IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 16, 2018, accepted July 19, 2018, date of publication August 6, 2018, date of current version August 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2863260

GLTM: A Global and Local Word Embedding-Based Topic Model for Short Texts

WENXIN LIANG[1], (Member, IEEE), RAN FENG[2], XINYUE LIU[2], YUANGANG LI[3], AND XIANCHAO ZHANG[2]

[1]School of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
[2]School of Software, Dalian University of Technology, Dalian 116620, China
[3]School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai 200433, China

Corresponding author: Xianchao Zhang (xczhang@dlut.edu.cn)

This work was supported by the National Science Foundation of China under Grant 61632019.

ABSTRACT Short texts have become a kind of prevalent source of information, and discovering topical information from short text collections is valuable for many applications. Due to the length limitation, conventional topic models based on document-level word co-occurrence information often fail to distill semantically coherent topics from short text collections. On the other hand, word embeddings as a powerful tool have been successfully applied in natural language processing. Word embeddings trained on large corpus are encoded with general semantic and syntactic information of words, and hence they can be leveraged to guide topic modeling for short text collections as supplementary information for sparse co-occurrence patterns. However, word embeddings are trained on large external corpus and the encoded information is not necessarily suitable for training data set of topic models, which is ignored by most existing models. In this article, we propose a novel global and local word embedding-based topic model (GLTM) for short texts. In the GLTM, we train global word embeddings from large external corpus and employ the continuous skip-gram model with negative sampling (SGNS) to obtain local word embeddings. Utilizing both the global and local word embeddings, the GLTM can distill semantic relatedness information between words which can be further leveraged by Gibbs sampler in the inference process to strengthen semantic coherence of topics. Compared with five state-of-the-art short text topic models on four real-world short text collections, the proposed GLTM exhibits the superiority in most cases.

INDEX TERMS Text mining, context modeling, natural language processing, topic model, short text.

I. INTRODUCTION

With the spur of social networks and mobile devices, short texts such as tweets have become an important source of information in people's daily life. Since there is abundant information in short texts, mining semantic information from short text collections is valuable for many downstream applications such as collaborative filtering, community discovery and so on.

Traditional probabilistic topic models such as Latent Dirichlet Allocation (LDA) [1] and Probabilistic Latent Semantic Index (PLSI) [2] have been successfully employed in various text corpora. Based on the document-level word co-occurrence patterns, traditional topic models are able to distill semantically coherent topics from long text collections.

In the context of topic model, a topic is a multinomial distribution over words; a document is regarded as a multinomial distribution over topics; and words are generated according to topics. Meanwhile, topics range among the whole corpus. Therefore, each topic can be regarded as a cluster center of words which are clustered by co-occurrence patterns. Words occurring in same documents are preferably assigned in a same topic. However, due to the limitation of document length, word co-occurrence patterns are sparse in short text collections, which results in performance degradation of conventional topic models. In other words, semantic relation of words can not be effectively modeled in short text scenarios by conventional topic models.

There have been several ingenious strategies proposed to alleviate the sparsity of word co-occurrence patterns for short

text topic modeling. The most straightforward strategy is aggregating short text snippets into long pseudo-documents according to some auxiliary information. For example, in the context of Tweet collections, some metadata such as authorship, hashtag, timestamp and location can be leveraged to aggregate tweets before applying conventional topic models. Experimental results in literature [3] also demonstrate that the hashtag of tweet is the most effective indicator for aggregation strategies of Tweet collections. However, the aggregation strategy heavily depends on training dataset and has limited scalability. Instead of utilizing auxiliary metadata, Self-Aggregation based Topic Model (SATM) [4] aggregates short text documents according to their topic distribution in the inference process. Based on SATM, Pseudo-documents based Topic Model (PTM) [5] unifies the generation process of topics and long pseudo document indexes for short documents, which can reduce the time complexity of inference process. However, there are several important hyper-parameters in these models need to be tuned carefully, which restricts their usability.

Another strategy is to impose strong assumption for generation process of short text documents without auxiliary information. For example, Biterm Topic Model (BTM) [6] directly models the word co-occurrence patterns from short text collections, which can effectively capture semantic relationship between words. However, the unacceptable time complexity of inference process in this model impedes its broad application. Dirichlet Multinomial Mixture (DMM) model [7] assumes that each short document contains only one topic which effectively distills topics from short texts with the cost of scalability. On the other hand, word embeddings [8], [9] as a powerful tool have been successfully applied in many tasks of natural language processing. Word embeddings are trained on large corpus hence encoded with general semantic and syntactic information of words, which can be compensation for the sparsity of word co-occurrence patterns. Based on the DMM model, Li et.al [10] propose the General Pólya Urn Dirichlet Multinomial Mixture (GPU-DMM) model which utilizes the General Pólya Urn (GPU) model to incorporate the general word-pair semantic relatedness knowledge obtained from word embeddings to enhance topic coherence. Xun et.al [11] incorporate word embeddings into generation process of the DMM model, and each topic can be regarded as a cluster center of word embeddings. However, the quality of word embeddings heavily depends on their training datasets and training models. Thus, the encoded semantic information of word embeddings is not necessarily suitable for training dataset of topic models, which is ignored by most existing models.

In this article, we propose a novel Global and Local word embedding-based Topic Model (GLTM) for short texts to tackle the sparsity issue and obtain semantically coherent topics. Firstly, we train global word embeddings from large external corpus and employ the continuous skip-gram model with negative sampling (SGNS) [8], [9] to obtain local word embeddings from training dataset. The global word embeddings are encoded with general semantic and syntactic information of words, whereas the local word embeddings contain words' context information of training dataset. For each word in vocabulary of training dataset, we can obtain its semantically related word set according to the global and local word embeddings. Secondly, in the generation process of short text collections, we leverage spike-and-slab priors to capture the sparse structure of document-topic distributions. Thirdly, in the topic inference process, the GPU model [12] is employed as Gibbs sampler, which changes statistics of semantically related words simultaneously. Finally, we can obtain coherent topics by using maximum posterior estimation (MAP). Experimental results on four real-world datasets against five state-of-art short text topic models demonstrate the effectiveness of GLTM qualitatively and quantitatively. The main contributions of this article are summarized as follows.

