Knowledge and Information Systems (2019) 61:1123–1145
https://doi.org/10.1007/s10115-018-1314-7

REGULAR PAPER

Incorporating word embeddings into topic modeling of short text

Wang Gao[1] · Min Peng[1] · Hua Wang[2] · Yanchun Zhang[2] · Qianqian Xie[1] · Gang Tian[1]

Received: 10 September 2017 / Revised: 26 July 2018 / Accepted: 28 November 2018 /
Published online: 18 December 2018

Abstract
Short texts have become the prevalent format of information on the Internet. Inferring the topics of this type of messages becomes a critical and challenging task for many applications. Due to the length of short texts, conventional topic models (e.g., latent Dirichlet allocation and its variants) suffer from the severe data sparsity problem which makes topic modeling of short texts difficult and unreliable. Recently, word embeddings have been proved effective to capture semantic and syntactic information about words, which can be used to induce similarity measures and semantic correlations among words. Enlightened by this, in this paper, we design a novel model for short text topic modeling, referred as Conditional Random Field regularized Topic Model (CRFTM). CRFTM not only develops a generalized solution to alleviate the sparsity problem by aggregating short texts into pseudo-documents, but also leverages a Conditional Random Field regularized model that encourages semantically related words to share the same topic assignment. Experimental results on two real-world datasets show that our method can extract more coherent topics, and significantly outperform state-of-the-art baselines on several evaluation metrics.

Keywords Short text · Topic model · Word embeddings · Conditional Random Fields

1 Introduction

With the rapid development of the Internet and social media services such as Twitter and Facebook, web users and applications are generating more and more short texts, including news headlines, questions/answers, instant messages, tweets, product reviews, text advertisements and so on. Given the huge volume of short texts available, effective topic models to extract the hidden thematic structure from short texts become fundamental to many tasks

✉ Min Peng
pengm@whu.edu.cn

✉ Gang Tian
tiang2008@whu.edu.cn

1 School of Computer Science, Wuhan University, Wuhan, China

2 Centre for Applied Informatics, Victoria University, Melbourne, Australia

that require semantic understanding of textual content, such as short text classification [1], short text clustering [30], text compression [31], emerging topic tracking [13] and sentiment analysis [15].

Traditional topic modeling algorithms such as Probabilistic Latent Semantic Analysis (PLSA) [9] and Latent Dirichlet Allocation (LDA) [3] have been widely used to automatically discover the hidden topics from large archive of documents. These models view documents as mixtures of probabilistic topics, where each topic is a probability distribution over words. Essentially, the topic models capture word co-occurrence information and these highly co-occurring words are put together to compose a topic [27]. Therefore, the key to reveal high-quality topics is that the corpus must contain a large number of word co-occurrence patterns. However, conventional topic models have achieved great successes on long documents, but they work poorly on short texts. There are two main reasons: (1) only very limited word co-occurrence information is available in short texts compared with long documents such as news articles and academic papers and (2) it is more difficult for topic models to identify the senses of ambiguous words because of the limited contexts in short texts [5].

Several heuristic strategies have been adopted to tackle the data sparsity problem in short texts. One straightforward strategy follows the relaxed assumption that each short text belongs to only one topic, e.g., unigrams or Dirichlet Multinomial Mixture (DMM) model [39]. This simplification is unsuited to long documents, but it may be feasible for certain collections of short texts and help to alleviate the sparsity problem [40]. However, the assumption is not always applicable and these models cannot directly solve the problem of limited word co-occurrence information in short texts. The other strategy is to aggregate short texts into long pseudo-documents and then standard topic models are applied to infer the topics in these pseudo-documents [10,23,35]. Nevertheless, this strategy is highly data-dependent, which makes it difficult to be generalized to cope with more general forms such as news titles and questions/answers. Figure 1 illustrates an example to explain the weaknesses of existing strategies. As shown in the figure, s_1 and s_0 are likely to contain two topics. "*President*" and "*Trump*" tend to be related to the same topics, while "*football*" and "*baseball*" probably come from another topic. Thus, the assumption that each text only covers a single topic is unsuitable for these short texts. In addition, many traditional similarity metrics heavily rely on the co-occurrence of words between two documents. It is difficult to aggregate these three short texts into two pseudo-documents since they have no words in common. However, it is obvious that short text s_0 is much more close to s_1 than s_2.

Recently, many researches utilize latent word embeddings [26] to measure word similarities. These methods give semantically related words a better chance to share the same topic label [19,37]. Although these works have proven that vector representations are capable of helping improving topic models, they ignore most word embedding methods assume that

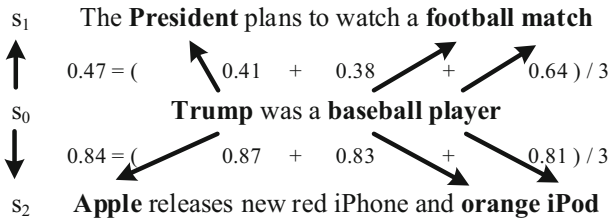


Fig. 1 An illustration of embedding-based minimum average distance. The distance between the two short texts is computed by averaging the shortest distances (arrows) between two words from different short texts

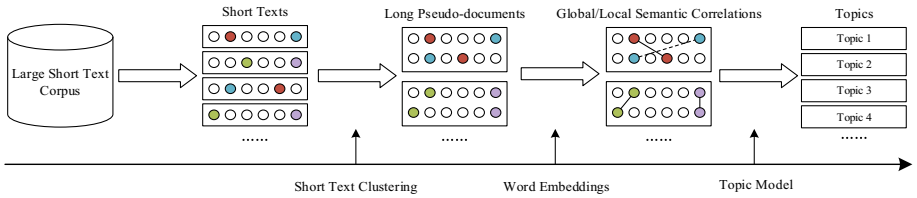


Fig. 2 The framework of Conditional Random Field regularized Topic Model of short texts

each word has a specific meaning and represent each word with a single vector [11], which restricts their applications in fields with polysemous and homonymous words. For instance, "*Apple*" can be either "*a fruit*" or "*an IT company*". As shown in Fig. 1, the word "*Apple*" has similar representativeness with "*orange*" in short text s_2. However, they will be unreasonably assigned to the same topic.

In this paper, we propose a novel topic model of short texts to address the above challenges. The main idea comes from the answers of the following two questions: (1) How to find a generalized solution for aggregation of short texts against data sparsity? (2) How to differentiate the subtleties of word sense across topics when exploiting word embeddings to improve topic modeling of short texts?

Specifically, we design a Conditional Random Field regularized Topic Model (CRFTM). As shown in Fig. 2, the proposed model adopts a two-phase framework to address both sparsity and word sense disambiguation issues in topic modeling over short texts. In the first phase, CRFTM aggregates short texts into long pseudo-documents by a new metric for the distance between short texts. The new metric, which we call the embedding-based minimum average distance (EMAD), is able to directly capture semantically related word pairs in two short texts. These word pairs are more probable to belong to the same topic. Figure 1 shows a schematic illustration of our new metric (see Sect. 3.1 for more details). Through the effective aggregation of short texts with similar topics, more useful word co-occurrences can be created, leading to a generalized solution that potentially alleviates the sparsity problem. In the second phase, we define a Conditional Random Field (CRF) on the latent topic layer of LDA to enhance the coherence of topic modeling. Our model defines two types of correlations: (1) Global semantic correlations are used for encouraging related words to share the same topic label, which can improve the coherence of learned topics; (2) CRFTM leverages local semantic correlations to effectively identify the senses of ambiguous words, hence the irrelevant words can be filtered. Both correlations are modeled explicitly using different sets of potential functions. To measure the performance of CRFTM, we conduct extensive experiments on two real-world short text datasets. Experimental results show that CRFTM discovers more coherent topics and achieves better classification accuracy than state-of-the-art baselines. The main contributions of this paper are summarized as follows:

1. We introduce a new metric for the distance between short texts, to aggregate short texts into pseudo-documents before the application of topic modeling. The aggregation directly captures the topical similarity between individual word pairs, which can be well generalized to more general genres of short texts.
2. We propose a novel topic model to discover the latent topics from short texts. The model utilizes both global semantic correlations and local semantic correlations to extract more meaningful topics. To the best of our knowledge, this is the first work to integrate word correlation knowledge based on word embeddings into a topic model with CRF model.

