Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Don't Forget the Quantifiable Relationship between Words: Using Recurrent Neural Network for Short Text Topic Discovery

Heng-Yang Lu, Lu-Yao Xie, Ning Kang, Chong-Jun Wang and Jun-Yuan Xie
National Key Laboratory for Novel Software Technology at Nanjing University
Nanjing University, Nanjing 210023, China
{hylu,xly}@smail.nju.edu.cn, nkangnju@gmail.com, {chjwang, jyxie}@nju.edu.cn

Abstract

In our daily life, short texts have been everywhere especially since the emergence of social network. There are countless short texts in online media like twitter, online Q&A sites and so on. Discovering topics is quite valuable in various application domains such as content recommendation and text characterization. Traditional topic models like LDA are widely applied for sorts of tasks, but when it comes to short text scenario, these models may get stuck due to the lack of words. Recently, a popular model named BTM uses word co-occurrence relationship to solve the sparsity problem and is proved effectively. However, both BTM and extended models ignore the inside relationship between words. From our perspectives, more related words should appear in the same topic. Based on this idea, we propose a model named RIBS-TM which makes use of RNN for relationship learning and IDF for filtering high-frequency words.Experiments on two real-world short text datasets show great utility of our model.

Introduction

Here comes a digital era. We are surrounded with large quantities of data like texts, pictures and so on. There are countless data emerging everyday which contain valuable information to be mined. Particularly in recent years, the Internet has totally changed our life. For example, more and more people express their opinions through social network, and journalists are used to post their news on the Internet. We can hardly analyse these massive data directly, that's why we need a tool like topic model to help us organize and summarize digital data automatically. By detecting topic information from these data, we could use these results for some interesting applications such as sentiment analysis (Lin and He 2009), question retrieval in Q&A sites (Ji et al. 2012) and personalized recommendation (Jiang et al. 2015).

Lots of work has been done in the research field of topic model. Early studies like probabilistic latent semantic analysis (PLSA) (Hofmann 1999) and latent dirichlet allocation (LDA) (Blei, Ng, and Jordan 2003) are two classic topic models widely used for discovering hidden topics from text corpus. They are both based on the assumption that each document is a mixture of topics and each topic is a probability distribution over words. These two topic models are designed for regular text and are really effective to documents with many words. However, posting short text data like tweets or online questions on the Internet is becoming popular, we have to deal with short text more often. Different from regular text data, the sparsity of short text content brings challenge to traditional topic models because words are too few to learn and analyze from original corpus.

One intuitive solution for the sparsity problem is extending the original short texts into longer ones by aggregating similar texts. For example, Weng et al. (Weng et al. 2010) aggregated texts which were posted by the same author before using LDA. There are also some methods utilizing additional knowledge from the Internet, like Jin et al. (Jin et al. 2011) extended short texts by bringing in related texts from online search results. The shortcoming is obvious because these methods need extra data for discovering topics. For example, if we get a data set without author information or we can find little suitable knowledge from the Internet, the effectiveness of this kind of methods will be greatly reduced.

Another creative idea alleviates the problem by constructing word pairs or word groups to represent the original texts. One representative work is biterm topic model (BTM) which uses word co-occurrence relationship from original corpus to learn topics (Yan et al. 2013). While word network topic model (WNTM) constructs pseudo documents with word groups learned from the word network (Zuo, Zhao, and Xu 2014). These models indeed have a superior performance than traditional methods. However, it's worth noting that they both lack quantifiable relationship between words. For instance, given a document with words (iPhone, iPad, house), BTM models biterms (iPhone, iPad) and (iPhone, house) equally for learning topics. But according to our knowledge, *iPad* may have a higher probability appearing in the same topic with *iPhone* than *house*. This means we can not ignore the prior knowledge of relationship between words. What's more, we can find another defect caused by high-frequency words. Although most work will delete some stop words before modeling, we may still have some high-frequency words which are worthless for topics. Modeling word pairs or groups is highly likely to bring these topic-irrelevant words into final results.

