblog messages. We collected the real-time trending hashtags on Sina Weibo and utilized the hashtag-search API[4] to crawl the posts matching the given hashtag queries. In the end, we built a corpus containing 596,318 posts during May 1 – July 31, 2014.

To examine the performance of models on various topic distributions, we split the corpus into 3 datasets, each containing messages of one month. Similar to Yan et al. (2013), for each dataset, we manually selected 50 frequent hashtags as topics, e.g. #mh17, #worldcup, etc. The experiments were conducted on the subsets of posts with the selected hashtags. Table 2 shows the statistics of the three subsets used in our experiments.

We preprocessed the datasets before topic extraction in the following steps: 1) Use FudanNLP toolkit (Qiu et al., 2013) for word segmentation, stop words removal and POS tagging for Chinese Weibo messages; 2) Generate a vocabulary for each dataset and remove words occurring less than 5 times; 3) Remove all hashtags in texts before input them to models, since the models are expected to extract topics without knowing the hashtags, which are ground-truth topics; 4) For LeadLDA, we use the CRF-based leader detection model (Li et al., 2015) to classify messages as leaders and followers. The leader detection model was implemented by using CRF++[5], which was trained on the public dataset composed of 1,300 conversation paths and achieved state-of-the-art 73.7% F1-score of classification accuracy (Li et al., 2015).

5 Experimental Results

We evaluated topic models with two sets of K, i.e., the number of topics. One is $K = 50$, to match the count of hashtags following Yan et al. (2013), and the other is $K = 100$, much larger than the "real" number of topics. We compared LeadLDA with the following 5 state-of-the-art basedlines.

TreeLDA: Analogous to Zhao et al. (2011), where they aggregated messages posted by the same author, TreeLDA aggregates messages from one conversation tree as a pseudo-document. Additionally, it includes a background word distribution to capture non-topic words controlled by a general Beta prior without differentiating leaders and followers. TreeLDA can be considered as a degeneration of LeadLDA, where topics assigned to all messages are generated from the topic distributions of the conversation trees they are on.

StructLDA: It is another variant of LeadLDA, where topics assigned to all messages are generated based on topic transitions from their parents. The strTM (Wang et al., 2011) utilized a similar model to capture the topic dependencies of adjacent sentences in a document. Following strTM, we add a dummy topic $\mathrm{T}_{\mathrm{start}}$ emitting no word to the "pseudo parents" of root messages. Also, we add the same background word distribution to capture non-topic words as TreeLDA does.

BTM: Biterm Topic Model (BTM)[6] (Yan et al., 2013) directly models topics of all word pairs (biterms) in each post, which outperformed LDA, Mixture of Unigrams model, and the model proposed by Zhao et al. (2011) that aggregated posts by authorship to enrich context.

SATM: A general unified model proposed by Quan et al. (2015) that aggregates documents and infers topics simultaneously. We implemented SATM and examined its effectiveness specifically on microblog data.

GMTM: To tackle word sparseness, Sridhar et al. (2015) utilized Gaussian Mixture Model (GMM) to cluster word embeddings generated by a log-linear word2vec model[7].

The hyper-parameters of BTM, SATM and GMTM were set according to the best hyper-parameters reported in their original papers. For TreeLDA and StructLDA, the parameter settings were kept the same as LeadLDA since they are its

[4] http://open.weibo.com/wiki/2/search/topics

[5] https://taku910.github.io/crfpp/

[6] https://github.com/xiaohuiyan/BTM

[7] https://code.google.com/archive/p/word2vec/

variants. And the background switchers were parameterized by symmetric Beta prior on 0.5, following Chemudugunta et al. (2006). We ran Gibbs samplings (in LeadLDA, TreeLDA, StructLDA, BTM and SATM) and EM algorithm (in GMTM) with 1,000 iterations to ensure convergence.

Topic model evaluation is inherently difficult. In previous works, perplexity is a popular metric to evaluate the predictive abilities of topic models given held-out dataset with unseen words (Blei et al., 2003b). However, Chang et al. (2009) have demonstrated that models with high perplexity do not necessarily generate semantically coherent topics in human perception. Therefore, we conducted objective and subjective analysis on the coherence of produced topics.

5.1 Objective Analysis

The quality of topics is commonly measured by coherence scores (Mimno et al., 2011), assuming that words representing a coherent topic are likely to co-occur within the same document. However, due to the severe sparsity of short text posts, we modify the calculation of commonly-used topic coherence measure based on word co-occurrences in messages tagged with the same hashtag, named as hashtag-document, assuming that those messages discuss related topics[8].