3. The performance of the proposed model is evaluated on two real-world short text collections against a few state-of-the-art methods. Experimental results demonstrate our model outperforms the baseline models on several evaluation metrics.

The rest of the paper is organized as follows. Section 2 reviews related work on topic models for short texts and topic models with word embeddings. Section 3 details our model for short text topic modeling. We report the datasets and experimental results in Sect. 4 and conclude the paper in Sect. 5.

2 Related work

In this section, we briefly summarize the related work from the following two perspectives: topic models on short texts and topic models with word embeddings.

Topic models on short texts Data sparseness has long been the bottleneck of topic modeling over short texts. One intuitive way is to make use of external knowledge to enrich the representation of short texts. For example, Phan et al. proposed a method for discovering hidden topics using Wikipedia as an external corpus to deal with the data sparsity problem in short text classification [33]. Jin et al. [14] proposed the Dual-LDA model that uses not only the short texts but also their related long texts to generate topics. In practice, however, such auxiliary data may not be always available or just too costly for collection.

Because of the length of short texts, few term co-occurrence information prevents conventional topic models from superior topic inferences. A straightforward approach to increase the word co-occurrence information per document is to merge short texts into lengthy pseudo-documents. Several studies have shown that by training a topic model on aggregated messages we can obtain a higher quality of learned model [10,35]. Weng et al. have created a pseudo-document from the collection of a user's tweets, and then the standard LDA model is applied to infer latent topics [35]. Similarly, Mehrotra et al. [23] show that hashtag-based short text aggregation leads to drastically improved topic modeling on Twitter content. However, the above strategies cannot be generalized to tackle more general forms of short texts which hardly contain any useful context information.

Many efforts have been spent toward designing customized topic models for short texts. The biterm topic model (BTM) proposed by Cheng et al. [5] directly models word co-occurrence patterns (i.e., biterms) extracted from short texts. Their experimental results show BTM learns more coherent topics than LDA. Inspired by the aforementioned aggregation strategies, the self aggregation methods without using auxiliary information become an emerging solution for providing additional useful word co-occurrence information. Zuo et al. [41] proposed a pseudo-document-based topic model (PTM) for short text topic modeling, which leverages much less pseudo-documents to self-aggregate tremendous short texts. However, the inference process of PTM involving both topic sampling and text aggregation is complicated and very time-consuming. In contrast, the current work aggregates short texts into long pseudo-documents only once in the entire process.

Topic models with word embeddings Word embeddings are distributed representations of words and contain some semantic and syntactic information [26], which have been found useful for many natural language processing (NLP) tasks, including part-of-speech tagging, named entity recognition, topical coding and parsing [2,12,32].

Das et al. [6] proposed Gaussian LDA which models each topic as a Gaussian distribution over the word embedding space. Nguyen et al. [29] proposed topic models that incorporate

the information of word embeddings in inferring topic-word distributions. Xie et al. [37] incorporated word correlation knowledge into LDA by building a Markov random field regularization model, which takes advantage of an existing pre-trained word embedding. If the similarity between the embeddings of two words in a document exceeds a threshold, they generate a must-link between the two words. Based on the DMM model, Li et al. [19] proposed a topic model which promotes the semantically related words under the same topic during the sampling process by using the Generalized Pólya Urn (GPU) model [22].

These methods measure the semantic relatedness between two words by their word embeddings, which neglect the fact that each word is represented as a single vector in most word embedding methods. Nevertheless, polysemous and homonymous words have multiple senses in practical use. It is not appropriate to exploit the similarities among these ambiguous words based on word embeddings to improve topic modeling.

Unlike these researches, CRFTM takes into consideration both global and local semantic correlation knowledge provided by word embeddings by imposing a CRF in the topic inference process. Compared with existing solutions which incorporate word embeddings into topic modeling, CRFTM reduces the noise of the topic inference process caused by ambiguous words. To the best of our knowledge, this work is the first to combine word embeddings and CRF model for solving both sparsity and word sense disambiguation during the topic inference of short texts.

3 CRFTM

In this section, we first discuss how to cluster short texts into regular-sized pseudo-documents which can be easily applied to ordinary forms of short texts, and then discuss how to incorporate the word correlation knowledge with word embeddings to improve the coherence of topic modeling.

3.1 Short text clustering

As short texts cannot provide sufficient word co-occurrence or context shared information, traditional text clustering methods may fail to achieve satisfactory results when they are directly applied to short text tasks [25]. Recently, Kusner et al. proposed a simple method to compute the cumulative cost that words from one text need to travel to match exactly the words of the other text as the distance of texts [16]. Inspired by this approach, we devise a novel method to improve the performance of short text clustering. First, we present a new metric, called the Embedding-based Minimum Average Distance (EMAD), to measure the distance between short texts. Secondly, we implement a k-medoids clustering algorithm with a constraint that each cluster has the same number of short texts to alleviate the data sparsity problem.

Due to the length of short texts, the words with high semantic correlations may not frequently co-occur in the same short texts. Our goal is to integrate the topic similarity between semantically related word pairs (e.g., "*President*" and "*Trump*") into the short text distance metric. More useful word co-occurrences can be created through our effective aggregation of short texts with similar topics.

Given pre-trained word embeddings of each word, we measure the distance between words by the cosine distance between their vector representations. Let $\mathbf{w}_a \in \mathbb{R}^o$ be the word vector

corresponding to the a_{th} word in the o-dimensional space. We define the distance between $\mathbf{w}_a$ and $\mathbf{w}_b$ as

$$d(\mathbf{w}_a, \mathbf{w}_b) = 1 - \frac{\mathbf{w}_a \cdot \mathbf{w}_b}{\| \mathbf{w}_a \| \| \mathbf{w}_b \|}. \tag{1}$$

Consider a collection of short texts, $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_i, \ldots, \mathbf{s}_n\}$, where $\mathbf{s}_i$ represents the i_{th} short text. We assume short text $\mathbf{s}_i$ of length U is represented as

$$\mathbf{v}_i = \mathbf{w}_1 \oplus \mathbf{w}_2 \oplus \cdots \oplus \mathbf{w}_i \oplus \cdots \oplus \mathbf{w}_U, \tag{2}$$

where $\mathbf{v}_i$ is the vector of $\mathbf{s}_i$, $\oplus$ is the concatenation operator, $\mathbf{w}_i$ is the i_{th} word vector in the short text $\mathbf{s}_i$. In general, each short text is represented as a concatenation of words. Our approach is different from [16] that assumes documents are represented as normalized bag-of-words (nBOW) vectors, which in turn has a high time complexity and needs more storage.

Let $\mathbf{v}_i$ and $\mathbf{v}_j$ be the representation of two short texts containing U and R words, respectively. First, let $\mathbb{T} \in \mathbb{R}^{U \times R}$ be a distance matrix where $\mathbb{T}_{u,r}$ denotes the distance between word u in $\mathbf{v}_i$ and word r in $\mathbf{v}_j$. Further, we average the minimum value for each row of $\mathbb{T}$ to represent the EMAD of $\mathbf{v}_i$ from $\mathbf{v}_j$, namely, $d(\mathbf{v}_i \parallel \mathbf{v}_j) = \frac{1}{U} \sum_u \min(\mathbb{T}_u)$. Correspondingly, the EMAD of $\mathbf{v}_j$ from $\mathbf{v}_i$ can be calculated by averaging the minimum value of each column, namely, $d(\mathbf{v}_j \parallel \mathbf{v}_i) = \frac{1}{R} \sum_r \min(\mathbb{T}_r)$. It should be emphasized that EMAD is not a genuine metric because it is not symmetric. We observe that for some words, the number of semantic relatedness words is very small. The EMAD of the short text s_i containing these words from other short texts is often larger than the EMAD of other sentences from s_i. Thus, we create a symmetric distance measure by setting $d(\mathbf{v}_i, \mathbf{v}_j) = \min(d(\mathbf{v}_i \parallel \mathbf{v}_j), d(\mathbf{v}_j \parallel \mathbf{v}_i))$ to capture more semantically related word pairs.