- We propose a novel Global and Local word embedding-based Topic Model (GLTM) for short texts. In GLTM, a new generation process is proposed to capture the sparse structure of topic distributions for short text documents to achieve better interpretability.
- In GLTM, we obtain semantical relationship between words by both global and local word embeddings which can be further leveraged by Gibbs sampler to strengthen topic coherence.
- We conduct extensive experiments on four real-world short text datasets. The experimental results indicate that the proposed GLTM gains a clear edge compared with five state-of-the-art short text topic models in both topic coherence test and short text classifications.

II. RELATED WORK

Conventional probabilistic topic models such as LDA and PLSA distill topical information from text collections based on document-level word co-occurrence patterns. Short documents are characterized with short document length, hence two words hardly co-occur in same documents even they are semantically related, which impedes conventional topic models distilling coherent topics from short text collections. Some heuristic strategies have been proposed, for example, [3], [13], [14] aggregates tweets according to some auxiliary information before applying LDA model. DualLDA model [15] learns topics from short text corpus with auxiliary long texts. Chen et.al [16]–[18] transfer learned topical information from prior domain to target domain which can be regarded as a kind of transfer learning for topic modeling.

There are some efforts towards proposing topic models which cater for characteristics of short texts to intensify word co-occurrence patterns directly without auxiliary information. Biterm topic model [6] is the first short text topic model which can be directly applied in general short text corpora. It takes the assumption that words in a biterm is generated by same topic, which explicitly increases co-occurrence information of words. DMM model [7] rigidly assumes that

each document contains only one topic, which may be not suitable for long documents but kind of rational for short texts. SATM [4], PTM, and SparsePTM [5] assume that each short is just a snippet of a long pseudo document, and topics are associated with long pseudo documents which alleviates the sparsity problem. Shi et.al. [19] propose SeaNMF model which leverages the word-context semantic correlation in training process to improve semantic coherence of topics.

Word embeddings trained on large corpora are inherently encoded with general semantic and syntactic information of words which can be regarded as prior knowledge. Sridhar et.al in literature [20] leverage Gaussian Mixture Model to cluster word embeddings to obtain topics. GuassianLDA [21] changes the generation process of LDA by generating word embeddings instead textural words, and each topic is a multivariate Gaussian distribution over word embeddings. Based on GaussianLDA, Xun et.al [11] incorporate word embeddings into the generation process of DMM model, and also introduce background topic to distill coherent topics from short text collections. Generating word embeddings in generation process of topic models can escape the curse of data sparsity but incur unacceptable time complexity. Based on DMM model, GPU-DMM model [10] leverages word embeddings to obtain semantic relatedness information between words which can be further utilized by GPU model in inference process to improve semantic coherence of topics. Qiang et.al [22] employ word embeddings as prior knowledge to cluster short documents before applying Markov-Random-Field-LDA [23].

The most similar study to our work is the GPU-DMM model. However, the GPU-DMM model assumes that information encoded in word embeddings always fits for training dataset of topic model. And the GPU-DMM model also adopts the assumption of DMM model that each short document contains only one topic, which is too strict under some circumstances. In this article, we prohibit the above insufficiency to develop a new topic model for short texts.

III. THE PROPOSED GLTM

In this section, we first describe the generation process and graphic model representation of the GLTM. Then, we will show how to incorporate the semantic relatedness information into inference process. Finally, we exhibit the Gibbs sampling scheme for the GLTM.

A. THE GENERATION PROCESS

In this article, we take the assumption that each short document contains more than one topics. Due to the limitation of document length, each short document focuses on a minority topics instead all topics in corpus. Thus, we employ the spike-and-slab prior [24] to capture the sparse topical structure for each short document. The spike-and-slab prior is a well established method in mathematics, which has been widely applied in many applications because it can effectively decouple the sparsity and smoothness of probability distribution [15]. Specifically, it employs auxiliary Bernoulli variables for random variables to determine whether the corresponding variables appear or not. In the context of topic modeling, a Bernoulli variable indicates a topic is selected by a document or a word is selected by a topic. In this article, we adopt the strategy of Dual-Sparsity Topic Model [15], which defines a weak smoothing prior and a smoothing prior for the spike-and-slab structure to bypass the issue of ill-definition distributions. Moreover, it also simplifies the inference process. We do not apply the spike-and-slab structure for word distribution of topics, because according to the literature [25], sparsity structure for "topic-word" distributions is not necessary. The required notations in this article are shown in Table 1, and the detail of generation process of GLTM can be referred as follows.

TABLE 1. Definitions of notations.

Notation	Meaning
$\boldsymbol{V}$	vocabulary of short documents
$\boldsymbol{D}$	collection of short documents
N_d	length of the document d
K	number of topics
$\boldsymbol{\psi}_d$	Bernoulli distribution over topic selectors of document d
γ	Beta prior for ψ_d
a	smoothing prior for topic distribution
b	weak smoothing prior for topic distribution
$\boldsymbol{\theta}_d$	Multinomial distribution over topics of document d
$\boldsymbol{\phi}_k$	Multinomial distribution over words of topic k
β	Dirichlet prior for $\boldsymbol{\phi}_k$
$\boldsymbol{y}_{d,k}$	indicator of topic k in document d
$\boldsymbol{w}_{d,i}$	i-th word in document d
$\boldsymbol{z}_{d,i}$	topic index of word $\boldsymbol{w}_{d,i}$
$\boldsymbol{y}_{d,\neg k}$	indicators of topics in document d except topic k
$\boldsymbol{z}_{d,\neg i}$	topics of all words in document d except i-th word
$n_d^{y=1}$	number of topics occurring in document d
$f_{d,k}^{w}$	frequency of word w generated by topic k in document d
$f_{d,k}$	frequency of words generated by topic k in document d
f_d^w	frequency of word w in document d
f_k^w	frequency of word w generated by topic k in whole corpus
f_k	frequency of words generated by topic k in whole corpus
$\lvert\boldsymbol{C}\rvert$	cardinality of collection $\boldsymbol{C}$

