Collaboratively Modeling and Embedding of Latent Topics for Short Texts

ZHENG LIU (Member, IEEE), TINGTING QIN, KE-JIA CHEN (Member, IEEE), AND YUN LI (Member, IEEE)
School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
Corresponding author: Zheng Liu (zliu@njupt.edu.cn)

This work was supported in part by the Jiangsu Provincial Natural Science Foundation of China under Grant BK20171447, and in part by the Nanjing University of Posts and Telecommunications under Grant NY219084.

ABSTRACT Deriving a successful document representation is the critical challenge in many downstream tasks in NLP, especially when documents are very short. It is challenging to handle the sparsity and the noise problems confronting short texts. Some approaches employ latent topic models, based on global word co-occurrence, to obtain topic distribution as the representation. Others leverage word embeddings, which consider local conditional dependencies, to map a document as a summation vector of them. Unlike the existing works which explore the strategy of utilizing one to help the other, i.e., topic models for word embeddings or vice versa, we propose CME-DMM, a collaboratively modeling and embedding framework for capturing coherent latent topics from short texts. CME-DMM incorporates topic and word embeddings through the attention mechanism and implants them into the latent topic models, which significantly improve the quality of latent topics. Extensive experiments demonstrate that CME-DMM could perceive more coherent topics than other popular methods, resulting in a better performance in downstream NLP tasks such as classification. Besides the interpretable latent topics, the corresponding topic embeddings can describe the meanings of latent topics in the semantic space. The attention vectors, as a by-product of the learning process, can identify the keywords in noisy short texts.

INDEX TERMS Latent topic models, short texts, text classification, text representation, topic embedding.

I. INTRODUCTION
The overwhelming volume of short texts from social networks like Twitter brings new challenges of understanding and processing these short texts. Many downstream NLP tasks on large corpora of short texts expect good document representations serving as the cornerstones of their solutions. Short texts are often informal, noisy, and in short of regular patterns, featuring inordinate structure and colloquialism. It is not an easy task to obtain a satisfactory representation of short texts.

Traditional approaches for modeling long documents such as Latent Dirichlet Allocation (LDA) [1] and Latent Semantic Analysis (LSA) [2] are unable to capture high-quality latent topic representations because they depend on the word co-occurrences in documents. While in short texts, there are not adequate instances [3].

One way to apply traditional topic models on short texts is to merge similar short texts into long virtual documents [4], [5], where similar short texts could be the short texts of the same author or the same theme. Consequently, it seems that merging short texts is promising, but these virtual documents create lots of non-existing word co-occurrence instances [6]. Another direction is to modify the classic topic models with additional constraints to fit them into the circumstances of short texts. For example, in Dirichlet Multinomial Mixtures (DMM) [7], it complies that each short text can only contain a single topic. Methods in this manner outperform the classic topic models on short texts, but the quality of latent topics might deteriorate due to the additional constraints.

Topic models capture the global word co-occurrences while ignoring their sequence information [8], i.e., local contextual dependencies, which are valuable in document representations. Word embeddings, on the other hand, exploit local conditional dependencies in small context windows to map words into continuous vectors as the word representation.
in the semantic space [9]. Then the summation vector of words in a document could serve as its representation. Unlike the document representations based on discrete latent topics, which are interpretable semantic topic distributions, the approaches based on word embeddings represent documents as continuous vectors in the semantic space of unexplainable dimensions.

Essentially, both topic modeling and word embeddings leverage word co-occurrences in documents where their difference lies in the different position constraints on co-occurring words. There are a few research works [8], [10], [11] exploring the strategy of utilizing one to help the other, i.e., topic models for word embeddings or vice versa. For example, pre-trained word embeddings are introduced to the topic model to supplement the semantic information between words. The generation process in topic models employs the long short-term memory (LSTM) learning structure to predict the next word.

Base on the above observations, we identify the following key challenges in modeling latent topics for short texts: (1) The topic models should handle the sparsity issue in short texts, where the word co-occurrence instances are inadequate, without imposing additional constraints. (2) The topic models should capture local contextual dependencies in noisy short texts because of their limited length. (3) The latent topic models based on word co-occurrences should be consistent with their corresponding representations based on local contextual dependencies, resulting in improved qualities on both sides.

To address these challenges, We proposed CME-DMM, a Collaboratively Modeling and Embedding framework, for capturing latent topics in short texts. As the name indicates, we incorporate the DMM topic model and the word/topic embeddings collaboratively through the attention mechanism, in order to improve both the quality of topic models and embeddings.

- CME-DMM considers both global word co-occurrence relationship through DMM, and the local (contextual) word co-occurrence relationship through word/topic embeddings, which significantly improve the quality of the latent topics, as we can see from the experimental evaluations.

- The quality of the word and the topic embeddings learned by CME-DMM is improved by making the topic embeddings and the word embeddings more consistent with each other in the latent semantic space. The attention scores reveal this consistency and make sure that the learning process pays more attention to topic-relevant words.

We employ Dirichlet Multinomial Mixtures (DMM) [7] as the topic model for short texts, but it is worth noting that our framework is compatible with many other topic models.

The contributions of this paper are summarized as follows:

- We proposed a stochastic EM algorithm to infer the optimal model parameters by maximizing the overall generation probability with pre-trained word embeddings.

- We conducted extensive experiments to compare our proposed model with existing hybrid models to demonstrate the high quality of the discovered topic models and the embeddings, as well as the better performance in downstream NLP tasks.

The following paper is organized as follows. We discuss the related work in Section II. We introduce the preliminary background in Section III and our proposed framework in Section IV. We explain in detail how to infer the model parameters in Section V. With extensive experiments, we demonstrate in Section VI that the acquired latent topics are of high quality, and can obtain better results in downstream NLP tasks. Finally, we conclude this paper in Section VII.

II. RELATED WORK

Many researchers proposed varied topic models, starting off with the classic LDA [1] proposed by Blei et al., because of the wide usage of topic models in various applications. In this section, we first summarize typical applications of latent topic models. Recently, topic models utilizing word embeddings and deep learning techniques show better performance, and we conduct a detailed discussion of these models, as well as their pros and cons compared with the proposed model in this paper. To avoid redundancy, we do not include traditional latent topic models in this section, which readers can find in many related surveys [12], [13].

A. TOPIC MODELS IN APPLICATIONS

Latent topic models are widely used in tasks that involve handling texts in natural languages, such as text mining and opinion analysis. For example, social media analysis based on latent topic models can help understand users’ relationships and conversations in online communities. Hereinafter, we briefly introduce some applicable domains of latent topic models.

In computational linguistics, latent topic models such as LDA is employed to compute the statistical correlations between words in documents to identify and quantify the latent topics in these documents. Vulić et al. [14] discovered the translations of terms in corpus without any language resource by introducing the bilingual Latent Dirichlet Allocation (Bilingual-LDA). Eidelman et al. [15] computed topic-related lexical weighted probabilities by incorporating weighted topic distributions into the translation model as features to guide the machine translation system through topic-relevant translations. Heintz et al. [16] employed LDA on Wikipedia to align source subjects and target concepts to identify sentences for discovering potential metaphors. Liu et al. [17], based on LDA, proposed a framework to identify the languages used in documents and calculate the relative proportions in multilingual documents.