Figure 1 illustrates the EMAD metric on three short texts. After removing stop words, each arrow represents the minimum distance between two words from different short texts. As can be seen in Fig. 1, both short texts s_1 and s_2 do not share words with s_0. However, the distance from s_0 to s_1 (0.47) is significantly smaller than to s_2 (0.84). Differently from [16], EMAD achieves both fast speed and less storage as it employs the concatenation of words to represent short texts instead of nBOW representation. Furthermore, the Euclidean similarity used in [16] is not an optimal semantic measure for word embeddings. By contrast, we exploit the cosine similarity, which better describes the semantic relatedness between word embeddings [20], to directly capture word pairs that are likely to come from the same topic.

After obtaining the distance between short texts, we then aggregate short texts into long pseudo-documents based on a clustering algorithm. The k-means method is based on the centroid technique to represent the cluster, and it is sensitive to outliers. This means, a data object with an extremely large value may substantially disrupt the distribution of data. To overcome the problem, we use k-medoids method that is based on representative object techniques where centroids are replaced with medoids to represent clusters. The medoid is the most centrally located data object in a cluster.

The k-medoids algorithm used in this paper is described in Algorithm 1. Here, M short texts are selected randomly as medoids to represent M clusters. Each remaining short text is clustered with the medoid to which it is the most similar. Meanwhile, to tackle the data sparseness problem, we add a constraint that each group has nearly the same number of short texts. After processing all short texts, new medoids are determined as those which represent clusters in a better way and the entire process is repeated. Again all short texts are bound to the clusters based on the

new medoids. In each iteration, medoids change their location step by step until no more changes are done. Finally, this algorithm aims at minimizing an objective function:

$$J = \sum_{j=1}^{m} \sum_{i=1}^{n} d(\mathbf{v}_i, \mathbf{c}_j), \tag{3}$$

where $d(\mathbf{v}_i, \mathbf{c}_j)$ is a EMAD measure between short text $\mathbf{v}_i$ and the cluster medoid $\mathbf{c}_j$. The objective function is an indicator of the distance of short texts from their respective cluster medoids. After k-medoids clustering, all short texts are grouped into M long pseudo-documents.

Algorithm 1: Short text clustering algorithm

Input: M: the number of clusters, D: a dataset containing n short texts
Output: a set of M clusters

```
arbitrarily choose M short texts in D as the initial representative medoids;
repeat
    assign each short text to the nearest medoid until this cluster is full (the number of short texts in this
    cluster equals to M/n), then assign remaining short texts, without taking the full cluster into account
    anymore;
    compute the sum of distance J by Eq. (3);
    select a non medoid short text v_random to replace the cluster medoid c_j randomly ;
    assign each short text and compute the current J* ;
    if J* < J then
        swap c_j with v_random to form the new set of M medoid;
    end
until no change;
```

3.2 Model and inference

In this section, we present how to infer the topics from the long pseudo-documents using CRFTM and parameter estimation based on collapsed Gibbs sampling. CRFTM takes advantage of global semantic correlations to encourage semantically related words to share the same topic. Local semantic correlations are used to filter irrelevant words resulting from the property of word embeddings. Next, we present the details of the proposed model CRFTM.

Conditional Random Field (CRF) is a probabilistic graphical model which is used to encode various known relationships between observations [17]. As shown in Fig. 3, CRFTM extends the standard LDA model by imposing a CRF on the latent topic layer to incorporate both global and local semantic correlations in topic assignments. Among the variables, M denotes the number of pseudo-documents, K denotes the number of hidden topics. Each pseudo-document m has N_m words. w_{mn} is the observed value of the n_{th} word in pseudo-document m. α and β are hyper-parameters, α denotes the relative strength of latent hidden topics in the pseudo-document set, and β denotes the probability distribution of all hidden topics. θ_m denotes the topic probability distribution for certain pseudo-document m. ϕ_k denotes the word distribution for certain hidden topic k. z_{mn} denotes the topic label assigned to the n_{th} word in pseudo-document m.

Global semantic correlation It is reasonable that the words with a high semantic correlation should be clustered together under the same topic [19]. To do this, we continue to measure

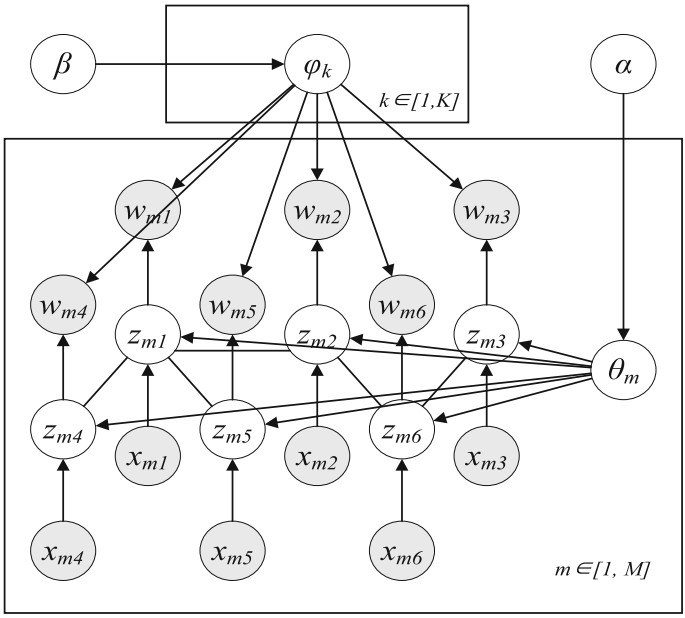


Fig. 3 Graphical representation of CRFTM

the distance between two words by their cosine distance in the embedding space. The key idea is that if the distance between two words in one pseudo-document is smaller than a threshold μ, we assume that they have a global semantic correlation with each other and they are more likely to belong to the same topic. For example, in Fig. 1, "*President*" and "*Trump*" ("*baseball*" and "*football*") have a high probability to come from the same topic. Based on this idea, CRFTM defines a CRF over the latent topic layer. Given a pseudo-document m containing N_m words $\{w_{mi}\}_{i=1}^{N_m}$, we examine each word pair (w_{mi}, w_{mj}). If they are semantically correlated, namely $d(w_{mi}, w_{mj}) < \mu$, we create an undirected edge between their topic labels (z_{mi}, z_{mj}). Shown in Fig. 3, there are five edges $\{(z_{m1}, z_{m2}), (z_{m1}, z_{m4}), (z_{m1}, z_{m5}), (z_{m2}, z_{m6}), (z_{m3}, z_{m6})\}$.

Local semantic correlation Word embedding learning methods learn one representation per word, which is problematic for polysemous and homonymous words and could incur some noise into the topic inference process. Next, we introduce the local semantic correlation that is capable of alleviating the semantic ambiguity problem on short text topic modeling. Specifically, for each word $\{w_{mi}\}_{i=1}^{N_m}$ in m, we find its P-nearest words in the current pseudo-document m according to the distance metric, called contextual words, and store them in $\{x_{mi,p}\}_{p=1}^{P}$. For instance, in Fig. 1, the top 2-nearest words of "*Apple*" in short text s_2 are {"*iPhone*", "*iPod*"}. If the difference of the average distance between word w_{mi} and its contextual words x_{mi} and the average distance between another word w_{mj} and x_{mi} is smaller than a threshold ε, i.e., $\frac{1}{P}\sum_{p=1}^{P}|d(w_{mi}, x_{mi,p}) - d(w_{mj}, x_{mi,p})| < \varepsilon$, we argue that w_{mi} has a local semantic correlation with w_{mj}. Accordingly, it is reasonable that words (e.g., Fig. 1, "*Apple*" and "*orange*") with no local semantic correlation should not be assigned to the same topic, even if they are globally correlated.

Under the CRFTM model, the joint probability of all topic assignments $\{z_{mi}\}_{i=1}^{N_m}$ in pseudo-document m can be written as:

$$p(\mathbf{z_m} \mid \theta_\mathbf{m}, \mathbf{x_m}, \lambda) = \prod_{i=1}^{N_m} p(z_{mi} \mid \theta_m) \mathbf{\Psi}(\lambda, z_{mi}, x_{mi}), \tag{4}$$

where $\mathbf{\Psi}$ denotes the potential function which takes into consideration both the effect of global and local semantic correlations, having the form:

$$\mathbf{\Psi}(\lambda, z_{mi}, x_{mi}) = \exp\left(\frac{\lambda}{A}\left(\sum_{(mi,mj)\in E} f(z_{mi}, z_{mj}) + \sum_{(mi,mj)\in E} g(z_{mi}, z_{mj}, x_{mi})\right)\right). \tag{5}$$

In standard LDA, topic label z_{mi} only depends on topic proportion vector θ_m. In CRFTM, z_{mi} not only depends on θ_m, but also depends on the topic labels of globally and locally correlated words. In Eq. 5, A is a normalization term and E represents all edges which connect the topic assignments of correlated words. Trade-off parameter λ controls the amount of promotion for each semantically related word w_{mj} when working on word w_{mi}. If λ is set to zero, CRFTM model is reduced to LDA.