1) For each topic $k \in 1, \ldots, K$
   - Draw word distribution for topic k: $\boldsymbol{\phi}_k \sim Dirichlet(\beta)$
2) For each document $d \in 1, \ldots, |\boldsymbol{D}|$
   - Draw Bernoulli distribution $\boldsymbol{\psi}_d \sim Beta(\gamma)$
   - For each topic $k \in 1, \ldots, K$
     * Draw a topic indicator $\boldsymbol{y}_{d,k} \sim Bernoulli(\boldsymbol{\psi}_d)$
   - Draw topic distribution for document d, $\boldsymbol{\theta}_d \sim Dirichlet(a\boldsymbol{y}_d + b\vec{1})$
   - For each word position $i \in 1, \ldots, N_d$
     * Draw a topic $\boldsymbol{z}_{d,i} \sim Multinomial(\boldsymbol{\theta}_d)$
     * Draw a textual word $\boldsymbol{w}_{d,i} \sim Multinomial(\boldsymbol{\phi}_{\boldsymbol{z}_{d,i}})$

As we mentioned before, each short document contains several topics. Therefore, the number of topics occurring in each document is determined by topic indicators of the document. If a topic k is selected by document d, then $\boldsymbol{y}_{d,k} = 1$, otherwise $\boldsymbol{y}_{d,k} = 0$. Compared to symmetric Dirichlet prior, the spike-and-slab prior constructs asymmetric parameters of Dirichlet distribution. By setting $a \gg b$, if $\boldsymbol{y}_{d,k} = 0$,

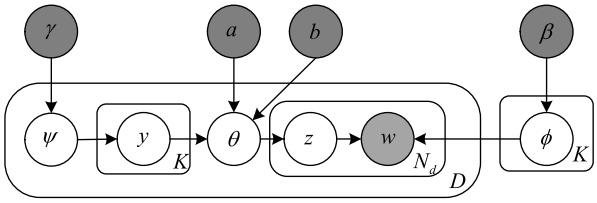


FIGURE 1. Graphic model representation of the GLTM.

then the probability of topic k occurring in document d will approach to zero. So the topic number of each short text document is determined by its content which can achieve better interpretability. The corresponding graphic model representation of the GLTM is illustrated in Figure 1. Nodes in Figure 1 colored with gray are observed variables such as hyper-parameters and words in documents which are fixed, and other nodes are parameters and latent variables which need to be estimated from the training dataset.

B. MODEL INFERENCE

Due to the coupling between latent variables in the GLTM, exact posterior inference is intractable for the model. Thus, in this article we employ the collapsed Gibbs Sampling [26] to conduct approximate inference, where "collapsed" means we integrate some parameters when sampling latent variables. In GLTM, latent variables need to be sampled are topic indicators in documents $\boldsymbol{y}$ and topic assignment of words z. Parameters such as $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, and $\boldsymbol{\psi}$ can be estimated by the maximum posterior estimation (MAP).

1) SAMPLING TOPIC INDICATORS IN DOCUMENTS Y

For a topic k in a document d, the value of indicator variable $\boldsymbol{y}_{d,k}$ is determined by variables $\boldsymbol{y}_{d,\neg k}, \boldsymbol{z}_d, a, b$ and γ according to the graphic model representation in Figure 1 since the parameter $\boldsymbol{\theta}$ is integrated. For simplicity, we utilize the ratio shown in Equation 1 to sample the indicator variable $\boldsymbol{y}_{d,k}$.

$$\frac{p(\boldsymbol{y}_{d,k}=1|\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma)}{p(\boldsymbol{y}_{d,\neg k}=0|\boldsymbol{y}_{d,k},\boldsymbol{z}_d,a,b,\gamma)} = \frac{p(\boldsymbol{y}_{d,k}=1,\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma)}{p(\boldsymbol{y}_{d,k}=0,\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma)} \quad (1)$$

Because the numerator and denominator in Equation 1 are in the same form, we can get the following form by expanding the numerator.

$$p(\boldsymbol{y}_{d,k}=1,\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma) = \int p(\boldsymbol{y}_{d,k}=1,\boldsymbol{y}_{d,\neg k},\boldsymbol{\psi}|\gamma)d\boldsymbol{\psi}\int p(\boldsymbol{z}_d,\boldsymbol{\theta}_d|\boldsymbol{y}_d,a,b)d\boldsymbol{\theta}_d \quad (2)$$

After further derivation, we can get the sampling equation for latent variable $\boldsymbol{y}_{d,k}$ by Equation 3 and 4.

$$\begin{aligned} p(\boldsymbol{y}_{d,k}=1|\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma) &\propto (\gamma_1+n_{d,\neg k}^{y=1}) \\ &*\Gamma((n_{d,\neg k}^{y=1}+1)*a+K*b) \\ &*\Gamma(a+b+f_{d,k}) \\ &*\Gamma(n_{d,\neg k}^{y=1}*a+K*b+N_d), \quad (3) \\ p(\boldsymbol{y}_{d,k}=0|\boldsymbol{y}_{d,\neg k},\boldsymbol{z}_d,a,b,\gamma) &\propto (\gamma_0+n_{d,\neg k}^{y=0}) \\ &*\Gamma(n_{d,\neg k}^{y=1}*a+K*b) \\ &*\Gamma(a+b) \\ &*\Gamma((n_{d,\neg k}^{y=1}+1)*a+K*b+N_d), \quad (4) \end{aligned}$$

where the subscript $\neg k$ in above equations indicates topic k is excluded when calculate corresponding statistics and $\Gamma(x)$ is the gamma function of x.

2) SAMPLING TOPIC ASSIGNMENTS Z

The strategy of sampling topic assignment variables for the GLTM is like the LDA model [1]. The difference is that $\boldsymbol{\theta}$ in our model is sampled from spike-and-slab priors instead of symmetric Dirichlet priors. We use the smoothing and weak smoothing priors to replace the scalar in LDA model, and the result sampling equation can be referred in Equation 5.