In social network analysis, topic models are effective in user behavior analysis and evaluation. Researches proposed
various LDA-based methods to analyze short texts on social networks, such as tweets on Twitter. Weng et al. [4] proposed TwitterRank, a method extended from the PageRank algorithm, to rate uses on Twitter for identifying the most influential Twitterers, where LDA detects potential topics in Tweets. Hong et al. [18] proposed an approach for predicting message popularity based on the number of possible retweets and tweets in the future. They used LDA to model the subject distributions of tweets and showed that the discovered tweets could attract thousands of retweets. Shi et al. [19] proposed dynamic topic modeling via a self-aggregation method (SADTM) to capture time-varying topics, which aggregates short texts into pseudo documents to resolve the sparsity issue.

In software engineering, topic models show high potentials in analyzing source codes and software evolutions. Linstead et al. [20] are the first to use LDA in source code analysis and software similarity visualization for effective project management and software refactoring, where topic distributions obtained from LDA serve as the features of source files. Gethers and Poshhyvanyk [21] proposed a new coupling metric based on a relational topic model (RTM) to analyze software relationships from latent topic distributions among software management data, which showed a valuable impact on the analysis of sizeable open-source software systems. Gao et al. [22] proposed a weighted Conditional random field regularized Correlated Topic Model (CCTM), which utilized semantic correlations between topics, for generating topic evolutionary graphs for question titles from Stack Overflow, an open community for coding questions.

In social studies, topic models are necessary for opinion or activity analysis. For example, in political science, analyzing speeches such as plenary sessions of parliament can provide insight into the political priorities of the politicians under consideration. Greene and Cross [23] proposed a new two-layer matrix factorization method to extract topics that change over time in a large corpus of political speech and to identify topics related to events in time. Some researches applied topic modeling methods in the field of crime prediction and activity analysis. Chen et al. [24] developed an early warning system based on LDA and a collaborative representation classifier to detect criminal activity intentions. Sharma et al. [25] introduced a crime intensity geographic model to detect the safest path between two locations, which employs a naïve Bayes classifier with features derived from the LDA model. Kou et al. [26] proposed a Spatial and Temporal Topic Model (STTM), which focuses on analyzing influential social events based on both spatial and temporal characteristics.

Other surprising applications of topic models include healthcare and geography. Topic modeling is advantageously applied to large biomedicine data sets to analyze and evaluate useful content. Xiao et al. [27] demonstrated that LDA-based approaches could successfully reveal probabilistic patterns between adverse drug reactions (ADRs). Zhang et al. [28] built a healthcare recommendation system by using topic models for doctor characteristics and obtained improved recommendation accuracy. Yin et al. [29] extracted topics from GPS-related documents and combine them with geographic clustering for geographic topic analysis. Tang et al. [30] proposed a multi-scale LDA model for clustering satellite images, which is a combination of multi-scale image representations and probabilistic topic models.

B. TOPIC MODELS WITH WORD/TOPIC EMBEDDINGS

Many researchers proposed various topic models for short texts. Yan et al. [31] proposed a Bi-term Topic Model (BTM) to decompose topic concepts into words as the document features. Quan et al. [32], based on BTM, proposed a generative topic model for virtual documents merged from short texts. The large number of parameters for virtual documents lower the converging speed significantly. Nigam et al. [7] proposed Dirichlet Multinomial Mixtures (DMM) for short texts, which is simple yet effective, and applied widely in many tasks involving short texts. Yin and Wang [33] combined DMM with partition rules, and derived a DMM variation for text clustering.

Recently, researches paid some effects on coupling the latent topic models with state-of-the-art deep learning techniques, intending to improve the performance and quality of both sides. In the following, we will discuss some typical strategies that are closely related to this paper, including latent topic models utilizing word embeddings, generative topic embeddings, and topic model variations based on neural networks.

1) LATENT TOPIC MODELS UTILIZING WORD EMBEDDINGS

Nguyen et al. [10] proposed LF-LDA and LF-DMM, variations of LDA and DMM with Latent Features. Both models use a categorical distribution to generate words based on word embeddings. These external word embeddings complement the relationships between words in short texts. They employed a switch variable to decide whether the word is generated from LDA/DMM or word embeddings. Hu and Tsujii [34] proposed a Latent Concept Topic Model (LCTM) for revealing topics via co-occurrences of latent concepts. These latent concepts are localized Gaussian distributions in the semantic space of word embeddings. Shi et al. [35] proposed a unified framework for word embedding and topic modeling for long documents. They considered only contextual word-occurrence relationships based on the skip-gram model, resulting in latent topics represented by bi-gram distributions and word embeddings per topic, which significantly increased the size of topic representations and extra computation cost. Gao et al. [36] designed a Conditional Random Field regularized Topic Model (CRFTM), which assigns semantically related words to the same topic more probably. Word embeddings serve as the similarity measure to quantify the semantic correlations among words.

Unlike the above methods with excessive topic representations or additional computational cost, our proposed CME-DMM considers both global word-occurrence relationship through DMM, and the local (contextual) word-occurrence
relationship through word/topic embeddings. CME-DMM captures the topic influence in learning the word and the topic embeddings and improves the quality of topic models by taking it into consideration that words in different topics should have different importance in latent topics during the word generation process.

2) GENERATIVE TOPIC EMBEDDING
Li et al. [37] correlated the latent topics based on topic embedding in order to lower the computational cost of modeling topics. Topic embeddings and topic mixing proportions together represent documents in low-dimensional continuous space. Jiang et al. [8] proposed Latent Topic Embedding (LTE), in which each sentence in documents is associated with one topic. Words in the same sentence have the same topic. LTE simply performed an element-wise addition of word embeddings and topic embedding as the representation of sentences.

Although the sources of word generation in LTE are similar to the ones in the proposed CME-DMM, it does not examine the diverse influences of words to topics, resulting in topics with possible incompatible words. CME-DMM captures the topic influence during learning word and topic embeddings, which yields more consistent words and topics in the latent semantic space. Besides, we target on short texts and each short text could contain several sentences. In our CME-DMM, we use DMM, which is proved to be working well on short texts by many other publications. Moreover, we can easily change the topic models based on the characteristics of the corpus.

3) TOPIC MODEL VARIATIONS BASED ON NEURAL NETWORKS
Zaheer et al. [11] brought Long Short-Term Memory (LSTM) to the topic models. With the assumption that the topic of a word in a sliding window depends on previous words, they proposed three models, i.e., Topic LLA, Word LLA, and Char LLA, where LLA means Latent LSTM Allocation. Topic LLA employs LSTM to predict the topic of the current word based on the topics of previous words. Word LLA employs LSTM to predict the word based on previous words. The input feature of LSTM could be word sequences or topic sequences. Li et al. [38] proposed the Recurrent Attentional Topic Model (RATM) for document embedding, which takes into account the adjacent sentences, and uses attention mechanism to model the relations among successive sentences. RATM focuses on the coherence of successive sentences, i.e., previous sentences and the current sentence share some coherent topics, so the attention mechanism is employed at the granularity of sentences.