In this paper, we focus on topic model in short text scenario and aim to solve the existing problems mentioned above. There are various evaluation metrics for relation-

ship between words. For example, Chen et al. (Chen and Kao 2015) used pointwise mutual information (PMI) to describe this relationship. Unfortunately, PMI is simply based on statistics. For example, if (A, B) co-occurs as many times as (A, C) does, PMI will fail to distinguish the influence caused by different distances between (A, B) and (A, C). So we prefer to learn this relationship by training recurrent neural networks (RNN) not only relying on its learning skills but also on its intelligent memory. At the same time, to filter high-frequency words, we apply classic inverse document frequency (IDF) (Sparck Jones 1972) for each word. We call this model as RNN-IDF based Biterm Short-text Topic Model (RIBS-TM), the main contributions include:

- Bringing quantifiable relationship between words learned from whole corpus to describe biterms better.
- Using IDF of words to help filter high-frequency common words in a probabilistic way.
- Discovering RNN's positive effects on RIBS-TM which means we can optimize this topic model by optimizing RNN's performance.

This paper is organized as follows. Section 2 shows related researches. Section 3 presents our topic model named RIBS-TM. Section 4 contains the experiments and finally Section 5 concludes.

Related Work

As we know, topic model has developed for years, especially researches on regular text. In this section, we focus on recent work in short text scenario and give a brief summarization.

With the explosive growth of short text data and high value of applications like text categorization (Wang et al. 2014) and news clustering (Xia et al. 2015), short text topic model has become a promising research field, more and more researchers have shown interest in it. The main challenge brought by short text lies in the lack of words, which may cause the word-document matrix seriously sparse. This kind of phenomenon is harmful for topic discovery because we can hardly describe topics without enough words. Most models are proposed based on the following ideas. One idea in early years is document aggregation. For example, Hong et al. aggregated tweets which share the same key words before using LDA (Hong and Davison 2010), Jin et al. extended short texts with auxiliary related texts (Jin et al. 2011). These models need extra text data which may be limited or hard to get. Some other researchers think they can propose assumptions for modeling. For example, Zhao et al. assumed that each document would only contain one topic (Zhao et al. 2011), similar to this idea, Lin et al. assumed that each document would contain the most related subset of topics and each topic could be composed by limited words (Lin et al. 2014). This kind of ideas need to impose several limits on the model, as we consider, which might not be the best choice. Another novel idea in recent years is constructing word groups or word pairs. Using word groups to construct pseudo document is feasible because semantic related word groups can stand for the same topic, work like WNTM (Zuo, Zhao, and Xu 2014) is based on this idea. Using word pairs is also popular, Yan et al. proposed a novel topic model named BTM (Yan et al. 2013) which could learn topics by modeling the generation of word co-occurrence patterns directly. Further work like d-BTM (Xia et al. 2015) extended BTM by deleting some redundant biterms, as shown in Figure 1.

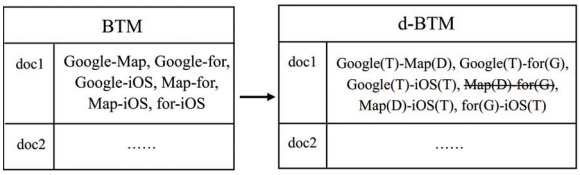


Figure 1: A simple illustration for biterm extraction of BTM and d-BTM

For doc1 "Google Map for IOS", we can see BTM extracts every co-occurrence word pair in a document as a biterm, while d-BTM tries to exclude some unimportant biterms. It labels each word as a topic term (T), general term (G) and document specific term (D) respectively, biterms without topic terms will be deleted. For example, *Map* is a document specific term and *for* is a general term, so biterm *Map-for* will be deleted.

From our perspectives, BTM and extended models are more suitable and universal for short text scenario. That's why we do research on this basis. However, few models have taken quantifiable relationships between words into consideration. For BTM, this model ignores different relationship between words and biterms while d-BTM tries to filter some useless biterms simply by deleting them. We believe it is more rational to describe the relationship between words and filter redundant biterms by bringing in prior knowledge.

RIBS-TM

In this section, we'd like to describe how to discover topics with RIBS-TM. First, we will give the problem setting of short text topic model. After that, we will give a detailed introduction to RIBS-TM.

Problem Setting

Given the corpus D with N_D documents whose vocabulary size is W, topic model aims to discover topics of each document and learn topic representation with words. If the corpus has K topics, topic model should give an $N_D \times K$ matrix for topic distribution over document and a $K \times W$ matrix for word distribution over topic by learning observed words.