$$p(z_{d,i}|z_{d,\neg i},\boldsymbol{w}_{d,i},\boldsymbol{y}_d,a,b) \propto (f_{d,\neg(d,i)}^{k}+y_{d,k}*a+b)*\frac{f_{k,\neg(d,i)}^{w_{d,i}}+\beta}{f_{k,\neg(d,i)}+|V|*\beta}, \quad (5)$$

where the subscript $\neg(d,i)$ indicates that the i-th word in document d is removed when calculating these statistics. Given sufficient samples, the posterior probability $p(w|k)$ of topic k generating word w is estimated by maximum posterior estimation (MAP). That is,

$$p(w|k)=\phi_{k,w}=\frac{f_k^w+\beta}{f_k+|V|*\beta}. \quad (6)$$

C. ACQUIRING SEMANTIC SIMILARITY INFORMATION BETWEEN WORDS

As we mentioned before, word co-occurrence patterns are sparse in short documents. Two words hardly co-occur in the same document even they are semantically related. Therefore, we turn to word embeddings to obtain extra semantic knowledge of words. However, word embeddings are usually trained on large external text corpus and encoded with characteristics of the text corpus inevitably. And this characteristic information is not necessarily compatible to training dataset of topic models, which is neglected by most existing models. Inspired by [27], the continuous skip-gram model with negative sampling (SGNS) [8], [9] can reveal the semantic relationship between words and their context, which is critical to capture local semantic information for words in training dataset. So we leverage the SGNS to train local word embeddings which are encoded with local context semantic information of words. Specifically, in this article, we define the word embeddings trained from large external corpus as the global word embeddings, and define the word embeddings trained from training dataset of topic models as

the local word embeddings. We distill semantic similarity between words via both global and local word embeddings as follows.

$$SR(w_1, w_2) = cos(\tilde{\boldsymbol{v}}(w_1), \tilde{\boldsymbol{v}}(w_2)) = \frac{\tilde{\boldsymbol{v}}(w_1) \cdot \tilde{\boldsymbol{v}}(w_2)}{\|\tilde{\boldsymbol{v}}(w_1))\| * \|\tilde{\boldsymbol{v}}(w_2)\|}, \quad (7)$$

where the $SR(w_1, w_2)$ is the semantic similarity between word w_1 and w_2, and $\tilde{\boldsymbol{v}}(w_1)$ and $\tilde{\boldsymbol{v}}(w_2)$ are joint word embedding representations of w_1 and w_2, respectively. The joint word embedding $\tilde{\boldsymbol{v}}(w)$ of word w is defined as $\tilde{\boldsymbol{v}}(w) = [\boldsymbol{v}(w)_g, \boldsymbol{v}(w)_l]$ which is the joint vector of global word embedding $\boldsymbol{v}(w)_g$ and local word embedding $\boldsymbol{v}(w)_l$. Thus, the definition of semantic similarity between words is involved in general semantic information of words and local context information of training dataset. However, the dimension of local word embeddings is much smaller than global word embeddings, since the scale of training dataset is usually much smaller than large external text corpus. For a word w in the vocabulary of training dataset of topic models, we can obtain its semantically related word set $\boldsymbol{\mathcal{S}}$,

$$\boldsymbol{\mathcal{S}}(w) = \{w_o | w_o \in V, SR(w, w_o) > \epsilon\}. \quad (8)$$

D. INCORPORATING SEMANTIC INFORMATION BY GPU MODEL

Simple Pólya Urn (SPU) model [12] is a famous statistical model which has been widely used in many applications. In the SPU model, there is an urn originally contains some balls with some colors. When a ball is randomly drawn from the urn, its color is recorded, and the ball is put back in the urn along with a same color ball. In the context of topic modeling, topics and words can be regarded as urns and balls, respectively. Therefore, the proportion of balls in an urn can be analogous to word distribution under a topic. Actually, the process of Gibbs sampling of LDA model is just like the sampling process of SPU model. However, in this article, since words in a SRWS are supposed to share similar semantic information, when we see one of them in a topic, it is rational to see others under this topic with high probability. Therefore, we turn to the General Pólya Urn (GPU) model [10]. In the GPU model, when a ball is randomly drawn from the urn, the color of this ball is recorded. Then, it is put back in the urn along with a additional ball with the same color and some balls with similar colors as well. Here similar colors are corresponding to semantically related words in our model. Specifically, given a word w and its semantically related word set $\mathcal{S}$, the promotion amount of $w_o \in \mathcal{S}$ when we see w under a topic is given by Equation 7.

$$\lambda_{w,w_o} = \begin{cases} 1 & w = w_o \\ SR(w, w_o) & w_o \neq w, w_o \in \mathcal{S}(w) \\ 0 & \text{else.} \end{cases} \quad (9)$$

In this sense, a sampling word w from a topic k not only increases the proportion of word w under topic k, but also increases the proportion of words which are semantically related to w under topic k. The detail of the sampling process of the GLTM is illustrated in Algorithm 1.

Algorithm 1 Gibbs Sampling Process

Input: Hyper-parameters: $a, b, \beta, \gamma, K, \boldsymbol{\mathcal{S}}$
Output: Posterior topic-word distribution of K topics: $\{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_K\}$

1: Random initialize states before Gibbs sampling
2: **for** *iteration* $\leftarrow 1$ **to** *MaxIteration* **do**
3: **for** $d \leftarrow 1$ **to** $|\boldsymbol{D}|$ **do**
4: **for** $i \leftarrow 1$ **to** N_d **do**
5: $z_{old} \leftarrow \boldsymbol{z}_{d,i}$
6: $f_{z_{old}}^{\boldsymbol{w}_{d,i}} \leftarrow f_{z_{old}}^{\boldsymbol{w}_{d,i}} - 1$
7: $f_{z_{old}} \leftarrow f_{z_{old}} - 1$
8: **for** each $w_o \in \boldsymbol{\mathcal{S}}(\boldsymbol{w}_{d,i})$ **do**
9: $f_{z_{old}}^{w_o} \leftarrow f_{z_{old}}^{w_o} - \lambda_{\boldsymbol{w}_{d,i},w_o}$
10: $f_{z_{old}} \leftarrow f_{z_{old}} - \lambda_{\boldsymbol{w}_{d,i},w_o}$
11: **end for**
12: Sample new topic z_{new} according to Equation 3 for word $\boldsymbol{w}_{d,i}$
13: $\boldsymbol{z}_{d,i} \leftarrow z_{new}$
14: $f_{z_{new}}^{\boldsymbol{w}_{d,i}} \leftarrow f_{z_{new}}^{\boldsymbol{w}_{d,i}} + 1$
15: $f_{z_{new}} \leftarrow f_{z_{new}} + 1$
16: **for** each $w_o \in \boldsymbol{\mathcal{S}}(\boldsymbol{w}_{d,i})$ **do**
17: $f_{z_{new}}^{w_o} \leftarrow f_{z_{new}}^{w_o} + \lambda_{\boldsymbol{w}_{d,i},w_o}$
18: $f_{z_{new}} \leftarrow f_{z_{new}} + \lambda_{\boldsymbol{w}_{d,i},w_o}$
19: **end for**
20: **end for**
21: **end for**
22: **for** $d \leftarrow 1$ **to** $|\boldsymbol{D}|$ **do**
23: **for** $k \leftarrow 1$ **to** K **do**
24: $y_{old} \leftarrow \boldsymbol{y}_{d,k}$
25: $n_d^{y_{old}} \leftarrow n_d^{y_{old}} - 1$
26: Sample new value y_{new} for topic k in document d
27: $\boldsymbol{y}_{d,k} \leftarrow y_{new}$
28: $n_d^{y_{new}} \leftarrow n_d^{y_{new}} + 1$
29: **end for**
30: **end for**
31: **end for**