The above methods ignore the global word co-occurrences, and our CME-DMM advance in the following aspects. CME-DMM learns word and topic embedding simultaneously, and words can be generated by sampling from topic-word distribution, or based on topic and word embeddings with the help of attention mechanism. CME-DMM focuses on collaboratively modeling and embedding of latent topics, where the attention mechanism is employed at word/topic level, not the sentence level, because the goal is to discover topic models of high quality, as well as topic and word embeddings, which are more consistent with each other in the latent semantic space.

To summarize, CME-DMM conducts topic/word embedding and latent topic modeling collaboratively. Topic embeddings and word embeddings compromise each other through an attention mechanism. In this way, CME-DMM takes into conversion the relationships between topics and corresponding words and favors topic relevant words to avoid the problem of non-existing word co-occurrences. CME-DMM can fine-tune the topic and word embeddings during the learning process, resulting in more reasonable representational characteristics and better performance reported in Section VI.

III. PRELIMINARIES
The notations used in this paper are summarized in Table 1. Bold lowercase letters indicate vectors, and bold uppercase letters indicate matrices. We use $\otimes$ to represent the element-wise division of matrices or vectors, $o$ to represent the function combination, and $\Delta^p$ to represent the set of $p$ dimension vectors. $D = \{X_1, \ldots, X_n, \ldots, X_M\}$ represents a short text corpus of size $M$, where $X_n$ is the $n$-th short text.

In this section, we first introduce the background knowledge of Dirichlet Multinomial Mixture (DMM) [7], which we employ as the foundation of the proposed framework. The DMM model is a probabilistic generation model for short texts. Originally, DMM is proposed to augment labeled documents by unlabeled documents. DMM uses a mixture model for the generation process [39] and poses the constrain that there is a one-to-one relationship between individual samples (words) and classes (topics) [40].

Given a corpus of short texts, $D = \{X_n\}_{n=1}^M$, Fig. 1 shows the Bayesian network diagram of the generation process in DMM, where the topic distribution $\theta$ and the topic-word

| TABLE 1. Notations. |
|----------------------|
| Notation | Definition |
| $D$ | the corpus of short texts |
| $X$ | a piece of text sequence |
| $d$ | a document |
| $w$ | a word |
| $z_k$ | a topic |
| $\theta$ | the topic distribution |
| $\phi$ | the topic-word distribution |
| $\alpha, \beta, \lambda$ | prior parameters |
| $\xi$ | the weight variable |
| $\mu$ | the word probability |
| $u$ | the word representation vector |
| $\gamma$ | the topic weight vector |
| $T$ | the topic embedding matrix |
| $V$ | the word embedding matrix |
| $G$ | the topic-word interaction matrix |
| $\sigma$ | the parameter related to word/topic embeddings |
| $\eta$ | the attention score vector |
Algorithm 1 DMM Generation Process

1: Sample a topic distribution from the Dirichlet distribution, \( \theta \sim \text{Dir}(\alpha); \)
2: for each topic \( k \in (1, 2, \ldots, K) \) do
3: \( \text{Sample a Dirichlet multinomial distribution of words, } \phi_k \sim \text{Dir}(\beta); \)
4: end for
5: for each document \( d \in (1, 2, \ldots, D) \) do
6: Sample a topic from the topic distribution, \( Z_d \sim \text{Mul(}\theta); \)
7: for each word \( w_{d,i} \) in document \( d \), \( w_{d,i} \in (w_{d,1}, w_{d,2}, \ldots, w_{d,N}) \) do
8: Sample a word from the topic-word distribution, \( w_{d,i} \sim \text{Mul(}\phi_{z_d}); \)
9: end for
10: end for

The generation process of DMM is summarized in Algorithm 1. When generating a document \( d \), DMM first generates the topic distribution for each topic \( k = 1, 2, \ldots, K \) and the topic-word distributions based on the Dirichlet distributions with parameters \( \alpha \) and \( \beta \), respectively. Then it samples a topic \( k \) based on the topic distribution with probability \( p(z = k) \), and samples words in document \( d \) based on the multinomial distribution of the sampled topic \( k \). Let \( p(d|z = k) \) denote the conditional probability of document \( d \) with respect to the topic \( k \). The generation probability \( p(d) \) of document \( d \) is the sum of the probabilities of each latent topic, as shown in (1):

\[
p(d) = \sum_{k=1}^{K} p(d|z = k)p(z = k). \tag{1}
\]

IV. THE PROPOSED FRAMEWORK

In the DMM topic model, each short text is constrained to be associated with only a single topic. While relaxing that constrain is not appropriate for noisy short texts, in this section, we propose CME-DMM, a collaboratively modeling and embedding framework for capturing latent topics from short texts. CME-DMM alleviates the sparsity of short texts by incorporating topic and word embeddings. Moreover, we employ the attention mechanism to compromise topic embeddings and word embeddings. Consequently, the learning process of latent topics favors topic-relevant words and avoids the problem of too many non-existent word co-occurrence instances in other existing methods for topic modeling. In CME-DMM, the joint distribution of topic and word embeddings and the learning process of latent topics are interrelated. Latent topic models contribute to the optimization of the topic and the word embeddings, and the topic and the word embeddings rectify the word generation process in return. With the introduction of the attention mechanism, word embeddings are more topic-sensitive, resulting in a higher quality of both topic embeddings and word embeddings.

The Bayesian network diagram of the CME-DMM model is presented in Fig. 2. CME-DMM applies the attention mechanism with a combination of topic embeddings and word embeddings based on the DMM topic model. CME-DMM introduces a decision parameter \( \lambda \) that serves as an indicator for the trade-off between different sources in the word generation process. In Fig. 2, \( V \) represents the word embedding matrix composed of the embeddings of word sequences in short texts, where each column represents a word vector. \( T \) represents the topic embedding matrix, where each column represents a topic vector. The word vector and the topic vector are of the same length. \( y \) represents the weight of a topic. As shown in Fig. 2, we calculate the attention score vector \( \eta \) based on \( V \) and \( T \), and then use the score vector \( \eta \) to weight the word sequence to form the predicted embedding vector \( u \) of the next word \( w \). Finally, we add a sample classifier to acquire the probability \( \mu \) of the observed word \( w \).

Let \( T = \{t_1, t_2, \ldots, t_K\} \) denote the topic embedding matrix, where \( K \) is the number of latent topics. Let
\( V = \{ v_{w_1}, v_{w_2}, \ldots, v_{w_l} \} \) denote the embedding matrix of word sequence, where \( l \) is the number of words in the sequence. Suppose the dimensionality of the embedding space of topics and words is \( P \), then \( T \in \mathbb{R}^{P \times K} \) and \( V \in \mathbb{R}^{P \times N} \). We calculate the attention matrix \( G \) according to (2).

\[
G = \gamma^\top \cdot T^\top \cdot V \otimes \hat{G},
\]

(2)

where \( \otimes \) is the element-wise division. \( \gamma \) is derived topic distribution of short texts, whose item is defined in (3).

\[
p(z = k|d) \sim p(z = k) \prod_{i=1}^{N_d} p(w_i|z = k).
\]

(3)

\( \hat{G} \) is a normalized \( K \times N \) matrix. Each element \( \hat{g}_{k,n} \) in \( \hat{G} \) corresponds to the \( L_2 \) norm of word embeddings and topic embeddings.