The unary potential f produces a large value if the two topic labels are the same and generates a small value if the two topic labels are different. Therefore, it encourages globally correlated words to be assigned to the same topic. The unary potential is defined as:

$$f(z_{mi}, z_{mj}) = \begin{cases} 1 & \text{if } z_{mi} = z_{mj} \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

The pairwise potential g is used to reduce local semantic ambiguity during the topic inference process, which involves adding penalty to Eq. 5 if globally correlated word pair (w_{mi}, w_{mj}) has no local semantic correlation. That is:

$$g(z_{mi}, z_{mj}, x_{mi}) = \begin{cases} 0 & \text{if } \frac{1}{P}\sum_{p=1}^{P} |d(w_{mi}, x_{mi,p}) - d(w_{mj}, x_{mi,p})| < \varepsilon \\ -1 & \text{otherwise.} \end{cases} \tag{7}$$

Given the topic labels, the generation of words is the same as LDA. w_{mi} is generated from the topic-words multinomial distribution $\phi_{z_{mi}}$ corresponding to z_{mi}. The generative process can thus be summarized as follows:

1. Draw a topic proportion $\mathbf{\Theta} \sim Dir(\alpha)$
2. For each topic k
   (a) Draw a word proportion $\phi_k \sim Dir(\beta)$
3. For each pseudo-document m
   (a) Draw a topic assignment $\mathbf{z}_m$ according to Eq. 4
   (b) Draw the observed word $w_{mi} \sim Mult(\phi_{z_{mi}})$

Gibbs sampling Several approaches have been proposed for inferring topic models, such as expectation propagation, variational Bayes and Gibbs sampling. Collapsed Gibbs sampling integrates out irrelevant (nuisance) parameters when conducting inference. This results in a faster inference especially for a complex graphical model as ours where computational burden at each iteration is reduced considerably compared to the uncollapsed Gibbs sampling technique. In CRFTM, we adopt collapsed Gibbs sampling to do posterior inference according to the following condition distribution:

$$p(z_{mi} = k|\mathbf{z}_{m,-mi}, \mathbf{w}) = \frac{p(\mathbf{z}, \mathbf{w}|\alpha, \beta, \lambda)}{p(\mathbf{z}_{m,-mi}, \mathbf{w}|\alpha, \beta, \lambda)}, \tag{8}$$

where z_{mi} is the topic variable for word w_{mi} in the pseudo-document m, $\mathbf{z}_{m,-mi}$ is the topic assignment for all words except the current word w_{mi}. In CRFTM, this joint distribution $p(\mathbf{z}, \mathbf{w}|\alpha, \beta, \lambda)$ can be factored:

$$p(\mathbf{z}, \mathbf{w}|\alpha, \beta, \lambda) = p(\mathbf{w}|\mathbf{z}, \beta)p(\mathbf{z}|\alpha, \lambda). \tag{9}$$

Because the first term is independent of α and λ, and the second term is independent of β, both elements of the joint distribution can now be handled separately. The first term can be obtained by integrating over Φ,

$$p(\mathbf{w}|\mathbf{z}, \beta) = \prod_{z_{mi}=1}^{K} \frac{\triangle(\mathbf{n}_{z_{mi}} + \beta)}{\triangle(\beta)}, \qquad \mathbf{n}_{z_{mi}} = \{n_{z_{mi}}^{(w)}\}_{w=1}^{V}, \tag{10}$$

where we use the notation $n_{z_{mi}}^{(w)}$ to denote the number of times that word w has been observed with topic z_{mi}, V is the size of the vocabulary and $\triangle(\cdot)$ represents the Dirichlet delta function [8].

Analogous to $p(\mathbf{w}|\mathbf{z}, \beta)$, $p(\mathbf{z}|\alpha, \lambda)$ can be derived by integrating over Θ:

$$\begin{aligned} p(\mathbf{z}|\alpha, \lambda) &= \prod_{m=1}^{M} \frac{\triangle(\mathbf{n}_m + \alpha)}{\triangle(\alpha)} \prod_{i=1}^{N_m} \boldsymbol{\Psi}(\lambda, z_{mi}, x_{mi}), \\ \mathbf{n}_m &= \{n_m^{(k)}\}_{k=1}^{K}, \end{aligned} \tag{11}$$

where $n_m^{(k)}$ refers to the number of times that topic k has been observed with a word of pseudo-document m.

From the joint distribution $p(\mathbf{z}, \mathbf{w}|\alpha, \beta, \lambda)$, we can derive the full conditional distribution for word token w_{mi} in pseudo-document m:

$$\begin{aligned} p(z_{mi} = k|\mathbf{z}_{m,-mi}, \mathbf{w}) &\propto \frac{p(\mathbf{z}, \mathbf{w}|\alpha, \beta, \lambda)}{p(\mathbf{z}_{m,-mi}, \mathbf{w}_{m,-mi}|\alpha, \beta, \lambda)} \\ &\propto \frac{\triangle(\mathbf{n}_m + \alpha)}{\triangle(\mathbf{n}_{m,-mi} + \alpha)} \frac{\triangle(\mathbf{n}_{z_{mi}} + \beta)}{\triangle(\mathbf{n}_{z_{m,-mi}} + \beta)} \boldsymbol{\Psi}(\lambda, z_{mi} = k, x_{mi}), \\ &\propto (n_{m,-mi}^{(w_{mi})} + \alpha) \frac{n_{k,-mi}^{(w_{mi})} + \beta}{n_{k,-mi} + V\beta} \boldsymbol{\Psi}(\lambda, z_{mi} = k, x_{mi}) \end{aligned} \tag{12}$$

where the counts $n_{\cdot,-mi}^{(\cdot)}$ indicate that the token w_{mi} is excluded from the corresponding pseudo-document m or topic k.

According to their definitions as multinomial distributions with Dirichlet prior, we can estimate the multinomial parameter sets Θ and Φ as follows:

$$\phi_{k,w} = \frac{n_k^{(w)} + \beta}{\sum_{w=1}^{V} n_k^{(w)} + V\beta} \tag{13}$$

$$\theta_{m,k} = \frac{n_m^{(k)} + \alpha}{\sum_{k=1}^{K} n_m^{(k)} + K\alpha}. \tag{14}$$

The details of the Gibbs sampling process of CRFTM are described in Algorithm 2. At first, CRFTM initializes the topic assignment for each pseudo-document with a uniform distribution (Lines 1–3). This initialization process is the same as in LDA. Then, a word semantic correlation matrix $\mathbb{M}$ can be constructed by calling the function CalcCorrelation()

(Line 4), consisting of all the global and local semantic correlations between each word pair in each pseudo-document. In each iteration of Gibbs sampling, we firstly figure out whether each word w has global semantic correlations with other words in the current pseudo-document. If $w \in \mathbb{M}$, the topic of word w is resampled based on the conditional distribution in Eq. 12. Otherwise, the topic of word w is resampled as the same in LDA (Lines 5–17).

Algorithm 2: Gibbs sampling algorithm for CRFTM

Input: M pseudo-documents, α, β, λ, thresholds ε and μ, topic number K
Output: Θ, Φ

```
1  initialize Θ and Φ to zeros;
2  initialize topic assignment z, randomly for all word tokens;
3  initialize all count variables;
4  𝕄 ← CalcCorrelation(M);
5  while not finished do
6      for all pseudo-documents m ∈ [1, M] do
7          for all words n ∈ [1, N_m] in m do
8              if w_mn ∈ 𝕄 then
9                  sample topic index acc. to Eq. (12);
10             end
11             else
12                 sample topic index as the same in LDA;
13             end
14             update all count variables;
15         end
16     end
17 end
18 compute Φ by Eq. (13) and Θ by Eq. (14)
```

Complexity analysis We now analyze the time complexity of the Gibbs sampling algorithm of CRFTM with LDA. LDA draws a topic for each word occurrence, giving an overall time complexity in an iteration is $O(KM\bar{l})$, where K is the number of topics, M is the number of documents, $\bar{l}$ is the average number of words per document. Instead, CRFTM leverages similarity relationships among words to improve the coherence of learned topics, with the time complexity in an iteration $O(KM(\bar{l} + \bar{e}))$, where $\bar{e}$ is the average number of edges per pseudo-document. In other words, the $\bar{e}$ value depends on the average number of word pairs with semantic correlations in each pseudo-document. Since only few words have semantic correlations with other words in a pseudo-document, it is expected that $\bar{e} \ll \bar{l}$ (e.g., $\bar{e} = 29.78$ and $\bar{l} = 233.38$ in the StackOverflow collection). Therefore, the run-time of CRFTM is still comparable with LDA.