As shown in Algorithm 1, before Gibbs sampling, latent variables $\boldsymbol{y}$ and $\boldsymbol{z}$ are random initialized, and the corresponding statistics are accumulated. After random initialization, latent variable $\boldsymbol{z}_{d,i}$ is sampled for i-th word in document d. When a new topic is sampled, the statistics of word $\boldsymbol{w}_{d,i}$ under the new topic is promoted along with words which are semantically related with the $\boldsymbol{w}_{d,i}$ (Line 12 -20). The iterative process continues under the predefined number of maximum iteration is reached, then parameters of the model are obtained by MAP. By sampling strategy of the GPU model, semantic coherence of topics can be effectively strengthened. From Algorithm 1, we can read out the time complexity of one iteration of Gibbs sampling is $O(|\boldsymbol{D}|lK + |\boldsymbol{D}|K)$ where the $|\boldsymbol{D}|$ is the size of dataset $\boldsymbol{D}$, and l is the average length of documents in dataset $\boldsymbol{D}$.

TABLE 2. Statistical information of datasets.

Dataset	#docs	#avg doc-length	#terms	#categories
Web Snippet	12340	14.6	5432	8
Amazon Review	19980	17.5	14331	7
Yahoo Answers	6310	117.4	15776	-
Tweet2011	30946	7.5	8536	-

IV. EXPERIMENTS

In this section, we evaluate performance of the GLTM on four real short text datasets, compared with five state-of-the-art short text topic models. We conduct experiments for both topic coherence and short text classification to exhibit the promising experimental results of the GLTM in comparative study.

A. DATASETS

1) WEB SNIPPET [28]

This dataset contains 12340 search snippets, and each snippet is regarded as a short document. Snippets are categorized into 8 clusters.

2) AMAZON REVIEW [29]

The original dataset spanning May 1994 - June 2014 contains product reviews and metadata from Amazon. We randomly sample 20000 short reviews as training dataset and each review belongs to one of 7 categories.

3) YAHOO ANSWERS [30]

This dataset contains 6310 pairs of questions and corresponding answers. Each question and the corresponding answers together can be regarded as a short document.

4) TWEET2011

[1] From this dataset, we obtain 32000 short tweets. As the **Yahoo Answers** dataset, there are no labels for documents.

For all above datasets, we perform the following preprocessing:

- Lowercase all terms in documents.
- Remove stop words, non-alphabetic characters, and punctuations.
- Remove words occurring less than 5 documents.
- Remove documents with document length less than 3 words.

After preprocessing, the statistical information of the four datasets is shown in Table 2. The symbol ''#doc'' represents the number of documents in each dataset. ''#avg doc-length'' represents the average length of documents. ''#terms'' is the length of vocabulary of each dataset. And ''#categories'' is the category number of documents in each dataset.

Besides, we leverage 2016 Wikipedia dataset[2] to train global word embeddings for GLTM and GPU-DMM model. The word2vec[3] proposed by Google is employed to train word embeddings with SGNS model. And the dimension of word embeddings is set to 300. We also use this tool to train local word embeddings on four short text training datasets for the GLTM, the dimension of local word embeddings is 30.

B. COMPARED MODELS

We compare the performance of the GLTM with five state-of-the-art short text topic models, which can be referred as follows.

DMM [7] takes the simple assumption that each short document contains only one topic, which is not reasonable for normal documents, but rational for short documents in most cases.

BTM [6] models the generation process of biterms which are composed of unordered two words in a slide-window. By this mean, it directly models word co-occurrence patterns of corpus.

GPU-DMM [10] distills semantic relatedness between words from word embeddings, and leverages GPU model in Gibbs sampling process to utilize these semantic information to improve coherence of topics.

PTM [5] assumes that each short document is sampled from a long pseudo document, and topics are generated by long pseudo documents. By extracting the corresponding relationship between short documents and long pseudo documents, the sparsity problem could be effectively alleviated.

SPARSEPTM [5] replaces the symmetric Dirichlet prior in PTM with the *spike and slab* prior to obtain focused topics in each long pseudo document.

We set common parameters of these models uniformly, such as $\alpha = 50/K$, $\beta = 0.01$, the maximum iteration of all models $maxIteraion = 1500$. For the SparsePTM and GLTM, smoothing prior $a = 50/K$ and weak smoothing prior $b = 1E-7$. Other parameters of PTM and SparsePTM are set according to suggestion of the original literature [5], and the number of long pseudo documents of PTM and SparsePTM is set to 500 according to average document length in each dataset. To obtain semantic similarity between words, the parameter ϵ in the GLTM and GPU-DMM model is set to 0.5.

C. TOPIC COHERENCE

We employ the *topic coherence* proposed in literature [31] to measure the semantic coherence of topics distilled by topic models. Given a topic k, the *topic coherence* of k is calculated as follow.

$$C(k, \boldsymbol{v}^k) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} log \frac{|\boldsymbol{D}(\boldsymbol{v}_m^k, \boldsymbol{v}_l^k)| + 1}{|\boldsymbol{D}(\boldsymbol{v}_l^k)|}, \tag{10}$$

where $\boldsymbol{v}^k = [\boldsymbol{v}_1^k, \boldsymbol{v}_2^k, \ldots, \boldsymbol{v}_M^k]$ is the top M words under the topic k sorted by probability in descending order. $|\boldsymbol{D}(\boldsymbol{v}_m^k, \boldsymbol{v}_l^k)|$ is the number of documents in dataset containing both word

[1] https://trec.nist.gov/data/tweets/

[2] https://dumps.wikimedia.org/enwiki/

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

TABLE 3. Experimental results of topic coherence.