In order to capture the relative spatial information of consecutive words (e.g., phrases) in short texts, we introduce a nonlinear function \( ReLU \) in the calculation of attention scores. In particular, we consider a sequence of words whose length is \( 2r + 1 \), and the center word is the word \( w_n \). Then we use the local matrix \( G_{n-r:n+r} \) of the attention matrix \( G \) to calculate the attention score between topics and phrases by (4).

\[
s_n = ReLU(G_{n-r:n+r} \cdot W_1 + b_1),
\]

(4)

where \( s_n \in \mathbb{R}^K \), \( W_1 \in \mathbb{R}^{2r+1} \), and \( b_1 \in \mathbb{R}^K \) are parameters to be learned.

We compute the maximum attention value \( m_n \) of \( s_n \) by using the max-pooling function. The attention score is defined in (5).

\[
\eta_n = \frac{\exp(m_n)}{\sum_{n'=1}^{N_d} \exp(m_{n'})},
\]

(5)

where \( \eta \) is the attention score vector of length \( l \).

The embedding representation of the word to be generated is the weighted average of the word embeddings based on the topic attention scores, as shown in (6).

\[
u = \sum_{n=1}^{N} \eta_n v_n.
\]

(6)

We use cross-entropy to calculate the generation probability \( u \) of the next word \( w \), as shown in (7).

\[
\mu(w|V_{i-l:i-1}) = CE(X_w, f(u)),
\]

(7)

where \( CE(\cdot) \) is the cross-entropy function. The parameter \( X_w \) in \( CE(\cdot) \) is the one-hot vector representation of the word to be generated. \( u \) is the predicted vector representation of the word to be generated, and the other parameter \( f(u) \) in \( CE \) is defined as follows: \( f(u) = \text{SoftMax}(u') \), where \( u' = W_2 u + b_2 \), \( W_2 \in \mathbb{R}^{N \times P} \) and \( b_2 \in \mathbb{R}^{N} \).

In CME-DMM, we first integrate the parameter \( \theta \) and the parameter \( \phi \). Then we select the topic for each short text according to the multinomial distribution \( \text{Multi}(\theta) \). Finally, we generate words for each short text according to the multinomial distribution \( \text{Multi}(\phi) \). The generation process of the CME-DMM model is shown in Algorithm 2.

**Algorithm 2 CME-DMM Generation Process**

1: Sample a topic distribution from the Dirichlet distribution, \( \theta \sim \text{Dir}(\alpha) \);
2: \textbf{for} each topic \( k \in \{1, 2, \ldots, K\} \) \textbf{do}
3: \hspace{1em} Sample a topic-word distribution from the Dirichlet distribution, \( \phi_k \sim \text{Dir}(\beta) \);
4: \textbf{end for}
5: \textbf{for} each document \( d \in \{1, 2, \ldots, D\} \) \textbf{do}
6: \hspace{1em} Sample a topic from the topic distribution, \( Z_d \sim \text{Mul}(\theta) \);
7: \hspace{1em} \textbf{for} each word \( w \) in document \( d \) \textbf{do}
8: \hspace{2em} Generate a variable weight probability from the Bernoulli distribution, \( \xi_w \sim \text{Ber}(\lambda) \);
9: \hspace{2em} Sample a word from the topic-word distribution, \( w \sim (1 - \xi_w) \text{Mul}(\phi_{zd}) + \xi_w \mu(w|V_{i-l:i-1}) \);
10: \textbf{end for}
11: \textbf{end for}

V. MODEL PARAMETER INFERENCE

Let \( p(w|\alpha, \beta, \lambda, \sigma) \) denote the probability of the observed word \( w \) in short text \( d \), then our goal is to infer the optimal model parameters by maximizing \( p(w|\alpha, \beta, \lambda, \sigma) \). Unfortunately, it is difficult to calculate this posterior probability directly, so instead, we estimate the posterior probability \( p(w, \xi, z|\alpha, \beta, \lambda, \sigma) \).

Recall that \( p(w, \xi, z|\alpha, \beta, \lambda, \sigma) \) is the probability of the observed variables \( w, \xi, z \) with respect to the known parameters \( \alpha, \beta, \lambda, \sigma \). Since \( \alpha \) and \( \lambda \) are independent of \( \beta \) and \( \sigma \), \( p(w, \xi, z|\alpha, \beta, \lambda, \sigma) \) can be decomposed into \( p(z|\alpha)p(\xi|\lambda)p(w|z, \xi, \beta, \sigma) \).

\( p(z|\alpha) \) is the Dirichlet distribution of topics which is defined as follows:

\[
p(z|\alpha) = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \frac{\Gamma(E_k + \alpha_k)}{\Gamma(\sum_{k=1}^{K} E_k + \alpha_k)},
\]

(8)

where \( E_k \) is the number of short texts associated with the topic \( k \), and \( \Gamma(\cdot) \) denotes the Gamma function. \( p(\xi|\lambda) \) is the Bernoulli distribution which is defined as

\[
p(\xi|\lambda) = (1 - \lambda)^A \lambda^B,
\]

(9)

where \( A \) and \( B \) is the number of zeros and ones generated by the Bernoulli distribution, respectively. \( p(w|z, \xi, \beta, \sigma) \) represents the probability estimation of the observed word \( w \).
according to the parameters $\beta$, $\sigma$, $\xi$ and $z$. The definition is shown in (10).

\[
p(w|z, \xi, \beta, \sigma) = \left( \frac{\Gamma(\sum_{k=1}^{K} V^k) \prod_{k=1}^{K} \Gamma(F_k + \beta)}{\prod_{k=1}^{K} \Gamma(\sum_{v=1}^{V} V^k)} \right)^M \prod_{m=1}^{M} \prod_{w \in x_m} p(w|\sigma)^{(\xi=1)},
\]

(10)

where $F_{k,v}$ is the number of word $v$ assigned to topic $k$ by the topic-word multinomial distribution. $\Gamma(\cdot)$ denotes the Gamma function. $p(w|z, \xi, \beta, \sigma)$ is the estimation of the probability of the observed words $w$ according to the topic. In (10), the first and the second items are the estimation of the topic-word distribution in the entire corpus. The third item is the estimation of the probability of the observed word $w$ based on the topic and the word embeddings.