4 Experiments

In this section, we empirically evaluate the effectiveness of CRFTM[1] by comparing it with five baseline methods. The performance in terms of topic coherence and short text classification are reported over two real-world datasets, i.e., an English news dataset and a Q&A dataset. The experimental results show that our proposed model provides promising performance on both datasets.

[1] Code of CRFTM: http://github.com/nonobody/CRFTM.

4.1 Experimental setup

4.1.1 Datasets

Our method is tested on two real-world short-text corpora. In the following, we give brief descriptions to them.

News This is a dataset[2] of 31,150 English news articles extracted from RSS feeds of three popular newspaper websites (nyt.com, usatoday.com, reuters.com). Categories are: sport, business, U.S., health, sci&tech, world and entertainment. We retain news descriptions since they are typical short texts.

StackOverflow This dataset[3] consists 20,000 question titles from 20 different tags which are randomly selected by Xu et al. [38] from the challenge data published in Kaggle.com.

For these datasets, we performed the following preprocessing: (1) convert letters to lowercase; (2) remove all non-alphabetic characters and stop words[4]; (3) remove words with fewer than 3 characters; (4) remove words with document frequency less than 3 in the dataset.

4.1.2 Baseline methods

We compared our model with two normal text topic models and three typical methods for short text topic modeling:

- **LDA** is one of the most classical topic models, and works as the basis of our model [3].
- **MRF-LDA** is one novel model designed to incorporate word knowledge into topic modeling [37]. Instead of extracting word semantic relatedness knowledge from Web Eigenwords, we use Word2Vec [26] as external word correlation knowledge for comparison.
- **BTM** learns topics by directly modeling the generation of word co-occurrence patterns in the short text corpus [5]. In BTM, a biterm is an unordered word pair co-occurred in a short context.
- **GPU-DMM** is a recently published topic model which integrates word embeddings into DMM by using the generalized Pólya urn (GPU) model [19].
- **PTM** is a topic model for short texts. By leveraging much less pseudo-documents to self-aggregate a tremendous amount of short texts, PTM gains advantages in learning topic distributions without using auxiliary contextual information [41].

For BTM, GPU-DMM, MRF-LDA and PTM, we use the tools released by the authors. For LDA, we use jGibbLDA package[5] with collapsed Gibbs sampling which is provided online.

4.1.3 Evaluation measures

Topic coherence The evaluation of topic models is still an open problem. Traditionally, topic models are evaluated using perplexity that has been proved less correlated to human

[2] http://acube.di.unipi.it/tmn-dataset/.

[3] http://github.com/jacoxu/StackOverflow.

[4] Stop word list is from NLTK: http://www.nltk.org/.

[5] http://jgibblda.sourceforge.net.

interpretability. Many new metrics thus have been proposed by measuring the coherence of topics in documents and are proved more correlated to human judgments.

In this study, we use UCI topic coherence to compute topic coherence. The method presented in [28] uses the point-wise mutual information (PMI) to measure the coherence of topics. A higher UCI score indicates that the topic distributions are semantically more coherent. Given a topic k and its top N probable words $(w_1, w_2, \ldots, w_N)$, UCI score of k is calculated by:

$$UCI(k) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} PMI(w_i, w_j), \tag{15}$$

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}, \tag{16}$$

where $p(w_i, w_j)$ is the probability of words w_i and w_j co-occurring in the same sliding window. $p(w_i)$ and $p(w_j)$ are the probabilities of occurrence of words w_i and w_j in a sliding window, respectively. The UCI score for a topic model is the average score for all topics.

Classification measures Another reasonable way to evaluate a topic model is to apply the learned topics to an external task. The quality of the topics can be evaluated by their performance on the task [3]. Hence, we conduct short text classification experiments to compare the topic-level representation learned by our model and baselines. Considering topic modeling as a method for dimensionality reduction, the classification task reduces a short text to a fixed set of topical feature $p(z|d)$. Therefore, the quality of the topics can be assessed by the accuracy of short text classification using topic-level representation, as an indirect evaluation. A better classification accuracy implies the learned topics are more representative and discriminative. Following [19], we use summation over words (SW) representations to infer $p(z|d)$, which is a desired method for short texts:

$$p(z = k|d) \propto \sum_{w} p(z = k|w)p(w|d), \tag{17}$$

where $p(w|d)$ is estimated using the relative frequency of w in d.

Word intrusion To measure the interpretability of a topic model, a word intrusion detection task is proposed to involve subjective judgements [4]. The basic idea is to give human annotators a group of words with high probability in one topic, and an "intruder" word that is randomly chosen from another topic. The task of the annotators is to identify the intruders which are out of place or do not belong with the others. If the topic is semantically coherent, the intruder word should be easy to detect. If the topic is not strongly coherent, the annotators may choose an intruder word at random. Following [24], we use the accuracy of the word intrusion task as the evaluation metric (higher accuracy means higher interpretability).

In our experiments, we use the freely-available Word2Vec[6] word embedding which has a 300-dimensional embedding for 3 million words/phrases (from Google News), trained using the method in [26]. If a word has no embedding, the word is regarded as having no word semantic correlation knowledge.

For the baselines, we choose the parameters according to their original papers unless otherwise specified. For all the methods in comparison, we set the hyper-parameters $\alpha = 50/K$, $\beta = 0.01$ and run 1000 iterations of sampling. For GPU-DMM, MRF-LDA and our

[6] http://code.google.com/p/word2vec.

method, words pairs with distance lower than 0.3 ($\mu = 0.3$) are labeled as correlated. The number of pseudo-documents in PTM and our method is set to $n/50$ where n is the number of short texts in the corpus. For achieving the best classification results in GPU-DMM, we tune the promotion weight from 0.1 to 1.0 with a step of 0.1 and set it to 0.5 and 0.3, on the News and StackOverflow datasets, respectively. For MRF-LDA and CRFTM, λ is set to 1 as in their paper. For CRFTM, we empirically set $\varepsilon = 0.1$ and $P = 5$. All results reported below are averaged on five runs.

4.2 Experimental results

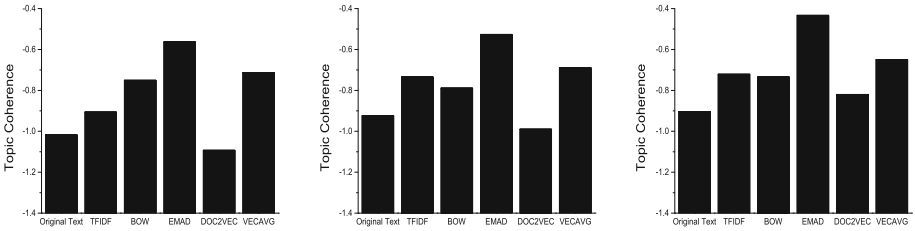
4.2.1 Effect of distance metric

In this paper, CRFTM utilizes EMAD to aggregate short texts into long pseudo-documents based on a clustering algorithm. To investigate the performance of our new distance metric EMAD, we compare its performance against four different types of document representation methods. The two most common ways documents are represented via bag of words (BOW) or by their term frequency-inverse document frequency (TF-IDF). Furthermore, we obtain a 300-dimensional distributed representation for each short text by learning a doc2vec [18] model from the News corpus. Doc2vec is inspired by the word embedding technique to extend the learning of embeddings from words to documents. In this work, we use a Gensim implementation[7] of the doc2vec algorithm with hyper-parameter settings recommended by [7] (Window Size = 15, Min Count = 5, Sub-Sampling = 10^{-5} and Epoch = 400). We also compute short text embeddings by taking the averaged vectors of the words they contain (VecAvg). The cosine similarity between their vectors is used to measure the similarity of two short texts. We study this by using a k-medoids clustering algorithm that is performed to aggregate short texts into pseudo-documents. Then, two normal text topic models and our model are applied for topics extraction from pseudo-documents. Since the clustering algorithm might lead to different clustering results given different starting points, we run each of them five times and report the average performance.