Web Snippet dataset						
#topics	PTM	SPTM	BTM	DMM	GPU-DMM	GLTM
20	-976.44	-1026.03	-900.72	-946.27	-914.61	**-886.48**
40	-967.30	-1035.15	-879.23	-909.18	-883.21	**-872.20**
60	-918.62	-1088.98	-859.49	-880.34	-859.56	**-842.20**
80	-907.55	-1096.38	-833.57	-870.76	-851.32	**-830.38**
Amazon Review dataset						
20	-941.02	-923.95	-851.76	-855.51	**-821.45**	-848.40
40	-956.82	-991.35	-848.21	-913.34	-865.03	**-843.84**
60	-939.71	-987.40	-878.32	-912.74	-896.44	**-866.47**
80	-917.86	-907.86	-888.90	-916.86	-907.82	**-859.87**
Yahoo Answers dataset						
20	-634.63	-631.93	-638.72	-713.33	-711.21	**-624.03**
40	-652.72	-644.45	**-625.26**	-709.29	-712.90	-647.74
60	-665.94	-659.01	-666.32	-710.11	-711.45	**-655.72**
80	-679.75	-660.29	-681.49	-712.73	-711.47	**-642.55**
Tweet2011 dataset						
20	-1150.63	-1122.34	-1081.29	-1107.77	-1046.71	**-1025.11**
40	-1115.14	-1126.58	-1034.27	-1062.75	**-992.57**	-1019.32
60	-1096.14	-1127.21	-1049.48	-1055.41	-997.88	**-993.11**
80	-1101.27	-1176.21	**-1026.84**	-1033.28	-1037.53	-1031.74

$\boldsymbol{v}_m^k$ and $\boldsymbol{v}_l^k$. The performance of a topic model can be quantified as the average *topic coherence* of topics distilled by the topic model. The higher *topic coherence*, the better model performs.

In this task, we set the number of topics for models, $K = \{20, 40, 60, 80\}$, and the experimental results of topic coherence are reported in Table 3. We can see that the GLTM gains a clear edge compared to other benchmark models on all datasets, which demonstrates the effectiveness of our model to some extent. The GPU-DMM model and BTM achieve the suboptimum results compared with other models. The BTM models the word co-occurrence patterns directly, so its performance is not influenced by length of documents and shows the robustness across different datasets. When word co-occurrence patterns are sparse, word embeddings can truly bring additional knowledge for topic model. This can be observed when comparing GPU-DMM model with DMM model on different datasets. However, GPU-DMM model and DMM model are crashed on Yahoo Answers dataset, this is because the average length of documents in this dataset is relatively larger than other datasets. The assumption that each short document contains only one topic is not applicable for this dataset. On the other hand, the GPU-DMM model does not consider the situation that word embeddings trained on large external corpus is not necessary suitable for training dataset. So the performance of GPU-DMM model is worse than the GLTM in most cases. We can also observe that PTM and SparsePTM perform poorly on above datasets excepting Yahoo Answers dataset. These two models are complex, which cause the poor performance when facing to small scale short text dataset.

Topics of each short document is decided by its content according to the spike-and-slab prior and MAP, and semantic coherence of each topic can be further strengthened by the GPU model in inference process. Therefore, the GLTM gains the best results in most cases when average length of documents differs largely among all datasets. We randomly select some topics distilled by the GLTM and exhibit them in Table 4.

Table 4 shows topics distilled by the GLTM on four datasets when $K = 20$. Each topic is represented by top

TABLE 4. Exhibition of topics distilled by the GLTM on four dataset ($K = 20$).

Web Snippet dataset	Topic 0: processor computer cpu intel hardware linux software processors server memory Topic 12: software browser xml programming web programs java algorithms language javascript
Amazon Review dataset	Topic 0: string guitar guitars bass drums keyboards mandolin sound tuner acoustic Topic 65: headset earphone headphone sound volume quality ear volumes **cord** hear
Yahoo Answers dataset	Topic 5: vegetarianism food meat livestock nutrition vegetable milk vegan vitamin seafood Topic 19: arabic language urdu alphabet arab islamic hindi israel palestinian sanskrit
Tweet2011 dataset	Topic 4: school collage education faculty university students academy teacher campus poetry Topic 11: email twitter facebook tweet hashtag myspace youtube linkedin blogging tweetdeck

10 words sorted by probability in descending order under this topic. We can see that words under each topic are semantically coherent. From textual words, we can easily infer the meaning of each topic, which qualitatively demonstrates the high quality of topics.

D. SHORT TEXT CLASSIFICATION

Topics distilled by topic models can be regarded as a low dimensional representation of short documents. Compared with bag of words representation, this low dimensional representation of short documents is dense, which can be leveraged directly by classical classifiers and clustering algorithms. Besides, quality of topics can be reflected by downstream applications. In this task, we leverage topics distilled by topic models as features of short documents and apply Random Forest[4] to classify these short documents. Quality of topics can be measured by results of classifications. Topic distribution under document d is given by the following equation.

$$p(z = k|\vec{d}) \propto \sum_{w \in D_d} p(z = k|w)p(w|\vec{d}), \tag{11}$$

where $p(w|\vec{d})$ can be estimated by word frequency in document d, and $p(z = k|w)$ can be obtained by bayesian rules: $p(z = k|w) \propto p(z = k)p(w|z = k)$. For parameters of Random Forrest, we set $n\_estimators = 80$ and $max\_depth = 15$, which is the tradeoff between efficiency and accuracy. Since the Yahoo Answers dataset and Tweet2011 dataset are lack of label information, we conduct short text classification experiments on Web Snippet dataset and Amazon Review dataset. We leverage the classification accuracy and F1 measure (Equation 12) to quantify results of classifications and