Specifically, we calculate the coherence score of a topic given the top N words ranked by likelihood as below:

$$\mathcal{C} = \frac{1}{K} \cdot \sum_{k=1}^{K} \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{D(w_i^k, w_j^k) + 1}{D(w_j^k)}, \tag{3}$$

where w_i^k represents the i-th word in topic k ranked by $p(w|k)$, $D(w_i^k, w_j^k)$ refers to the count of hashtag-documents where word w_i^k and w_j^k co-occur, and $D(w_i^k)$ denotes the number of hashtag-documents that contain word w_i^k.

Table 3 shows the absolute values of $\mathcal{C}$ scores for topics produced on three evaluation datasets (May, June and July), and the top 10, 15, 20 words of topics were selected for evaluation. Lower scores indicate better coherence in the induced topic.

We have the following observations:

- GMTM gave the worst coherence scores, which may be ascribed to its heavy reliance on relevant large-scale high-quality external data, without which the trained word embedding model failed to capture meaningful semantic features for words, and hence could not yield coherent topics.

N	Model	May		June		July	
		K50	K100	K50	K100	K50	K100
10	TREE	27.9	30.5	24.0	23.8	23.9	26.1
	STR	29.9	30.8	24.0	24.1	24.4	26.4
	BTM	**26.7**	28.9	27.8	25.5	25.4	25.2
	SATM	30.6	29.9	23.8	23.7	24.3	27.5
	GMTM	40.8	40.1	44.0	44.2	41.7	40.8
	LEAD	28.4	**26.9**	**19.8**	**23.4**	**22.6**	**25.1**
15	TREE	71.9	76.4	55.3	60.4	61.2	66.2
	STR	76.4	74.1	57.6	62.2	58.1	61.1
	BTM	69.6	71.4	58.5	60.3	59.1	63.0
	SATM	74.3	73.0	54.8	60.4	61.2	65.3
	GMTM	96.4	93.1	100.4	105.1	94.6	94.9
	LEAD	**67.4**	**65.2**	**52.8**	**57.7**	**55.3**	**57.8**
20	TREE	138.8	138.6	102.0	115.0	115.8	119.7
	STR	134.0	136.9	104.3	112.7	111.0	117.3
	BTM	125.2	131.1	109.4	115.7	115.3	120.2
	SATM	134.6	131.9	105.5	114.3	113.5	118.9
	GMTM	173.5	169.0	184.7	190.9	167.4	171.2
	LEAD	**120.9**	**127.2**	**101.6**	**106.0**	**97.2**	**104.9**

Table 3: Absolute values of coherence scores. Lower is better. K50: 50 topics; K100: 100 topics; N: # of top words ranked by topic-word probabilities; TREE: TreeLDA; STR: StructLDA; LEAD: LeadLDA.

- TreeLDA and StructLDA produced competitive results compared to the state-of-the-art baseline models, which indicates the effectiveness of using conversation structures to enrich context and thus generate topics of reasonably good quality.

- The coherence of topics generated by LeadLDA outperformed all the baselines on the three datasets, most of time by large margins and was only outperformed by BTM on the May dataset when $K = 50$ and $N = 10$. The generally higher performance of LeadLDA is due to three reasons: 1) It effectively identifies topics using the conversation tree structures, which provide richer context information; 2) It jointly models the topics of leaders and the topic dependencies of other messages on a tree. TreeLDA and StructLDA, each only considering one of the factors, performed worse than LeadLDA; 3) LeadLDA separately models the probabilities of leaders and followers containing topical or non-topic words while the baselines only model the general background information regardless of the different types of messages. This implies that leaders and followers do have different capacities in covering key topical words or background noise, which is useful to identify key words for topic representation.

[8] We sampled posts and their corresponding hashtags in our evaluation set and found only 1% mismatch.