Model Description

RIBS-TM utilizes prior knowledge to measure the relationship between words. If two words are more related, they are more likely to belong to the same topic. Different from BTM's generative process, we assume that two words in a biterm are drawn from a topic probabilistically based on their relationship, where a topic is still sampled from a topic mixture over the whole corpus. The generative process is described as follows, shown in Figure 2 as well.

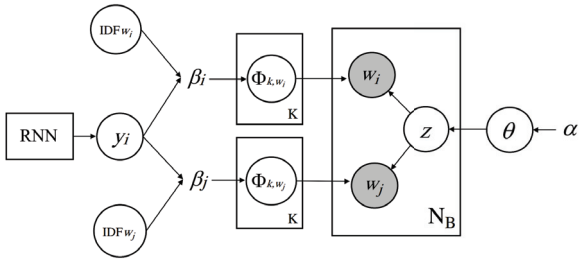


Figure 2: Graphical representation of RIBS-TM

1. Learn prior knowledge β from corpus D.
2. Draw $\theta \sim$ Dirichlet (α).
3. For each topic $k \in [1, K]$
   (a) draw $\phi_{k,w_i} \sim$ Dirichlet (β_i).
   (b) draw $\phi_{k,w_j} \sim$ Dirichlet (β_j).
4. For each biterm $b \in \mathbf{B}$, where $b = (w_i, w_j)$
   (a) draw $z \sim$ Multinomial (θ).
   (b) draw $w_i \sim$ Multinomial $(\phi_{z_{w_i}})$.
   draw $w_j \sim$ Multinomial $(\phi_{z_{w_j}})$.

where $\mathbf{B}$ is a biterm set which contains all the biterms and $N_B = |\mathbf{B}|$, z is a variable which represents topic id, θ is a K-dimensional multinomial distribution where θ_k represents the probability of topic z_k (we denote the topic as z_k when $z_k = k$, $k \in [1, K]$), while ϕ is a $K \times W$ matrix which is the word distribution over topics, we denote the k-th row in ϕ as ϕ_k to represent the word distribution over topic z_k. w_i and w_j with shadow background are two observed words. α and β are the symmetric Dirichlet priors for θ and ϕ. In RIBS-TM, we bring in prior knowledge for β.

Prior Knowledge Learning

As we have mentioned, most short text topic models like BTM ignore the quantifiable relationship between words. However, this kind of relationship is very important because if two words are more related, they may have a higher probability to appear in the same topic. We think the prior knowledge should satisfy the following properties:

- If two words are more likely to appear in the same generative sentence, they are more related.
- If two words are far away from each other in the same sentence, the relationship between them shall be weakened.

Fortunately, artificial neural networks have been found effective in learning relationship between words for sentence generation (Sutskever, Martens, and Hinton 2011). We find RNN is a good choice to satisfy these properties for the following reasons:

- Output of RNN can quantify word w_j's generation probability when given word w_i and previously observed words. This probability may reflect the similarity and tightness between two words.
- The learning process of RNN can guarantee that the earlier a word is learned, the less influence it will have on current learning word.

Encouraged by recent work which utilizes RNN for short text representation (Amiri and Daumé III 2016), we use a simple recurrent neural network called Elman (Elman 1990) to learn relationship between words. The network is shown in Figure 3.

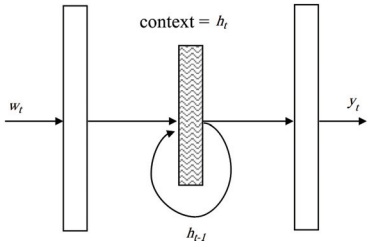


Figure 3: A simple Elman recurrent neural network

$w_t \in \mathbb{R}^L$ represents current word where L is the length of vectorized w_t, $h_t \in \mathbb{R}^H$ is a hidden layer where H is the size of the hidden layer, $y_t \in \mathbb{R}^W$ is the output layer. t is current input time.

Since the hidden layer h_{t-1} and h_t have a recurrent connection, we can believe that h_{t-1} has remembered all the words observed before time t. This means RNN can learn the relationship between the current word and previously observed words. Additionally, the influence by the previously observed words is decreasing over time.