As shown in Fig. 4, two standard topic models LDA, MRF-LDA and the proposed model are studied with different methods on number of top words per topic $T = 10$ and number of topics $K = 60$ on the News dataset. From the figures, we observe that all models have poor performance on the original short texts. As being described in [10], directly applying standard topic models on short texts will suffer from the severe data sparsity problem. The other observation is that aggregating short texts into clusters can boost the performance of all topic models and EMAD constantly performs the best. The reason is that our method aggregates similar short texts together according to the semantic similarity between individual word pairs, which brings in additional useful word co-occurrence information across short texts. On the contrary, BOW and TFIDF perform worse than EMAD mainly because short texts do not provide sufficient word co-occurrence or context shared information for tf/tf-idf similarity measure [33]. Additionally, doc2vec performs the worst in the clustering task may be because doc2vec is designed to scale to large data. The News dataset is not enough for good training, leading to its poor results. This confirms the previous findings by [21] that the clustering performance of doc2vec is sensitive to the size of corpora.

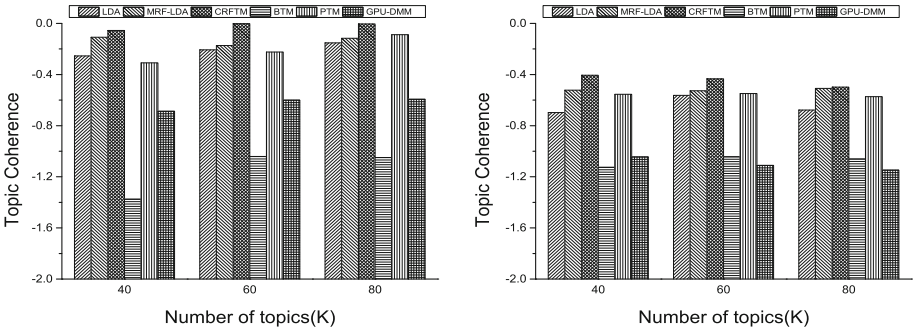
In conclusion, by adding informative cross-text word co-occurrence information, our method generates better pseudo-documents that can largely improve the quality of topics learned by standard topic models. In the following experiments, long text topic models (i.e.,

[7] https://radimrehurek.com/gensim/models/doc2vec.html.



(a) LDA **(b)** MRF-LDA **(c)** CRFTM

Fig. 4 Topic coherence of each model on the News dataset with different distance metrics



(a) Top-5 topic words **(b)** Top-10 topic words

Fig. 5 Topic coherence of each model on the News dataset with different settings on number of topics $K = \{40, 60, 80\}$

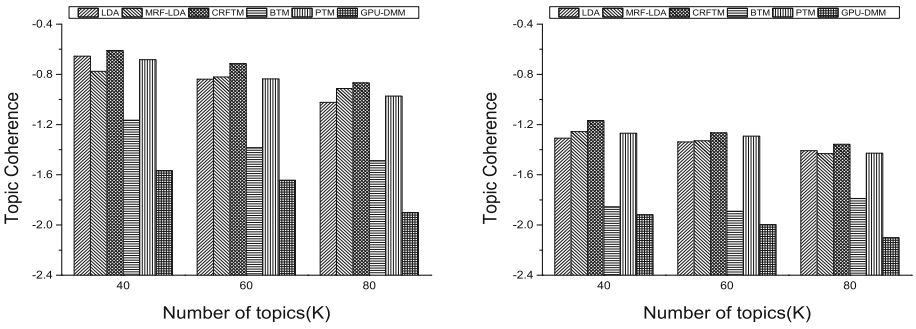
LDA and MRF-LDA) learn topics from the pseudo-documents, while short text topic models (i.e., BTM, GPU-DMM and PTM) learn topics from the original short texts due to their intrinsic characteristics, which do not fit long documents very well.

4.2.2 Topic evaluation by topic coherence

As we have stated in Sect. 4.1.3, UCI topic coherence is more appropriate for short texts [41]. In our experiments, we employ Palmetto,[8] which is a quality measuring tool for topics, to calculate UCI topic coherence. Palmetto uses 3 million English Wikipedia articles as an external corpus and the word co-occurrence counts are derived using a sliding window with the size 10.

The UCI topic coherence results of our method and all baselines on two datasets with number of top words per topic $T = \{5, 10\}$ and number of topics $K = \{40, 60, 80\}$ are presented in Figs. 5 and 6, respectively. From the figures, we can notice that CRFTM achieves the best performance compared with five baseline methods. The reason is that our method learns topics from regular-sized pseudo-documents generated by using the EMAD metric, which significantly enhance the quality of topics. As a result, MRF-LDA is the second best model in most cases and LDA always outperforms BTM and GPU-DMM on both datasets. Furthermore, CRFTM utilizes global semantic correlations to promote correlated words under the same topic label, and meanwhile, local semantic correlations are used to filter

[8] http://aksw.org/Projects/Palmetto.html.



(a) Top-5 topic words **(b)** Top-10 topic words

Fig. 6 Topic coherence of each model on the StackOverflow dataset with different settings on number of topics $K = \{40, 60, 80\}$

noise words. The experimental results validate that this mechanism guarantees the topics learned by our model are of high coherence and contain fewer irrelevant words. Another interesting phenomenon is that BTM and GPU-DMM perform the worst on the News and StackOverflow datasets, respectively. This may be because BTM brings in little additional word co-occurrence information and cannot alleviate the sparsity problem essentially, as discussed in [34,36]. As for GPU-DMM, one possible reason is that the StackOverflow dataset is very short, sparse, noisy and less topic-focused. In addition, even though it also exploits word embeddings, GPU-DMM lacks an appropriate strategy to accurately disambiguate word sense across topics and would falsely put irrelevant words into the same topic, resulting in its poorer results.

4.2.3 Topic evaluation by short text classification

Next, we evaluate the performance of all models in terms of short text classification. For each trained topic model, we perform five-fold cross-validation on both datasets. A linear kernel Support Vector Machines (SVM) classifier in sklearn[9] with default parameter settings is adopted for classification. A better classification accuracy means that the learned topics are more discriminative and representative.

The short text classification accuracy on the two datasets are shown in Table 1. We highlight the best results in bold. Note that BTM is not studied in the above classification task. The reason is that BTM uses a variant of SW post inference strategy by replacing w with biterm b [5], making it not directly comparable with other models. On the News dataset, CRFTM outperforms all other models in 2 out of 3 settings. PTM achieves the best performance in the rest setting. On the StackOverflow dataset, our model achieves the best performance across all settings. This demonstrates the effectiveness of our model against baselines in learning semantic representations of short texts. Additionally, the performance of LDA is relatively competitive, which further proves pseudo-documents generated by the proposed method contribute greatly to standard topic models when applied on short texts. Our model also consistently outperforms MRF-LDA on both datasets, which indicates exploiting local semantic correlations to filter ambiguous words is able to improve the performance of classification. It is worth noting that although PTM is the second best performing model on

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

Table 1 Average classification accuracy of the 5 models on two datasets, with different number of topic K settings

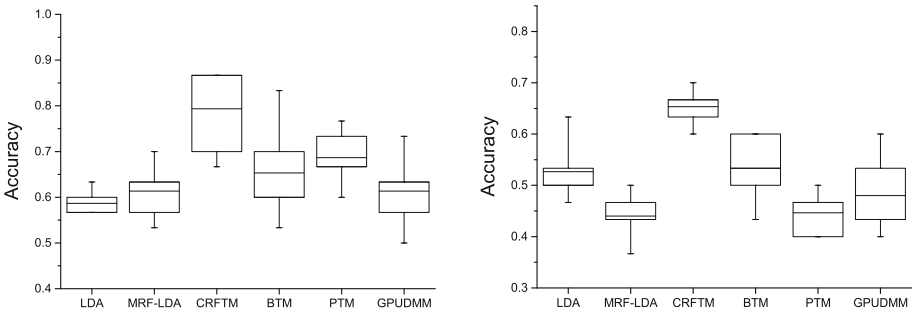
Dataset	Model	$K = 40$	$K = 60$	$K = 80$
News	LDA	75.52	76.24	76.99
	MRF-LDA	75.66	76.16	76.88
	PTM	75.82	**76.66**	77.18
	GPU-DMM	74.20	75.66	77.10
	CRFTM	**75.99**	76.31	**77.40**
StackOverflow	LDA	71.40	74.97	76.51
	MRF-LDA	70.58	75.03	76.73
	PTM	64.43	67.88	68.01
	GPU-DMM	68.44	70.98	71.83
	CRFTM	**71.51**	**75.71**	**77.03**

the News dataset, it produces the worst results on the StackOverflow dataset. This unstable performance implies that the performance of PTM is closely related to the training dataset, while CRFTM has a much more stable performance.