TreeLDA	StructLDA	BTM	SATM	LeadLDA
香港 微博 马航 家属 证实 入境处 客机 消息 曹格 投给 二胎 选项 教父 滋养 飞机 外国 心情 坠毁 男子 同胞	乌克兰 航空 亲爱 国民 绕开 飞行 航班 领空 所有 避开 宣布 空域 东部 俄罗斯 终于 忘记 公司 绝望 看看 珍贵	香港 入境处 家属 证实 男子 护照 外国 消息 坠毁 马航 报道 联系 电台 客机 飞机 同胞 确认 事件 霍家 直接	马航 祈祷 安息 生命 逝者 世界 艾滋病 恐怖 广州 飞机 无辜 默哀 远离 事件 击落 公交车 中国人 国际 愿逝者 真的	乌克兰 马航 客机 击落 飞机 坠毁 导弹 俄罗斯 消息 乘客 中国 马来西亚 香港 遇难 事件 武装 航班 恐怖 目前 证实
Hong Kong, microblog, family, confirm, immigration, airliner, news, Grey Chow, vote, second baby, choice, god father, nourish, airplane, foreign, feeling, crash, man, fellowman	Ukraine, airline, dear, national, bypass, fly, flight, airspace, all, avoid, announce, airspace, eastern, Russia, finally, forget, company, disappointed, look, valuable	Hong Kong, immigration, family, confirm, man, passport, foreign, news, crash, Malaysia Airlines, report, contact, broadcast station, airliner, airplane, fellowman, confirm, event, Fok's family, directly	Malaysia Airlines, prey, rest in peace, life, dead, world, AIDS, terror, Guangzhou, airplane, innocent, silent tribute, keep away from, event, shoot down, bus, Chinese, international, wish the dead, really	Ukraine, Malaysia Airlines, airliner, shoot down, airplane, crash, missile, Russia, news, passenger, China, Malaysia, Hong Kong, killed, event, militant, flight, terror, current, confirm

Figure 3: The extracted topics describing MH17 crash. Each column represents the similar topic generated by the corresponding model with the top 20 words. The 2nd row: original Chinese words; The 3rd row: English translations.

5.2 Subjective Analysis

To evaluate the coherence of induced topics from human perspective, we invited two annotators to subjectively rate the quality of every topic (by displaying the top 20 words) generated by different models on a 1-5 Likert scale. A higher rating indicates better quality of topics. The Fless's Kappa of annotators' ratings measured for various models on different datasets given $K = 50$ and 100 range from 0.62 to 0.70, indicating substantial agreements (Landis and Koch, 1977).

Table 4 shows the overall subjective ratings. We noticed that humans preferred topics produced given $K = 100$ to $K = 50$, but coherence scores gave generally better grades to models for $K = 50$, which matched the number of topics in ground truth. This is because models more or less mixed more common words when K is larger. Coherence score calculation (Equation (3)) penalizes common words that occur in many documents, whereas humans could somehow "guess" the meaning of topics based on the rest of words thus gave relatively good ratings. Nevertheless, annotators gave remarkably higher ratings to LeadLDA than baselines on all datasets regardless of K being 50 or 100, which confirmed that LeadLDA effectively yielded good-quality topics.

For a detailed analysis, Figure 3 lists the top 20 words about "MH17 crash" induced by different models[9] when $K = 50$. We have the following observations:

Model	May		June		July	
	K50	K100	K50	K100	K50	K100
TREE	3.12	3.41	3.42	3.44	3.03	3.48
STR	3.05	3.45	3.38	3.48	3.08	3.53
BTM	3.04	3.26	3.40	3.37	3.15	3.57
SATM	3.08	3.43	3.30	3.55	3.09	3.54
GMTM	2.02	2.37	1.99	2.27	1.97	1.90
LEAD	**3.40**	**3.57**	**3.52**	**3.63**	**3.55**	**3.72**

Table 4: Subjective ratings of topics. The meanings of K50, K100, TREE, STR and LEAD are the same as in Table 3.

- BTM, based on word-pair co-occurrences, mistakenly grouped "Fok's family" (a tycoon family in Hong Kong), which co-occurred frequently with "Hong Kong" in other topics, into the topic of "MH17 crash". "Hong Kong" is relevant here as a Hong Kong passenger died in the MH17 crash.
- The topical words generated by SATM were mixed with words relevant to the bus explosion in Guangzhou, since it aggregated messages according to topic affinities based on the topics learned in the previous step. Thus the posts about bus explosion and MH17 crash, both pertaining to disasters, were aggregated together mistakenly, which generated spurious topic results.
- Both TreeLDA and StructLDA generated topics containing non-topic words like "microblog" and "dear". This means that without distinguishing leaders and followers, it is difficult to filter out non-topic words. The topic quality of StructLDA nevertheless seems better than

[9] As shown in Table 3 and 4, the topic coherence scores of GMTM were the worst. Hence, the topic generated by GMTM is not shown due to space limitation.

TreeLDA, which implies the usefulness of exploiting topic dependencies of posts in conversation structures.

- LeadLDA not only produced more semantically coherent words describing the topic, but also revealed some important details, e.g., MH17 was shot down by a missile.