The input layer $x_t \in \mathbb{R}^{L+H}$ is defined as $x_t = [w_t, h_{t-1}]$, we can compute hidden and output layers with x_t:

$$h_t = \phi(\mathbf{U}x_t). \tag{1}$$

$$y_t = g(\mathbf{V}h_t). \tag{2}$$

where ϕ is the sigmoid function and g is the softmax function:

$$\phi(z) = \frac{1}{1 + e^{-z}}. \tag{3}$$

$$g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}}. \tag{4}$$

$\mathbf{U} \in \mathbb{R}^{H \times (L+H)}$ and $\mathbf{V} \in \mathbb{R}^{W \times H}$ are two weight matrices for us to learn.

Once learned the output, we can define the relationship between w_i and w_j as $y_i(j)$, which is the j-th value in y_i.

$$y_i(j) = P(w_j | w_i, h_{i-1}). \tag{5}$$

From Equation (5) we can see $y_i(j)$ represents given w_i, the probability of w_j appears, the given h_{i-1} guarantees previously observed words also have effects on w_j by distance.

What's more, to filter high-frequency words, work like d-BTM just deletes some topic-irrelevant biterms. Since we're short of words, deleting biterms may cause further information loss. So we decide to utilize IDF for measuring each word as follows:

$$\text{IDF}_{w_i} = log \frac{|N_D|}{|d \in D : w_i \in d|}. \tag{6}$$

where $|d \in D : w_i \in d|$ represents number of documents word w_i appears in. The more times w_i appears in documents, the smaller value of IDF_{w_i} will be. We can use this weight to decrease w_i's probability of generating a topic.

Now we can give the definition of prior knowledge β, for word w_i and w_j:

$$\beta_i = \epsilon \times y_i(j) \times \text{IDF}_{w_i}. \tag{7}$$

$$\beta_j = \epsilon \times y_i(j) \times \text{IDF}_{w_j}. \tag{8}$$

where ϵ is to avoid β being too small.

Biterm Construction

RIBS-TM constructs biterms by using any two distinct words in a document, which means we can generate C_n^2 biterms from an n-word document. Different from BTM's biterm extraction, we need to bring in prior knowledge. For each biterm $b \in \mathbf{B}$, the new definition is as follows:

$b = (w_i, w_j, r_{ij})$, where $r_{ij} = \langle \text{IDF}_{w_i}, \text{IDF}_{w_j}, y_i(j) \rangle$.

When scanning the whole corpus, biterm-construction process is executing at the same time.

Gibbs sampling for Parameter Estimation

We employ Gibbs sampling for learning parameters like BTM by taking prior knowledge into consideration. According to the chain rule on the joint probability of the corpus, we acquire the following conditional probability equation:

$$p(z|z_{-b}, \mathbf{B}) \propto \frac{(n_{-b,z} + \alpha)}{N_B + K\alpha} \frac{(n_{-b,w_i|z} + \beta_i)(n_{-b,w_j|z} + \beta_j)}{(\sum_w (n_{-b,w|z} + \beta))^2}. \tag{9}$$

where $n_{-b,z}$ is the number of biterms assigned to topic z without biterm b, $n_{-b,w_i|z}$ is the number of word w_i assigned to topic z without biterm b. Then we can estimate global topic parameter θ and topic-word distribution parameter ϕ as follows:

$$\theta_k = \frac{(n_{z_k} + \alpha)}{N_B + K\alpha}. \tag{10}$$

for word w_i and w_j

$$\phi_{k,w_i} = \frac{n_{w_i|z_k} + \beta_i}{\sum_w (n_{w|z_k} + \beta)}. \tag{11}$$

$$\phi_{k,w_j} = \frac{n_{w_j|z_k} + \beta_j}{\sum_w (n_{w|z_k} + \beta)}. \tag{12}$$

The Gibbs sampling procedure is shown in Algorithm 1. According to the definition of Eq. (11)(12), we can denote $\phi_k = [\phi_{k,w_1}, \phi_{k,w_2}, ..., \phi_{k,w_W}]$ as word distribution over topic z_k.