[4]http://scikit-learn.org/

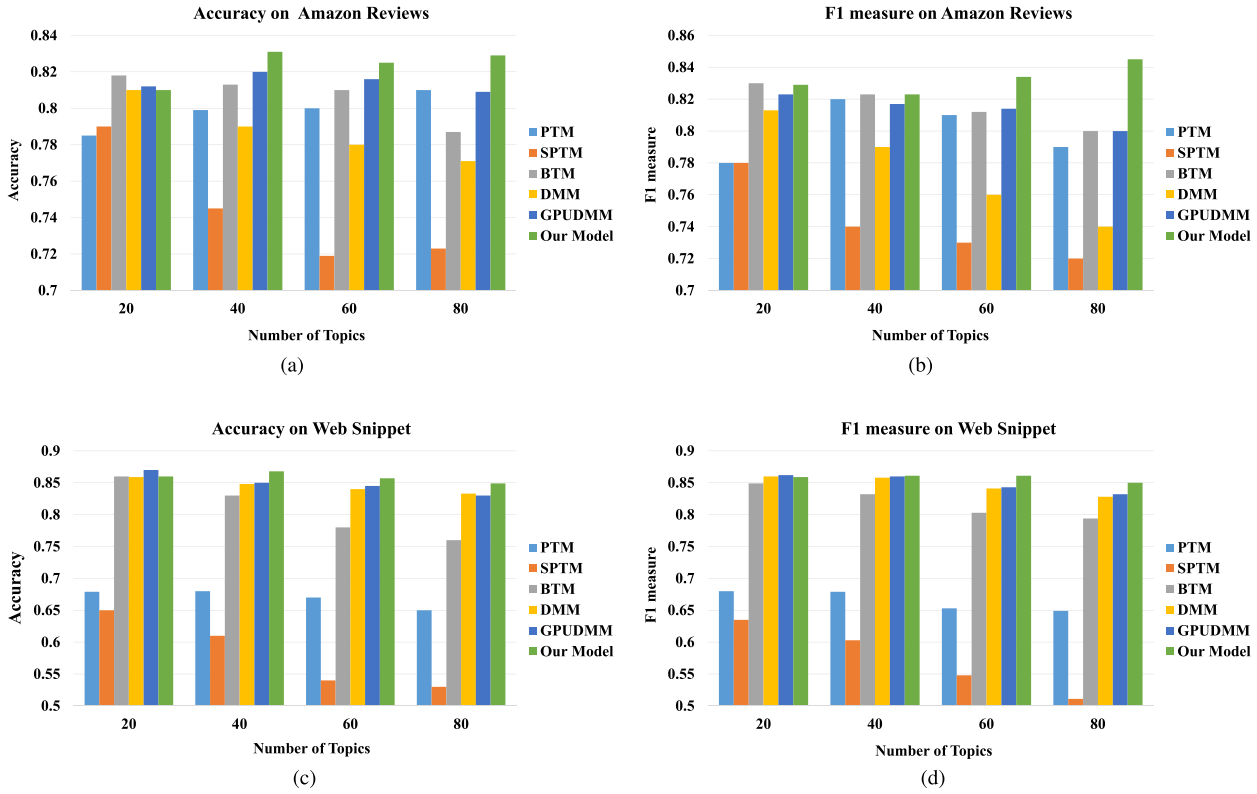


FIGURE 2. Performance evaluation for short text classification. (a) Accuracy of Amazon Reviews. (b) F1 measure of Amazon Reviews. (c) Accuracy of Web Snippet. (d) F1 measure of Web Snippet.

results are shown in Figure 2.

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \tag{12}$$

From Figure 2, we can see that the proposed GLTM gains the best performance in classification experiments in most cases, which are consistent with results of topic coherence. Besides, our model is robust in different settings of topic number, whereas performance of other models is mostly declined with the increase of topic number.

E. EFFICIENCY ANALYSIS

In this section, we evaluate the efficiency of different models. We leverage the Gibbs sampling for parameters estimation for all models. The models are implemented in Java language on the same hardware environment. The time costs of one iteration on four datasets are shown in Table 5.

The DMM is the most efficient model among these models, since the simple assumption. The proposed GLTM achieves a comparable performance with GPU-DMM model. The BTM has shown its robustness on different datasets without external knowledge, but time complexity impedes wide application of this model. The graphic structure of PTM is complicated, and there are plenty of latent variables need to be sampled which cause the huge time consumption. Considering the quality of distilled topics and efficiency of the GLTM, the experimental results demonstrate the high usability of the proposed GLTM.

TABLE 5. Time cost of one iteration on four datasets (seconds / iteration).

Web Snippet dataset						
#topics	PTM	SPTM	BTM	DMM	GPU-DMM	GLTM
20	1.24	0.441	0.435	0.051	0.112	0.213
40	1.79	0.536	0.778	0.097	0.203	0.319
60	2.21	0.672	1.25	0.156	0.274	0.436
80	2.68	0.791	1.54	0.194	0.357	0.572
Amazon Review dataset						
20	1.63	0.55	0.77	0.074	0.181	0.125
40	2.29	0.72	1.46	0.165	0.175	0.219
60	2.71	0.86	2.16	0.284	0.301	0.371
80	3.39	1.01	2.89	0.477	0.512	0.615
Yahoo Answers dataset						
20	2.12	0.503	1.36	0.225	0.353	0.493
40	2.76	0.783	2.53	0.383	0.487	0.614
60	3.49	0.913	3.67	0.448	0.712	0.847
80	4.25	1.34	4.71	0.516	0.989	1.15
Tweet2011 dataset						
20	1.92	0.472	0.267	0.055	0.219	0.333
40	2.37	0.644	0.481	0.102	0.351	0.559
60	2.93	0.720	0.718	0.174	0.598	0.792
80	3.46	0.983	0.101	0.263	0.792	1.02

V. CONCLUSION

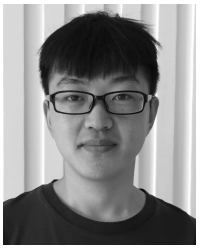
Due to the sparsity of word co-occurrence patterns in short texts, traditional topic models are prevented from extracting semantically coherent topics from short text corpus. Word embeddings encoded with general syntactic and semantic

information of words can be regarded as external knowledge for topic models. However, word embeddings are trained on large text corpus and the encoded information is not necessary suitable for training dataset of topic models. In this article, we use SGNS to train local word embeddings to capture context information for each word in training dataset. Through global word embeddings and local word embeddings, we obtain semantic relatedness information between words. Besides, we propose a new generation process for short text collections which incorporates spike-and-slab priors to decide topic number for each short document according to its content. In inference process, we employ GPU model as sampler, which leverages semantic relatedness information between words to strengthen semantic coherence of topics. Experimental results demonstrate the effectiveness of the GLTM in terms of model efficiency and quality of distilled topics.