With (8), (9) and (10), we can rewrite $p(w, \xi, z|\alpha, \beta, \lambda, \sigma)$ as follows:

\[
p(w, \xi, z|\alpha, \beta, \lambda, \sigma) = p(z|\alpha)p(\xi|\alpha)p(w|z, \xi, \beta, \sigma) \approx \left( \frac{\Gamma(\sum_{k=1}^{K} V^k) \prod_{k=1}^{K} \Gamma(E_k + \alpha)}{\prod_{k=1}^{K} \Gamma(\sum_{v=1}^{V} E_k)} \right)^M \prod_{m=1}^{M} \prod_{w \in x_m} p(w|\sigma)^{(\xi=1)} \times (1 - \lambda)^{M} \lambda^{|F|}.
\]

(11)

By Bayesian theorem, the probability of assigning topic $k$ to short text $d$ is

\[
p(z_d = k, \xi_d|w, z_{-d}, \xi_{-d}, \alpha, \beta, \lambda, \sigma) = \frac{p(\xi_d|\xi_{-d}, \lambda)p(z_d = k|w, z_{-d}, \xi_d, \alpha, \beta, \sigma)p(z_d = k, w_d|\xi_{-d}, \xi_d, \alpha, \beta, \sigma)}{p(\xi_d|\xi_{-d}, \lambda)p(z_d = k|w, z_{-d}, \xi_d, \alpha, \beta, \sigma)} = p(\xi_d|\xi_{-d}, \lambda)p(z_d = k|z_{-d}, \xi_d, \alpha, \beta, \sigma).
\]

(12)

Here, $p(\xi_d|\xi_{-d}, \lambda)$ is the Bernoulli distribution of document $d$ inferred from other documents, which is

\[
p(\xi_d|\xi_{-d}, \lambda) = (1 - \lambda)^{k_d} \lambda^{B_d}.
\]

(13)

$p(z_d = k|z_{-d}, \alpha)$ is the Dirichlet distribution of topics composed of all documents except document $d$, which is

\[
p(z_d = k|z_{-d}, \alpha) = \frac{E^{k}_{-d} + \alpha}{\sum_{k'=1}^{K} (E^{k'}_{-d} + \alpha)},
\]

(14)

where $E^{k}_{-d}$ is the number of documents with topic $k$ in the corpus excluding document $d$. $p(w_d|w_{-d}, \xi_d, \beta, \sigma)$, which is the probability of the words in document $d$ predicted by the Dirichlet distribution and the topic and the word embeddings of other documents except document $d$, is defined as follows:

\[
p(w_d|w_{-d}, \xi_d, \beta, \sigma) = \frac{\Gamma(F^{k}_{-d} + V\beta)}{\Gamma(F^{k}_{-d} + V\beta + F^{d}_{\xi}) \prod_{w \in W_{\xi}, \xi=0} p(w|\sigma)} \frac{\Gamma(F^{k}_{-d} + V\beta + F_w)}{\Gamma(F^{k}_{-d} + V\beta + F_w)} \prod_{w \in W_{\xi}, \xi=1} p(w|\sigma).
\]

(15)

where $F^{k}_{-d}$ is the number of words with topic $k$ in the corpus without document $d$. $F^{k}_{-d}$ is the frequency of word $w$ with the topic $k$ in the corpus excluding document $d$. $F^d_{\xi}$ is the frequency of word $w$ in document $d$.

With (13), (14) and (15), we can rewrite $p(z_d = k, \xi_d|w, z_{-d}, \xi_{-d}, \alpha, \beta, \lambda, \sigma)$ as follows:

\[
p(z_d = k, \xi_d|w, z_{-d}, \xi_{-d}, \alpha, \beta, \lambda, \sigma) = (1 - \lambda)^{k_d} \lambda^{B_d} \frac{E^{k}_{-d} + \alpha}{\sum_{k'=1}^{K} (E^{k'}_{-d} + \alpha)} \frac{\Gamma(F^{k}_{-d} + V\beta + F_{\xi})}{\Gamma(F^{k}_{-d} + V\beta + F_{\xi})} \prod_{w \in W_{\xi}, \xi=0} p(w|\sigma) \prod_{w \in W_{\xi}, \xi=1} p(w|\sigma).
\]

(16)

The number of possible combinations of $\xi_d$ is exponential, so we approximate (16) with the integration of $\xi_d$ as shown in (17), which is similar to the approach in [10].

\[
p(z_d = k|w, z_{-d}, \xi_{-d}, \alpha, \beta, \lambda, \sigma) \sim \frac{E^{k}_{-d} + \alpha}{\sum_{k'=1}^{K} (E^{k'}_{-d} + \alpha)} \prod_{w \in W} \left( 1 - \lambda \right) \frac{F^{k}_{-d} + \beta + \lambda p(w|\sigma)}{F^{k}_{-d} + V\beta + F_{\xi}}.
\]

(17)

The fraction part in (17) is the Dirichlet distribution of topics, and the remaining part represents two branches in the word generation process. One generates words based on the topic-word distributions, and the other one generates words based on the topic and the word embeddings.

For each word $w$ in short text $d$, we sample the potential indicator variable $\xi_w$ based on (18) and (19), respectively.

\[
p(\xi_w = 0|z_d = k) = (1 - \lambda) \frac{F^{k}_{-d} + \beta}{F^{k}_{-d} + V\beta},
\]

(18)

\[
p(\xi_w = 1|z_d = k) = \lambda p(w|\sigma).
\]

(19)

The above sampling process is executed repeatedly until it converges.

Now we will explain the optimization process of the topic and the word embeddings. For word $w$ in short text $d$ during the word generation process, we predict the possible word $w_i$ based on the word sequence $\{w_{i-1}, w_{i-2}, \ldots, w_{i-1}\}$. Therefore, the objective function is formulated as follows:

\[
\max \sum_{d=1}^{D} \sum_{i=1}^{N} p(w_{d, i}|V_{w_{d, i-1}, T}).
\]

(20)
Algorithm 3 The Stochastic EM Learning for CME-DMM

Input: a corpus of short texts;
1: Initialize the topic embeddings \( T \) and the word embeddings \( V \) with pre-trained word embeddings and uniform distribution, respectively;
2: Randomly assign a topic to each short text;
3: repeat
4: E-Step:
5: for each short text \( d \in \{1, 2, \ldots, D\} \) do
6: for each word \( w \) in short text \( d \) do
7: Obtain the \( p(w|\sigma) \) from the topic and the word embeddings in a forward pass;
8: end for
9: Sample the topic \( z_d \) according to (17);
10: end for
11: M-Step:
12: Collect sufficient statistics to obtain: \( \phi_{kw} = \frac{F_{k,w} + \beta}{F_{f,k} + \gamma} \);
13: Obtain the topic weights for each short text according to (3);
14: Optimize the topic and the word embedding parameter \( \sigma \) by stochastic gradient descent methods;
15: until convergence

We use cross-entropy to measure the difference between two distributions, so the above objective function is equivalent to minimize the following loss function:

\[
\min_{\theta} \sum_{m=1}^{M} \sum_{n=1}^{N} CE(X_{m,n}, f(u_{m,n})),
\]

where \( CE(.) \) is the cross-entropy function. The parameter \( X_{m,n} \) in \( CE \) is the one-hot vector representation of the \( n \)-th word in document \( m \). The other parameter \( f(u_{m,n}) \) in \( CE \) is defined as follows: \( f(u_{m,n}) = \text{SoftMax}(u_{m,n}) \), where \( u_{m,n} \) is the vector representation of the \( n \)-th word in document \( m \) predicted based on the topic and the word embeddings.