Topics Inference

Because RIBS-TM models topics on biterms, we have to infer the topic distribution over document by utilizing knowledge learned by Gibbs sampling. Deriving topic z_k's proportion of a document $d \in D$ is as follows:

$$P(z_k|d) = \sum_{b \in \mathbf{B}} P(z_k, b|d) = \sum_{b \in \mathbf{B}} P(z_k|b, d) P(b|d). \tag{13}$$

Algorithm 1 Gibbs sampling algorithm for RIBS-TM

Input: topic number K, α, β, biterm set $\mathbf{B}$.
Output: θ and ϕ.

```
Initialize topic assignments for each biterm randomly.
for iter ← 1 to N_iter do
    for each biterm b = (w_i, w_j, r_ij) ∈ B do
        Draw topic z_k from P(z|z_{-b}, B).
        Update n_{z_k}, n_{w_i|z_k}, n_{w_j|z_k}.
    end for
end for
Compute θ by Eq. (10) and φ by Eq. (11)(12).
```

we assume the topic of b denoted as z_k is conditionally independent of d, which means $P(z_k|b, d) = P(z_k|b)$, so we can get the following simplified equation:

$$P(z_k|d) = \sum_{b \in \mathbf{B}} P(z_k|b) P(b|d). \tag{14}$$

We can calculate $P(z_k|b)$ via Bayes formula:

$$P(z_k|b) = \frac{P(z_k)P(w_i|z_k)P(w_j|z_k)}{\sum_{k' \in K} P(z_{k'})P(w_i|z_{k'})P(w_j|z_{k'})}. \tag{15}$$

where $P(z_k) = \theta_k$, $P(w_i|z_k) = \phi_{k,w_i}$, θ and ϕ are parameters learned in RIBS-TM.

As to calculate $P(b|d)$, we can simply treat as a count problem:

$$P(b|d) = \frac{n_d(b)}{\sum_{b \in \mathbf{B}} n_d(b)}. \tag{16}$$

where $n_d(b)$ is the frequency of biterm b in document d. So the topic distribution over document d is $P(z|d) = [P(z_1|d), P(z_2|d), ..., P(z_K|d)]$.

Outputs of RIBS-TM are the $N_D \times K$ matrix for topic distribution over document and the $K \times W$ matrix for word distribution over topic, calculated as follows:

$$P(z|D) = [P(z|d_1), P(z|d_2), ..., P(z|d_{N_D})] \tag{17}$$

$$\phi = [\phi_{z_1}, \phi_{z_2}, ..., \phi_{z_K}] \tag{18}$$

Experiments

In this section, we conduct several experiments to prove RIBS-TM outperforms state-of-the-art topic models in short text scenario. We will give experimental results and analysis compared with three baseline models.

Data Sets

To prove the effectiveness of RIBS-TM, we choose two real-world short text datasets for topic discovery:

- Online Questions: the corpus contains 13865 questions from a famous Chinese online Q&A community named ZhiHu. Each question is attached with a label and the whole corpus is classified into 22 categories. The average length of single question is 6.14 words which definitely belongs to short text.

- Online News: the open source corpus contains famous Chinese news sites published and labeled by SogouLab. We sampled 24427 news titles from it randomly. Each title is attached with a label and the whole dataset is classified into 13 categories. The average length of each title is 6.13 words.

Both datasets are preprocessed by deleting stop words and documents with less than 4 words.

Baseline Models

We compared RIBS-TM with three topic models:

- LDA is a famous topic model which performs really well in regular text scenario. We use a standard open source LDA implemented by Gibbs sampling.
- BTM is a recently proposed topic model for short text. We do experiments with the standard code provided by BTM authors.
- d-BTM is extended from BTM by deleting some topic-irrelevant biterms. We implement this model based on BTM source code.

As to parameters, we set $\alpha = 50/K$, $\beta = 0.05$. For learning prior knowledge, we set $\epsilon = 50$ for Online Questions dataset and $\epsilon = 1$ for Online News Dataset. This assignment is determined by experimental attempts.