4.2.4 Topic evaluation by word intrusion

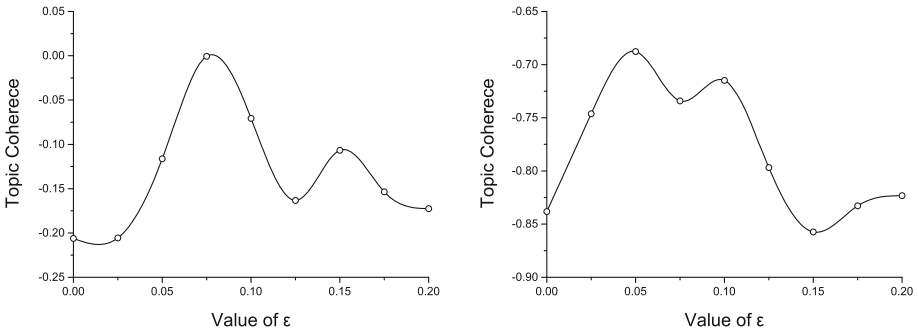
In the word intrusion task, for each model, we pick 60 topics and choose the top 5 words for each topic. An intruder is randomly sampled from a pool of words with low probability in the current topic but high probability in some other topics. In total, 6 words per topic are displayed to annotators in a random order. Five annotators, who are computational linguistics graduate students, are asked to identify the intruder words.

Figure 7 depicts boxplots of the accuracy for the 6 models on two datasets with number of topics $K = 60$. From this figure, we observe that the proposed model performs the best on both datasets, implying that the interpretability of CRFTM is better than all baselines. Although MRF-LDA produces a second best UCI topic coherence score, it does not perform as well in the word intrusion task. This may be because the semantic enhancement mechanism of MRF-LDA is likely to put noise words into the topics. Therefore, the intruder for one topic might be selected from these irrelevant words. Topics learned by LDA from short texts are often difficult to interpret [5]. However, in this experiment, LDA achieves a comparable



(a) News Dataset **(b)** StackOverflow Dataset

Fig. 7 Word intrusion detection accuracy Under $K = 60$



(a) News Dataset **(b)** StackOverflow Dataset

Fig. 8 Effect of the threshold ε under $K = 60$

performance with other state-of-the-art baselines, showing the effectiveness of integrating topic modeling with short text aggregation.

4.2.5 Effect of local semantic correlation threshold ε

We now study the effect of local semantic correlation threshold ε in CRFTM (see Eq. 7). All experiments are conducted by using number of topics $K = 60$ and number of top words per topic $T = 10$. The ε value determines the local semantic correlation. A small ε value leads to the result that few globally correlated word pairs have local semantic correlations with each other.

Figure 8 shows the UCI topic coherence score under $K = 60$ on the two datasets by varying the ε value from 0 to 0.2. Note that when $\varepsilon = 0$, CRFTM is reduced to the LDA model because no word pairs are promoted under the same topic. As shown in Fig. 8a, we observe that the proposed model gains significant improvements over LDA with any positive ε on the News dataset. Specifically, the best topic coherence is achieved when $\varepsilon = 0.075$. However, it is noted that the performance of CRFTM decreases when further increasing ε. This is reasonable because setting a larger ε value means more globally correlated words are assigned to the same topic. The chance of introducing noise words into a topic therefore becomes high. A similar pattern is also observed on the StackOverflow dataset, as shown in Fig. 8b.

4.2.6 Efficiency

In this section, we compare the running time of these models. To be fair, we implement CRFTM, MRF-LDA, BTM and LDA models in Java, and use the Java implementation provided by authors for PTM and GPU-DMM. All models use Gibbs sampling for inference, allowing direct computation time comparisons.

Table 2 shows the average running time (in seconds) per iteration for the models on the StackOverflow dataset. First, we can see the average iteration time increases with the number of topics. This is because there is a linear relationship between the complexity of the models and the number of topics. Secondly, not surprisingly, the simplest model LDA is the most efficient one. PTM is the slowest model in this comparison, which testifies our analysis in Sect. 2 that the inference process involving both topic sampling and text aggregation is

Table 2 The average running time (in seconds) per iteration over 100 iterations on StackOverflow dataset with different number of topic K settings

Model	$K = 40$	$K = 60$	$K = 80$
LDA	**0.052**	**0.077**	**0.097**
MRF-LDA	0.089	0.127	0.162
PTM	0.898	0.969	0.987
GPU-DMM	0.068	0.098	0.123
BTM	0.139	0.195	0.268
CRFTM	0.084	0.116	0.151

The best results are shown in bold

Table 3 Examples of noise word filtering on News dataset

Doc ID	Word pairs	Contextual words
389	Apple, apples	iPad, iPhone, Blackberry…
604	Obama, Cameron	Americans, NBC, Florida…
135	America, British	American, Americans, McDonald…
273	Twitter, tweet	Facebook, Myspace, Google…

time-consuming. Thirdly, the proposed model CRFTM is slightly slower than LDA over different topic numbers. As discussed in Sect. 3.2, although it increases the time complexity by applying a CRF model during the sampling process, the running time of CRFTM is still comparable with LDA. Furthermore, our model utilizes local semantic correlations to filter ambiguous words that are globally correlated, which leads to the fact that it is always faster than MRF-LDA.

4.2.7 Analysis on noise word filtering

Two baseline models (i.e., MRF-LDA and GPU-DMM) exploit word correlation knowledge from word embeddings to enrich topic modeling. However, using one embedding for a polysemous word or a homonymous word regardless of its specific word sense could improperly put noise words into the topics. In contrast, our model incorporate local semantic correlations computed by using cues from the word's context to identify the correct word sense.

In Table 3, we show four examples of noise word filtering by leveraging contextual words in CRFTM. The two words in each pair of the second column are globally correlated. The contextual words of the first word in each pair shown in the third column demonstrate there is no local semantic correlation between them, resulting in these word pairs appear with low probability in the same topic. Except for polysemous and homonymous words (e.g., "*Apple*"), we notice that noise word pairs often belong to the same named entity type such as person name (e.g., "*Obama*, "*Cameron*"), country name (e.g., "*America*", "*British*") and so forth. Additionally, CRFTM also filters the noise words that have little relationship with the sampled topics. For example, as shown in Table 3, although "*tweet*" is a message sent using "*Twitter*", the noise word "*tweet*" could be irrelevant to "*an IT company*" topic.

5 Conclusion

In this paper, we propose a new model for short text topic modeling, namely Conditional Random Field regularized Topic Model (CRFTM). CRFTM first utilizes the Embedding-based Minimum Average Distance (EMAD) to aggregate short texts into regular-sized pseudo-documents, which is a generalized solution to alleviate the sparsity problem. Next, our model incorporates global and local semantic correlations by using a Conditional Random Field (CRF) model to encourage semantically related words to share the same topic label. We conduct extensive experiments on two real-world short text collections. The experimental results validate the effectiveness of our model compared with existing state-of-the-art models. In the future, we will study how to apply our model on various data mining tasks such as tracing topic evolutions of short text streams or short texts retrieval.

Acknowledgements We thank anonymous reviewers for their very useful comments and suggestions. This research was partially supported by the National Science Foundation of China (NSFC, No. 61472291) and (NSFC, No. 61772382).