We design a Stochastic EM algorithm for learning the parameters in CME-DMM, as shown in Algorithm 3. In Algorithm 3, the topic and the word embeddings are initialized before learning. In this paper, we initialize the word embedding matrix with pre-trained word embeddings. The optimization process will further tune these word embeddings during learning. For topic embeddings, as well as those words not in the vocabulary, we use uniform distribution to initialize them. At line 2, a random topic is assigned for each short text. At line 7, Algorithm 3 derives the joint probability \( p(w|\sigma) \) from the topic and the word embeddings to generate word \( w \). At line 9, a topic is sampled for short text \( d \) according to (17). At line 14, Algorithm 3 optimizes the topic embeddings and word embeddings according to [41]. For a very large corpus, one can extend it into a parallel version based on distributed stochastic gradient descent in [42].

VI. EXPERIMENTAL EVALUATIONS

In this section, we report our experimental results and use both subjective and objective measures to evaluate the quality of the latent topics from short texts.

A. DATA SETS

The corpus of short texts used in the experiments for latent topic modeling contains 679,823 pieces of short texts, which are collected from Weibo.\(^1\) These short texts are in Chinese, and the length of them varies from 100 to 200. Since texts in Chinese are consecutive without space, we use a Chinese word segmentation tool called JieBa\(^2\) for text segmentation. The stop words and the punctuations in these texts are removed. We employ a news data set collected by Sogou Labs,\(^3\) which contains 3,897,400 new articles from 20 different fields, to generate the pre-trained word embeddings. In this paper, we use Google translate to translate them into English for better presentation and understanding.

B. BASELINE METHODS

We compare the proposed CME-DMM with the following approaches.

- **LDA** [1]: Latent Dirichlet Allocation is the classic topic model for general texts.
- **DMM** [33]: Dirichlet Multinomial Mixtures is the popular topic model for short texts, in which each short text is associated with a single topic.
- **LF-LDA** [10]: LF-LDA is a variation of LDA by introducing external word embeddings to complement the relationship between words.
- **LF-DMM** [10]: LF-DMM is a variation of DMM by introducing external word embeddings to complement the relationships between words.
- **Word-LLA** [11]: Word-LLA is a combination of LDA and LSTM (Long Short-Term Memory), an artificial recurrent neural network. LSTM is employed to predict document topics based on word sequences.
- **Topic-LLA** [11]: Topic-LLA model is similar to Word-LLA, where the difference between them is that Topic-LLA predicts topics based on topic sequences instead of word sequences.

Some papers in Section II do not reveal their algorithms in detail, which makes the implementation very difficult. Some topic models are based on bi-gram distributions, while almost all other topic models are in the form of topic-word distributions. Due to the above issues, we have to exclude these approaches from our experimental study. The comparison between topic models with various model structures is a potential future direction.

C. TOPIC COHERENCE EVALUATION

In this section, we compare the quality of latent topics discovered by various approaches. We use Pointwise Mutual

\(^1\) Weibo is a Twitter-like social network in China.
\(^2\) http://github.com/fsxjy/jieba
\(^3\) http://www.sogou.com/labs/
Information (PMI) \[43\] as the measure of topic coherence, which has proven to be an effective topic quality measure \[44\]. Given a topic \(k\) and its top \(T\) words \(W^k = (w^k_1, w^k_2, \ldots, w^k_T)\), which are the \(T\) words with the highest probabilities. Let \(p(w)\) denote the document frequency of word \(w\), and \(p(w_t, w_t')\) denote the document frequency in which word \(w_t\) and \(w_t'\) appear. The PMI score for topic \(k\) is defined as follows:

\[
PMI(k, W^k) = \sum_{i=1}^{T} \sum_{j=1}^{T} \log \left( \frac{p(w^k_i, w^k_j) + \epsilon}{p(w^k_i)p(w^k_j)} \right).
\] (22)

Higher PMI scores indicate more coherent latent topics. In the following experiments, the prior parameter \(\alpha\) is 0.1, and \(\beta\) is 0.01. The hyperparameter \(\lambda\) is 0.5. The number of training iterations is 2000, and the length of the context text window is 30.

Fig. 3-6 show the PMI scores of latent topics. We present the average PMI scores of top \(N\) \((N = 5, 10, 15, 20)\) words from each topic. As we can see, CME-DMM achieves the best results among all these models. The introduction of the topic and the word embeddings can significantly improve the topic quality compared with other models without them. Recall that \(k\) is the number of latent topics in the corpus. When \(k\) and \(N\) increase, the PMI scores also increase, resulting in improved coherence in latent topics.

CME-DMM has a relatively small variance when \(k\) varies, which shows CME-DMM is more robust and stable than other topic models.

### D. TEXT CLASSIFICATION

In this section, we present the performance of the proposed model in a downstream NLP task, i.e., text classification. With better classification results, we demonstrate that the proposed model can extract discriminative latent topics and achieve high representation ability.

We used a small data set containing 3506 tweets in Chinese labeled in one of the four classes: Sports, Campus, Literature, Female. All tweets in the data set without the label information are used for learning the topic/word embeddings and the topic model. The classifier is Support Vector Machine (SVM), and each tweet is represented by its topic distribution calculated by (3) in our paper. We conducted 10-fold cross-validation and reported the average classification performance.

The topic distribution \(p(z|d)\) for a piece of short text can serve as its representation, which could be derived using (3) based on \(p(w|z)\) and \(p(z)\). Therefore, a general classifier could classify short texts by using these document representations as the input features. The pros and cons of different topic models could be assessed based on the performance of the
TABLE 2. The performance of various topic models in text classification.

| Model  | Micro-P |  |  |  | Micro-R |  |  |  | Micro-F1 |  |  |  |
|--------|---------|---|---|---|---------|---|---|---|----------|---|---|---|
|        | k=10    | k=20 | k=30 | k=40 |         | k=10 | k=20 | k=30 | k=40 |         | k=10 | k=20 | k=30 | k=40 |
| LDA    | 0.516   | 0.567 | 0.507 | 0.584 | 0.580   | 0.615 | 0.545 | 0.600 | 0.543   | 0.587 | 0.522 | 0.574 |
| DMM    | 0.782   | 0.854 | 0.811 | 0.793 | 0.775   | 0.840 | 0.805 | 0.805 | 0.777   | 0.839 | 0.801 | 0.795 |
| LFDMM  | 0.669   | 0.809 | 0.701 | 0.849 | 0.650   | 0.815 | 0.765 | 0.825 | 0.650   | 0.797 | 0.731 | 0.811 |
| LF-LDA | 0.845   | 0.733 | 0.823 | 0.801 | 0.835   | 0.800 | 0.820 | 0.790 | 0.839   | 0.764 | 0.796 | 0.757 |
| Word-LLA | 0.533 | 0.555 | 0.571 | 0.583 | 0.525   | 0.570 | 0.560 | 0.600 | 0.518   | 0.560 | 0.558 | 0.586 |
| Topic-LLA | 0.709 | 0.685 | 0.724 | 0.720 | 0.700   | 0.685 | 0.705 | 0.735 | 0.698   | 0.681 | 0.710 | 0.721 |
| CME-DMM | 0.884   | 0.859 | 0.896 | 0.890 | 0.835   | 0.835 | 0.855 | 0.870 | 0.844   | 0.841 | 0.863 | 0.872 |

classifier. Better classification results mean higher representability of topic models.