Experiments and Analysis

Better topic discovery ability of RIBS-TM Topic model is designed for topic discovery, so topic quality is a significant judgement of model performance. This experiment aims to prove RIBS-TM has a better performance in topic discovery than baselines. We choose coherence (Mimno et al. 2011) as the evaluation metric. The main idea of coherence is that a good topic should consist of words in cohesive semantic similarity. It is calculated as follows:

$$C = \frac{1}{K}\sum_{z=1}^{K}\sum_{m=2}^{M}\sum_{l=1}^{m-1} log\frac{n_D(w_m^z, w_l^z) + \epsilon'}{n_D(w_l^z)}. \qquad (19)$$

where $[w_1^z, w_2^z, ..., w_M^z]$ denotes the M most representative words of topic z. $n_D(w_l)$ is the word frequency of w_l and $n_D(w_m, w_l)$ is the co-occurrence count in the corpus. As we can see, C is a negative number, a higher value indicates a better performance. We conduct this experiment with $K = 10, 20, 30$ and calculate coherence by choosing M from 5 to 20. The final result is via experimenting ten times. For online questions, we list result with $K = 30$ in Table 1. For online news titles, we list result with $K = 20$ in Table 2.

Table 1: Coherence of Questions

	M=5	M=10	M=20
LDA	-38.4 $\pm$ 0.8	-216.7 $\pm$ 1.6	-1097.0 $\pm$ 4.2
BTM	-19.4 $\pm$ 0.6	-116.0 $\pm$ 1.8	-644.3 $\pm$ 1.6
d-BTM	-20.5 $\pm$ 0.2	-119.9 $\pm$ 1.7	-663.4 $\pm$ 0.7
RIBS-TM	**-17.6 $\pm$ 0.5**	**-105.3 $\pm$ 0.9**	**-601.7 $\pm$ 2.0**

Table 2: Coherence of News Titles

	M=5	M=10	M=20
LDA	-30.9 $\pm$ 0.6	-186.8 $\pm$ 2.4	-995.7 $\pm$ 5.2
BTM	-18.0 $\pm$ 0.4	-115.0 $\pm$ 1.6	-639.8 $\pm$ 3.9
d-BTM	-18.2 $\pm$ 0.2	-115.7 $\pm$ 2.3	-650.8 $\pm$ 1.8
RIBS-TM	**-16.4 $\pm$ 0.2**	**-106.3 $\pm$ 0.9**	**-602.4 $\pm$ 3.9**

As we can see, all three short text topic models outperform LDA on both two datasets, which means LDA is really unsuitable for short texts for lacking enough words. Results show that biterm construction is good for short text topic discovery. No matter what value M is, coherence of RIBS-TM is always more close to 0, the improvement over both BTM and d-BTM lies in quantifiable relationship brought by RIBS-TM. This kind of prior knowledge is learned from the whole corpus and remembers observed words over time, which can help two semantic related words occur in the same topic. Take the learned topic about *Internet* for example, listed in Table 3 (we have translated the source Chinese words into English).

Table 3: The 10 most probable words from Questions, italic words are topic-relevant words judged by human

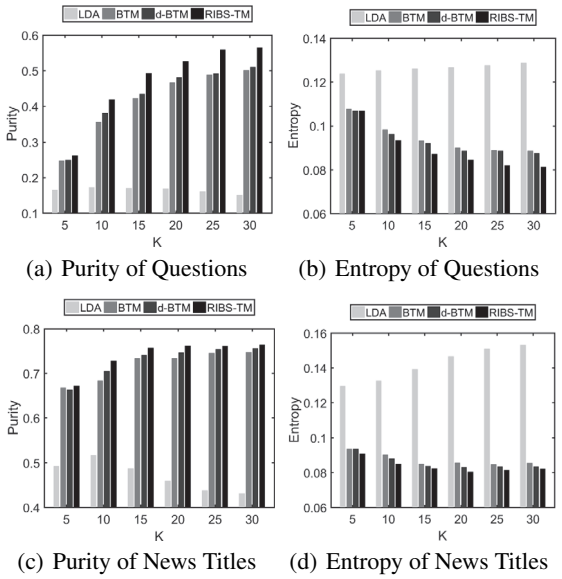
BTM	company, *invest*, Internet, product, Google, *Manager*, *Fund*, google, Baidu, *regard*
RIBS-TM	Google, company, google, Baidu, Internet, Apple, product, Microsoft, *review*, *domestic*

RIBS-TM discovers less irrelevant words than BTM because it utilizes quantifiable relationship between words. For example, google and Microsoft ($y_{google}(Microsoft) = 0.3587$) is quite related while google and Fund ($y_{google}(Fund) = 0.0002$) is almost irrelevant, so google and Microsoft are assigned to topic *Internet* by RIBS-TM instead of google and Fund by BTM. The coherence performance and kind of topic examples show that RIBS-TM has a better topic discovery ability.