6 Conclusion and Future Works

This paper has proposed a novel topic model by considering the conversation tree structures of microblog posts. By rigorously comparing our proposed model with a number of competitive baselines on real-world microblog datasets, we have demonstrated the effectiveness of using conversation structures to help model topics embedded in short and colloquial microblog messages.

This work has proven that detecting leaders and followers, which are coarse-grained discourse derived from conversation structures, is useful to model microblogging topics. In the next step, we plan to exploit fine-grained discourse structures, e.g., dialogue acts (Ritter et al., 2010), and propose a unified model that jointly inferring discourse roles and topics of posts in context of conversation tree structures. Another extension is to extract topic hierarchies by integrating the conversation structures into hierarchical topic models like HLDA (Blei et al., 2003a) to extract fine-grained topics from microblog posts.

Acknowledgment

This work is supported by General Research Fund of Hong Kong (417112), the Innovation and Technology Fund of Hong Kong SAR (ITP/004/16LP), Shenzhen Peacock Plan Research Grant (KQCX20140521144507925) and Innovate UK (101779). We would like to thank Shichao Dong for his efforts on data processing and anonymous reviewers for the useful comments.

References

David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. 2003a. Hierarchical topic models and the nested chinese restaurant process. In *Proceedings of the 17th Annual Conference on Neural Information Processing Systems, NIPS*, pages 17–24.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003b. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Jonathan Chang, Jordan L. Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems, NIPS*, pages 288–296.

Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. 2006. Modeling general and specific aspects of documents with a probabilistic topic model. In *Proceedings of the 20th Annual Conference on Neural Information Processing Systems, NIPS*, pages 241–248.

Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235.

Thomas L. Griffiths, Mark Steyvers, David M. Blei, and Joshua B. Tenenbaum. 2004. Integrating topics and syntax. In *Proceedings of the 18th Annual Conference on Neural Information Processing Systems, NIPS*, pages 537–544.

Tom Griffiths. 2002. Gibbs sampling in the generative model of latent dirichlet allocation.

Sanda M. Harabagiu and Andrew Hickl. 2011. Relevance modeling for microblog summarization. In *Proceedings of the 5th International Conference on Web and Social Media, ICWSM*.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *In Proceedings of the 22nd Annual International, ACM SIGIR*, pages 50–57.

Liangjie Hong and Brian D Davison. 2010. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning, ICML*, pages 282–289.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Jing Li, Wei Gao, Zhongyu Wei, Baolin Peng, and Kam-Fai Wong. 2015. Using content-level structures for summarizing microblog repost trees. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 2168–2178.

Cindy Xide Lin, Bo Zhao, Qiaozhu Mei, and Jiawei Han. 2010. PET: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th International Conference on Knowledge Discovery and Data Mining, ACM SIGKDD*, pages 929–938.

Rishabh Mehrotra, Scott Sanner, Wray L. Buntine, and Lexing Xie. 2013. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th International conference on research and development in Information Retrieval, ACM SIGIR*, pages 889–892.

David M. Mimno, Hanna M. Wallach, Edmund M. Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 262–272.

Xipeng Qiu, Qi Zhang, and Xuanjing Huang. 2013. Fudannlp: A toolkit for chinese natural language processing. In *51st Annual Meeting of the Association for Computational Linguistics, ACL*, pages 49–54.

Xiaojun Quan, Chunyu Kit, Yong Ge, and Sinno Jialin Pan. 2015. Short and sparse text topic modeling via self-aggregation. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence, IJCAI*, pages 2270–2276.

Daniel Ramage, Susan T. Dumais, and Daniel J. Liebling. 2010. Characterizing microblogs with topic models. In *Proceedings of the 4th International Conference on Web and Social Media, ICWSM*.

Kenneth Rapoza. 2011. China's weibos vs us's twitter: And the winner is? *Forbes (May 17, 2011)*.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations. In *Proceedings of the 2010 Conference of the North American Chapter of the Association of Computational Linguistics, NAACL*, pages 172–180.

Vivek Kumar Rangarajan Sridhar. 2015. Unsupervised entity linking with abstract meaning representation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 1130–1139.

Xuerui Wang and Andrew McCallum. 2006. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining, ACM SIGKDD*, pages 424–433.

Hongning Wang, Duo Zhang, and ChengXiang Zhai. 2011. Structural topic model for latent topical structure analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 1526–1535.

Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the 3rd International Conference on Web Search and Web Data Mining, WSDM*, pages 261–270.

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd International World Wide Web Conference, WWW*, pages 1445–1456.

Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval - 33rd European Conference on IR Research, ECIR*, pages 338–349.