We use the most popular Micro-F1 to evaluate the performance of the classifier with various topic models. The micro precision (Micro-P) and the micro recall (Micro-R) are defined in (23) and (24), respectively. Based on them, Micro F1 is defined in (25).

\[
\text{Micro-P} = \sum_{i=1}^{n} \frac{TP_i}{(TP_i + FP_i)}, \quad (23)
\]

\[
\text{Micro-R} = \sum_{i=1}^{n} \frac{TP_i}{(TP_i + FN_i)}, \quad (24)
\]

\[
\text{Micro-F1} = \frac{2 \times \text{Micro-P} \times \text{Micro-R}}{\text{Micro-P} + \text{Micro-R}}. \quad (25)
\]

In (23) and (24), TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives.

The classification performance of various models is presented in Table 2. The classification based on the LDA model is not satisfactory, while the classification based on the DMM model is much better. With the help of word embeddings, the classifiers based on LF-LDA and LF-DMM perform better than ones based on LDA and DMM. It shows that word embeddings do supplement the semantic information to short texts. CME-DMM outperforms all other topic models in Micro-P, Micro-R, and Micro-F1, which demonstrates that the collaboratively embedding and modeling of latent topics is beneficial to the quality of latent topics. The classification performance of CME-DMM increases with the number of latent topics, indicating that the influence of the topic and the word embeddings are enhanced.

E. REPRESENTATIONAL CHARACTERISTICS

Fig. 7 presents three cumulative probabilities of top \( N \) words, which could be a guide to the selection of the number of words characterizing each topic. The words in each topic are ordered by the corresponding topic-word distribution. The maximum/minimum/average curve is the cumulative probabilities of the maximum/minimum/average word probabilities among all topics. Based on the observation, we could select around 500 words for representing each topic.

In CME-DMM, topic embeddings and word embeddings are in the same latent semantic space. We visualize the topic and the word embeddings on a 2D map by using t-SNE [45], as shown in Fig. 8. There are ten topics and 500 words from each topic. Small dots indicate word embeddings, and large dots with black circles indicate topic embeddings. Different colors mean different topics. As we can see, the five topics on the lower left side are relatively good and surrounded by words of the same topic with high probabilities, which proves the validity of our proposed topic embeddings. While the topics on the upper right side look like a little bit messy, due to the following two reasons: (1) Different topics share some common words, which affects the visualization. (2) Fig. 8 only presents ten topics. For a large corpus of short texts, it is difficult for a small number of topics to be well separated.

We assessed the quality of topic embeddings by calculating the cosine similarity between a topic embedding and the word embeddings of the top \( N \) words belonging to the same topic. Fig. 9 shows their consistency. The horizontal axis is the sorted topic index, and the vertical axis is the
average cosine similarity. A high-quality topic should have considerable cosine similarity with its words, which means the semantic meaning of a topic should be consistent with the semantic meanings of words on the same topic.

We introduced the attention mechanism in CME-DMM. As a useful by-product, we can identify the keywords in short texts based on the attention scores. Fig. 10 and Fig. 11 present two examples from the data set. We mark the corresponding keywords in dark yellow. The text sequences in Fig. 10 and Fig. 11 are the topics related to ‘sports’ and ‘campus’, respectively, which show that keywords are correctly detected based on their attention scores.

In Fig. 12, we present side-by-side the words from the topic-word distribution and the words similar to the topic embeddings. We adopt the word cloud as the presentation style, where a larger word size means larger probability or similarity. For each topic, we show the top 10 words with the highest probabilities in Column (a) and the top 10 words in Column (b), which are the most similar words to the topic based on their embeddings. We can see that the words from different sources are not the same, but they correlate with each other closely in their semantic meanings.

VII. CONCLUSIONS

In this paper, we have proposed a collaboratively modeling and embedding framework, CME-DMM, to capture coherent latent topics for better document representation and performance in downstream tasks. Topic embeddings and word embeddings accommodate each other through the attention mechanism and rectify the word generation probabilities in the learning process of latent topics. In return, the topic and the word embeddings are fine-tuned according to the latent topic distributions iteratively. The results from extensive experiments show that CME-DMM can significantly enhance the coherence and quality of latent topics, and enrich the representation characteristics of the topic and the word embeddings.