Advantage of document characterization Document characterization is a common application of topic model, so we conduct clustering and classification experiments to prove the advantage of RIBS-TM from another perspective.

Clustering aims to gather unlabeled documents into several clusters, each of which contains semantic similarly documents. This is an effective method to measure topic quality. For fair comparison, we use the same clustering method as BTM does. We take each topic as a cluster, and assign each document d to the topic cluster z with highest value of conditional probability $P(z|d)$. Purity and entropy are two common evaluation metrics for clustering, result shows in Figure 4.

Purity computes the ratio of dominant category in each cluster which a larger value means a better performance. Entropy is used for measuring chaos in a set of data so that a smaller entropy indicates a better performance. In



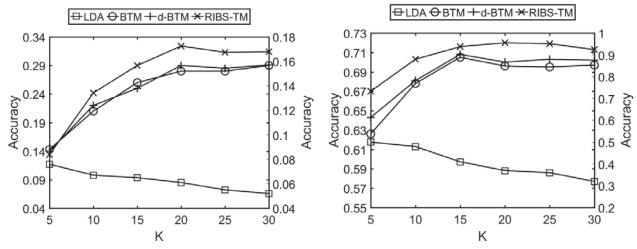
(a) Purity of Questions (b) Entropy of Questions

(c) Purity of News Titles (d) Entropy of News Titles

Figure 4: Clustering performance

this experiment, we set K from 5 to 30 with step size as 5. We can see RIBS-TM has the best performance on both datasets. Although d-BTM indeed improves BTM in clustering, the improvement is not as good as RIBS-TM. We think deleting some biterms may reduce several topic-irrelevant ones, but will also lose some word-topic information at the same time. Different from d-BTM, RIBS-TM utilizes probabilistic knowledge learned from IDF for reducing high-frequency words which can remain word-topic information as much as possible. We think it's beneficial to achieve higher topic quality.

Classification aims to give each document a label by learning from label-observable documents. This is a direct way to measure semantically document representation by topics. We use topic distribution over documents as features and choose standard CART decision tree for classification. Results measured by accuracy are shown in Figure 5.



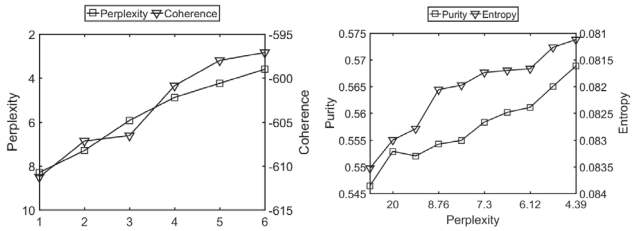
(a) For Online Questions (b) For Online News Titles

Figure 5: Classification performance using Decision Tree

Figure 5(a) shows the accuracy of online questions, we can see that when K is around 20, RIBS-TM has the best performance, the same as K around 10 in Figure 5(b). This may be related to the actual categories of two datasets, which are 22 categories and 13 categories respectively.

From above experiments, we can conclude RIBS-TM has a better performance than other baselines in document characterization.

Utility of RNN RNN plays an important role in RIBS-TM. This experiment aims to prove RNN is indeed suitable for learning quantifiable relationship between words by exploring how RNN's performance affects RIBS-TM. We choose perplexity to evaluate RNN which a smaller value indicates a better performance. Experimental results on online questions are shown in Figure 6.



(a) RNN with Topic Quality (b) RNN with Clustering

Figure 6: Exploring how RNN affects RIBS-TM

Figure 6(a) shows the trend of perplexity and coherence, we can see with perplexity of RNN getting better, coherence is getting better as well. Figure 6(b) shows clustering performance with the change of RNN. In general, the changing trend of both purity and entropy is consistent with perplexity. This result not only proves the effectiveness of RNN but also inspires us that maybe we can optimize our topic model by optimizing RNN which is quite encouraging.