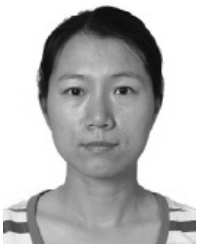
REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[2] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Mach. Learn.*, vol. 42, no. 1, pp. 177–196, Jan. 2001.
[3] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, "Improving LDA topic models for microblogs via tweet pooling and automatic labeling," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2013, pp. 889–892.
[4] X. Quan, C. Kit, Y. Ge, and S. J. Pan, "Short and sparse text topic modeling via self-aggregation," in *Proc. IJCAI*, 2015, pp. 2270–2276.
[5] Y. Zuo *et al.*, "Topic modeling of short texts: A pseudo-document view," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 2105–2114.
[6] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 1445–1456.
[7] J. Yin and J. Wang, "A Dirichlet multinomial mixture model-based approach for short text clustering," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 233–242.
[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean. (2013). "Efficient estimation of word representations in vector space." [Online]. Available: https://arxiv.org/abs/1301.3781
[10] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma, "Topic modeling for short texts with auxiliary word embeddings," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2016, pp. 165–174.
[11] G. Xun, V. Gopalakrishnan, F. Ma, Y. Li, J. Gao, and A. Zhang, "Topic discovery for short texts using word embeddings," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, 2016, pp. 1299–1304.
[12] H. Mahmoud, *Pólya Urn Models*. Boca Raton, FL, USA: CRC Press, 2008.
[13] W. X. Zhao *et al.*, "Comparing Twitter and traditional media using topic models," in *Proc. Eur. Conf. Inf. Retr.*, 2011, pp. 338–349.
[14] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proc. 1st Workshop Social Media Anal.*, 2010, pp. 80–88.
[15] T. Lin, W. Tian, Q. Mei, and H. Cheng, "The dual-sparse topic model: Mining focused topics and focused terms in short text," in *Proc. 23rd Int. Conf. World Wide Web*, 2014, pp. 539–550.
[16] Z. Chen, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh, "Leveraging multi-domain prior knowledge in topic models," in *Proc. IJCAI*, vol. 13, 2013, pp. 2071–2077.
[17] Z. Chen and B. Liu, "Topic modeling using topics from many domains, lifelong learning and big data," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 703–711.
[18] Z. Chen and B. Liu, "Mining topics in documents: Standing on the shoulders of big data," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 1116–1125.
[19] T. Shi, K. Kang, J. Choo, and C. K. Reddy, "Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations," in *Proc. World Wide Web Conf. (WWW)*, 2018, pp. 1105–1114.
[20] V. K. R. Sridhar, "Unsupervised topic modeling for short texts using distributed representations of words," in *Proc. 1st Workshop Vector Space Modeling Natural Lang. Process.*, 2015, pp. 192–200.
[21] R. Das, M. Zaheer, and C. Dyer, "Gaussian LDA for topic models with word embeddings," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 795–804.
[22] J. Qiang, P. Chen, T. Wang, and X. Wu, "Topic modeling over short texts by incorporating word embeddings," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, 2017, pp. 363–374.
[23] P. Xie, D. Yang, and E. Xing, "Incorporating word correlation knowledge into topic modeling," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2015, pp. 725–734.
[24] C. Wang and D. M. Blei, "Decoupling sparsity and smoothness in the discrete hierarchical Dirichlet process," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1982–1989.
[25] J. Tang, Z. Meng, X. Nguyen, Q. Mei, and M. Zhang, "Understanding the limiting factors of topic modeling via posterior contraction analysis," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 190–198.
[26] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 1, pp. 5228–5235, 2004.
[27] Y. Goldberg and O. Levy, "word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method," *CoRR*, Feb. 2014. [Online]. Available: https://arxiv.org/abs/1402.3722
[28] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi, "Learning to classify short and sparse text & Web with hidden topics from large-scale data collections," in *Proc. 17th Int. Conf. World Wide Web*, 2008, pp. 91–100.
[29] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 165–172.
[30] M.-W. Chang, L.-A. Ratinov, D. Roth, and V. Srikumar, "Importance of semantic representation: Dataless classification," in *Proc. AAAI*, vol. 2, 2008, pp. 830–835.
[31] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 262–272.

WENXIN LIANG (M'08) received the B.E. and M.E. degrees from Xi'an Jiaotong University, China, in 1998 and 2001, respectively, and the Ph.D. degree in computer science from the Tokyo Institute of Technology, Japan, in 2006. He was a Post-Doctoral Research Fellow with CREST of Japan Science and Technology Agency and a Guest Research Associate with GSIC, Tokyo Institute of Technology, from 2006 to 2009. He was an Associate Professor with the School of Software, Dalian University of Technology, China, from 2009 to 2018. He is currently with the School of Software Engineering, Chongqing University of Posts and Telecommunications, China. His main research interests include data engineering, artificial intelligence, and social networks. He is a Senior Member of the China Computer Federation, and a member of the ACM, ACM SIGMOD Japan Chapter, and the Database Society of Japan.



RAN FENG received the B.S. degree in software engineering from the Dalian University of Technology, China, in 2015, where he is currently pursuing the master's degree in software engineering with the School of Software. His research interests include probabilistic machine learning and data mining.



XINYUE LIU received the B.S. and M.S. degrees from Northeast Normal University, China, in 2003 and 2006, respectively, and the Ph.D. degree in computer science from the Dalian University of Technology, China, in 2012. She is currently an Associate Professor with the School of Software, Dalian University of Technology, China. Her research interests include data mining, machine learning, and information retrieval.

YUANGANG LI received the MBA degree from the Dongbei University of Finance and Economics in 2006. He is currently pursuing the Ph.D. degree with the School of Information Management and Engineering, Shanghai University of Finance and Economics, China. His research interests include complex system modeling and data mining.

XIANCHAO ZHANG received the bachelor's and master's degrees in mathematics from the National University of Defense Technology, China, in 1994 and 1998, respectively, and the Ph.D. degree in computer science from the University of Science and Technology of China in 2000. From 2000 to 2003, he was a Research and Development Manager in some international companies. He joined Dalian University of Technology in 2003. He is currently a Full Professor with the Dalian University of Technology, China. His research interests include design and analysis of algorithms, machine learning, data mining, and information retrieval.