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] S. T. Dumais, “Latent semantic analysis,” Annu. Rev. Inf. Sci. Technol., vol. 38, no. 1, pp. 188–230, 2004.
[3] N. Naveed, T. Gottron, J. Kunegis, and A. C. Alhadi, “Searching microblogs: Coping with sparsity and document quality,” in Proc. 20th ACM Int. Conf. Inf. Knowl. Manage. (CIKM), 2011, pp. 183–188.
[4] J. Weng, E.-P. Lim, J. Jiang, and Q. He, “TwitterRank: Finding topic-sensitive influential twitterers,” in Proc. 3rd ACM Int. Conf. Web Search Data Mining (WSDM), 2010, pp. 261–270.
[5] L. Hong and B. D. Davison, “Empirical study of topic modeling in Twitter,” in Proc. 1st Workshop Social Media Analytics (SOMA), 2010, pp. 80–88.
[6] Z. Liu, C. Liu, B. Xia, and T. Li, “Multiple relational topic modeling for noisy short texts,” Int. J. Softw. Eng. Knowl. Eng., vol. 28, nos. 11–12, pp. 1559–1574, Nov. 2018.
[7] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, “Text classification from labeled and unlabeled documents using EM,” Mach. Learn., vol. 39, nos. 2–3, pp. 103–134, 2000.
[8] D. Jiang, L. Shi, R. Lian, and H. Wu, “Latent topic embedding,” in Proc. 7th Int. Conf. Comput. Linguistics, Tech. Papers (COLING), 2016, pp. 2689–2698.
[9] 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.
[10] D. Q. Nguyen, R. Billingsley, L. Du, and M. Johnson, “Improving topic models with latent feature word representations,” Trans. Assoc. Comput. Linguistics, vol. 3, pp. 299–313, Dec. 2015.
[11] M. Zaheer, A. Ahmed, and A. J. Smola, “Latent LSTM allocation joint clustering and non-linear dynamic modeling of sequential data,” in Proc. 34th Int. Conf. Mach. Learn., vol. 70, 2017, pp. 3967–3976.
[12] D. Blei, L. Carin, and D. Dunson, “Probabilistic topic models,” IEEE Signal Process. Mag., pp. 55–65, Nov. 2010.
[13] Q. Jipeng, Q. Zhenyu, L. Yun, Y. Yunhao, and W. Xindong, “Short text topic modeling techniques, applications, and performance: A survey,” 2019, arXiv:1904.07695. [Online]. Available: http://arxiv.org/abs/1904.07695
[14] I. Vulic, W. De Smet, and M.-F. Moens, “Identifying word translations from comparable corpora using latent topic models,” in Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol., Portland, OR, USA: Association for Computational Linguistics, Jun. 2011, pp. 479–484. [Online]. Available: https://www.aclweb.org/anthology/P11-2084
[15] V. Eidelman, J. Boyd-Graber, and P. Resnik, “Topic models for dynamic translation model adaptation,” in Proc. 50th Annu. Meeting Assoc. Comput. Linguistics, vol. 2. Jeju Island, South Korea: Association for Computational Linguistics, Jul. 2012, pp. 115–119. [Online]. Available: https://www.aclweb.org/anthology/P12-2023
[16] I. Heintz, R. Gabbard, M. Srivastava, D. Barner, D. Black, M. Friedman, and R. Weisschedel, “Automatic extraction of linguistic metaphors with LDA topic modeling,” in Proc. 1st Workshop Metaphor (NLP), Atlanta, GA, USA: Association for Computational Linguistics, Jun. 2013, pp. 58–66. [Online]. Available: https://www.aclweb.org/anthology/W13-0008
[17] M. Lai, J. H. Lau, and T. Baldwin, “Automatic detection and language identification of multilingual documents,” Trans. Assoc. Comput. Linguistics, vol. 2, pp. 27–40, Dec. 2014. https://www.aclweb.org/anthology/Q14-1003
[18] L. Hong, O. Dan, and B. D. Davison, “Predicting popular messages in Twitter,” in Proc. 20th Int. Conf. Companion World Wide Web (WWW), New York, NY, USA: Association for Computing Machinery, 2011, pp. 57–58, doi: 10.1145/1963392.1963222.
[19] L. Shi, J. Du, M. Liang, and F. Kou, “Dynamic topic modeling via self-aggregation for short text streams,” Peer-to-Peer Netw. Appl., vol. 12, no. 5, pp. 1403–1417, Sep. 2019, doi: 10.1007/s12083-018-00692-7.
[20] E. Linstead, P. Rigor, S. Bajracharya, C. Lopes, and P. Baldi, “Topic models for dynamic coupling among classes in object-oriented software systems,” in Proc. IEEE Int. Conf. Softw. Maintenance, Oct. 2010, pp. 1–10.
[21] W. Gao, M. Peng, H. Wang, Y. Zhang, W. Han, G. Hu, and Q. Xie, “Generation of topic evolution graphs from short text streams,” Neurocomputing, vol. 383, pp. 282–294, Mar. 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231219316807
[22] D. Greene and J. P. Cross, “Unveiling the political agenda of the European parliament plenary: A topical analysis,” in Proc. ACM Web Sci. Conf., vol. 12, no. 5, 2013, pp. 281–293, Sep. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877750317305318
[23] C. Xiao, P. Zhang, W. Chaowalitwongse, J. Hu, and F. Wang, (2017). Adverse Drug Reaction Prediction With Symbolic Latent Dirichlet Allocation. [Online]. Available: https://www.journals.elsevier.com/future-generation-computing-systems/
[24] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang, “Geographical topic discovery and comparison,” in Proc. 20th Int. Conf. World Wide Web, Jan. 2011, pp. 247–256.
[25] H. Tang, L. Shen, Y. Qi, Y. Chen, Y. Shu, J. Li, and D. A. Clausi, “A multiscale latent Dirichlet allocation model for object-oriented clustering of VHR panchromatic satellite images,” IEEE Trans. Geosci. Remote Sens., vol. 51, no. 3, pp. 1680–1692, Mar. 2013.
[26] X. Yan, J. Guo, Y. Lan, and X. Cheng, “A bitemporal topic model for short texts,” in Proc. 22nd Int. Conf. World Wide Web (WWW), 2013, pp. 1445–1456.
[27] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, “Automatic evaluation of topic coherence,” in Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, vol. 2. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 380–386. [Online]. Available: https://www.aclweb.org/anthology/P16-2062
[28] B. Shi, W. Lam, S. Jameel, S. Schockaert, and K. P. Lai, “Jointly learning word embeddings and latent topics,” in Proc. 40th Int. Conf. SIGIR Conf. Res. Develop. Inf. Retr. New York, NY, USA: Association for Computing Machinery, 2017, pp. 75–84, doi: 10.1145/3071363.3080806.
[29] W. Gao, M. Peng, H. Wang, Y. Zhang, Q. Xie, and G. Tian, “Incorporating word embeddings into topic modeling of short text,” Knowl. Inf. Syst., vol. 61, no. 2, pp. 1123–1145, Nov. 2019, doi: 10.1007/s10115-018-1314-7.
[30] S. Li, T.-S. Chua, J. Zhu, and C. Miao, “Generative topic embedding: A continuous representation of documents,” in Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, vol. 1. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 666–675. [Online]. Available: https://www.aclweb.org/anthology/P16-1063
[31] S. Li, Y. Zhang, R. Pan, M. Mao, and Y. Yang. (2017). Recurrent Attentional Topic Model. [Online]. Available: https://aaai.org/ojs/index.php/AAAI/article/view/14400
[32] G. J. McLachlan and K. E. Basford, Mixture Models: Inference and Applications to Clustering, New York, NY, USA: M. Dekker, 1988.
[33] J. Yu and L. Qiu, “ULW-DMM: An effective topic modeling method for microblog short text,” IEEE Access, vol. 7, pp. 884–893, 2019.
[34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980. [Online]. Available: https://arxiv.org/abs/1412.6980
[35] X. Shi, Q. He, X. Luo, Y. Bai, and M. Shang, “Large-scale and scalable latent factor analysis via distributed alternative stochastic gradient descent for recommender systems,” IEEE Trans. Big Data, early access, Feb. 11, 2020, doi: 10.1109/TBDATA.2020.2973141.
[36] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, “Automatic evaluation of topic coherence,” in Proc. Hum. Lang. Technol., Ann. Conf. North Amer. Chapter Assoc. Comput. Linguistics, 2010, pp. 100–108.
[37] A. Fang, C. Macdonald, I. Ounis, and P. Habel, “Topics in tweets: A user study of topic coherence metrics for twitter data,” in Proc. Eur. Conf. Inf. Retr. Cham, Switzerland: Springer, 2016, pp. 492–504.
[38] L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, pp. 2579–2605, Nov. 2008
TINGTING QIN is currently pursuing the master’s degree with the Nanjing University of Posts and Telecommunications, China. Her current research interests include latent topic models and comparative analysis of patents.

KE-JIA CHEN (Member, IEEE) received the master’s degree from the LAMDA Group, Nanjing University, and the Ph.D. degree from the Université de Technologie de Compiègne, France. She joined the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, China, in 2017. She is currently an Associate Professor at the Nanjing University of Posts and Telecommunications, China. Her current research interests include data mining and machine learning with applications in complex network analysis. She once published articles in TOIS, ASONAM, ICDM, DASFAA, PAKDD, and ECML. She also serves as a reviewer for international journals, like Information Systems and Neural Networks.

YUN LI (Member, IEEE) received the Ph.D. degree in computer science from Chongqing University, Chongqing, China. He was a Postdoctoral Fellow with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. He is currently a Professor at the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China. He has published more than 60 refereed research articles. His current research interests include machine learning, data mining, and parallel computing.

* * *