Conclusion and Future Work

Topic model is widely accepted as an effective tool for organizing and summarizing digital data. With the explosive growth of social network on the Internet, topic model for short text has become a promising research field. Analysing data like online Q&A suffers from the sparsity problem. In this paper, we propose a novel short text topic model named RIBS-TM which brings prior knowledge learned from RNN and IDF into biterms. To the best of our knowledge, few short text models have taken quantifiable relationship between words into consideration. Firstly, RIBS-TM learns semantic tightness between words by training a recurrent neural network which can remember previously observed words and effectively reflect similarity between words. Secondly, RIBS-TM learns IDF for each word so that the model can filter high-frequency words by reducing its probability occurring in topics. Experimental results show that this kind of prior knowledge is quite important and useful for short text topic discovery. Additionally, RIBS-TM is encouraging because we find RNN's performance has a positive effect on model which means we can optimizing topic learning model by optimizing a neural network.

As for future work, since most short text data are emerging continuously, we'd like to extend RIBS-TM into an online model and apply it to more real world applications.

Acknowledgments

This paper is supported by the National Key Research and Development Program of China (Grant No. 2016YFB1001102) and the National Natural Science Foundation of China(Grant No.61375069, 61403156, 61502227), this research is supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University.

References

Amiri, H., and Daumé III, H. 2016. Short text representation for detecting churn in microblogs. In *Proceedings of the 30th AAAI conference on Artificial Intelligence*, 2566–2572.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Chen, G., and Kao, H. 2015. Word co-occurrence augmented topic model in short text. *Computational Linguistics & Chinese Language Processing* 45.

Elman, J. L. 1990. Finding structure in time. *Cognitive science* 14(2):179–211.

Hofmann, T. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 50–57. ACM.

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

Ji, Z.; Xu, F.; Wang, B.; and He, B. 2012. Question-answer topic model for question retrieval in community question answering. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2471–2474. ACM.

Jiang, S.; Qian, X.; Shen, J.; Fu, Y.; and Mei, T. 2015. Author topic model-based collaborative filtering for personalized poi recommendations. *IEEE Transactions on Multimedia* 17(6):907–918.

Jin, O.; Liu, N. N.; Zhao, K.; Yu, Y.; and Yang, Q. 2011. Transferring topical knowledge from auxiliary long texts for short text clustering. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, 775–784. ACM.

Lin, C., and He, Y. 2009. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management*, 375–384. ACM.

Lin, T.; Tian, W.; Mei, Q.; and Cheng, H. 2014. The dual-sparse topic model: mining focused topics and focused terms in short text. In *Proceedings of the 23rd international conference on World wide web*, 539–550. ACM.

Mimno, D.; Wallach, H. M.; Talley, E.; Leenders, M.; and McCallum, A. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 262–272. Association for Computational Linguistics.

Sparck Jones, K. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28(1):11–21.

Sutskever, I.; Martens, J.; and Hinton, G. E. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning*, 1017–1024.

Wang, P.; Zhang, H.; Wu, Y.; Xu, B.; and Hao, H. 2014. A robust framework for short text categorization based on topic model and integrated classifier. In *2014 International Joint Conference on Neural Networks*, 3534–3539. IEEE.

Weng, J.; Lim, E.-P.; Jiang, J.; and He, Q. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, 261–270. ACM.

Xia, Y.; Tang, N.; Hussain, A.; and Cambria, E. 2015. Discriminative bi-term topic model for headline-based social news clustering. In *FLAIRS Conference*, 311–316.

Yan, X.; Guo, J.; Lan, Y.; and Cheng, X. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web*, 1445–1456. ACM.

Zhao, W. X.; Jiang, J.; Weng, J.; He, J.; Lim, E.; Yan, H.; and Li, X. 2011. Comparing twitter and traditional media using topic models. In *European Conference on Information Retrieval*, 338–349. Springer.

Zuo, Y.; Zhao, J.; and Xu, K. 2014. Word network topic model: a simple but general solution for short and imbalanced texts. *Knowledge and Information Systems* 1–20.