References

1. Alsmadi I, Hoon GK (2018) Term weighting scheme for short-text classification: Twitter corpuses. Neural Comput Appl 1–13
2. Bansal M, Gimpel K, Livescu K (2014) Tailoring continuous word representations for dependency parsing. In: Proceedings of the annual meeting of the association for computational linguistics (ACL), pp 809–815
3. Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022
4. Chang J, Gerrish S, Wang C, Boyd-Graber JL, Blei DM (2009) Reading tea leaves: How humans interpret topic models. In: Proceedings of advances in neural information processing systems (NIPS), pp 288–296
5. Cheng X, Yan X, Lan Y, Guo J (2014) Btm: topic modeling over short texts. IEEE Trans Knowl Data Eng 26(12):2928–2941
6. Das R, Zaheer M, Dyer C (2015) Gaussian LDA for topic models with word embeddings. In: Proceedings of the annual meeting of the association for computational linguistics (ACL), pp 795–804
7. Lau JH, Baldwin T (2016) An empirical evaluation of doc2vec with practical insights into document embedding generation. In: Proceedings of the workshop on representation learning for NLP (RepL4NLP), pp 78–86
8. Gregor H (2005) Parameter estimation for text analysis. Technical Report
9. Hofmann T (1999) Probabilistic latent semantic analysis. In: Proceedings of the conference on uncertainty in artificial intelligence (UAI), pp 289–296
10. Hong L, Davison BD (2010) Empirical study of topic modeling in twitter. In: Proceedings of the workshop on social media analytics (SOMA), pp 80–88
11. Huang EH, Socher R, Manning CD, Ng AY (2012) Improving word representations via global context and multiple word prototypes. In: Proceedings of the annual meeting of the association for computational linguistics (ACL), pp 873–882
12. Huang F, Ahuja A, Downey D, Yang Y, Guo Y, Yates A (2014) Learning representations for weakly supervised natural language processing tasks. Computational Linguistics 40(1):85–120
13. Huang J, Peng M, Wang H, Cao J, Gao W, Zhang X (2017) A probabilistic method for emerging topic tracking in microblog stream. World Wide Web J 20(2):325–350
14. Jin O, Liu NN, Zhao K, Yu Y, Yang Q (2011) Transferring topical knowledge from auxiliary long texts for short text clustering. In: Proceedings of the ACM conference on information and knowledge management (CIKM), pp 775–784
15. Khan FH, Qamar U, Bashir S (2017) A semi-supervised approach to sentiment analysis using revised sentiment strength based on SentiWordNet. Knowl Inf Syst 51(3):851–872
16. Kusner M, Sun Y, Kolkin N, Weinberger K (2015) From word embeddings to document distances. In: Proceedings of international conference on machine learning (ICML), pp 957–966
17. Lafferty JD, Mccallum A, Pereira FCN (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the international conference on machine learning (ICML), pp 282–289

18. Le Q, Mikolov T (2014) Distributed representations of sentences and documents. In: Proceedings of the international conference on machine learning (ICML), pp 1188–1196
19. Li C, Wang H, Zhang Z, Sun A, Ma Z (2016) Topic modeling for short texts with auxiliary word embeddings. In: Proceedings of the ACM conference on research and development in information retrieval (SIGIR), pp 165–174
20. Li S, Chua TS, Zhu J, Miao C (2016) Generative topic embedding: a continuous representation of documents. In: Proceedings of the annual meeting of the association for computational linguistics (ACL), pp 666–675
21. Ma S, Zhang C, He D (2016) Document representation methods for clustering bilingual documents. In: Proceedings of the annual meeting of the association for information science and technology (ASIST), pp 1–10
22. Mahmoud H (2008) Polya urn models. CRC press
23. Mehrotra R, Sanner S, Buntine W, Xie L (2013) Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In: Proceedings of the ACM conference on research and development in information retrieval (SIGIR), pp 889–892
24. Menini S, Nanni F, Ponzetto SP, Tonelli S (2017) Topic-based agreement and disagreement in us electoral manifestos. In: Proceedings of the conference on empirical methods in natural language processing (EMNLP), pp 2938–2944
25. Metzler D, Dumais S, Meek C (2007) Similarity measures for short segments of text. In: Proceedings of European conference on information retrieval (ECIR), pp 16–27
26. Mikolov T, Yih WT, Zweig G (2013) Linguistic regularities in continuous space word representations. In: Proceedings of the conference of the North American chapter of the association for computational linguistics: human language technologies (HIT-NAACL), pp 889–892
27. Mimno D, Wallach HM, Talley E, Leenders M, McCallum A (2011) Optimizing semantic coherence in topic models. In: Proceedings of the conference on empirical methods in natural language processing (EMNLP), pp 262–272
28. Newman D, Lau JH, Grieser K, Baldwin T (2010) Automatic evaluation of topic coherence. In: Proceedings of the conference of the North American chapter of the association for computational linguistics (NAACL), pp 100–108
29. Nguyen DQ, Billingsley R, Du L, Johnson M (2015) Improving topic models with latent feature word representations. Trans Assoc Comput Linguist 3:299–313
30. Ni X, Quan X, Lu Z, Wenyin L, Hua B (2011) Short text clustering by finding core terms. Knowl Inf Syst 27(3):345–365
31. Peng M, Gao W, Wang H, Zhang Y, Huang J, Xie Q, Hu G, Tian G (2017) Parallelization of massive textstream compression based on compressed sensing. ACM Trans Inf Syst 36(2):1–18
32. Peng M, Xie Q, Zhang Y, Wang H, Zhang X, Huang J, Tian G (2018) Neural sparse topical coding. In: Proceedings of the annual meeting of the association for computational linguistics (ACL), pp 2332–2340
33. Phan XH, Nguyen LM, Horiguchi S (2008) Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In: Proceedings of the conference on world wide web (WWW), pp 91–100
34. Quan X, Kit C, Ge Y, Pan SJ (2015) Short and sparse text topic modeling via self-aggregation. In: Proceedings of the international joint conferences on artificial intelligence (IJCAI), pp 2270–2276
35. Weng J, Lim EP, Jiang J, He Q (2010) Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the ACM conference on web search and data mining (WSDM), pp 261–270
36. Xia Y, Tang N, Hussain A, Cambria E (2015) Discriminative bi-term topic model for headline-based social news clustering. In: Proceedings of the Florida artificial intelligence research society conference (FLAIRS), pp 311–316
37. Xie P, Yang D, Xing E (2015) Incorporating word correlation knowledge into topic modeling. In: Proceedings of the conference of the North American chapter of the association for computational linguistics: human language technologies (HIT-NAACL), pp 725–734
38. Xu J, Wang P, Tian G, Xu B, Zhao J, Wang F, Hao H (2015) Short text clustering via convolutional neural networks. In: Proceedings of the conference of the North American chapter of the association for computational linguistics: human language technologies (HIT-NAACL), pp 62–69
39. Yin J, Wang J (2014) A Dirichlet multinomial mixture model-based approach for short text clustering. In: Proceedings of the ACM international conference on knowledge discovery and data mining (SIGKDD), pp 233–242
40. Zhao WX, Jiang J, Weng J, He J, Lim EP, Yan H, Li X (2011) Comparing twitter and traditional media using topic models. In: Proceedings of European conference on information retrieval (ECIR), pp 338–349

41. Zuo Y, Wu J, Zhang H, Lin H, Wang F, Xu K (2016) Topic modeling of short texts: a pseudo-document view. In: Proceedings of the ACM international conference on knowledge discovery and data mining (SIGKDD), pp 2015–2114

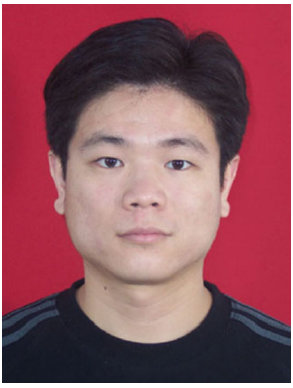
Wang Gao received M.S. degree in Software Engineering from Wuhan University of Technology, China, in 2011. He is currently pursuing Ph.D. degree in the School of Computer Science at Wuhan University. His current research interests include natural language processing and information retrieval.

Min Peng received the M.S. and Ph.D. degree from the Wuhan University, Wuhan, China, in 2002 and 2006. She is currently a Professor at School of Computer Science, Wuhan University. Currently, she works on machine comprehension, automatic text summarization and social network analysis. She is a member of the China Computer Federation (CCF).

Hua Wang is a full time Professor at the Centre for Applied Informatics, Victoria University. Before joining Victoria University, he was a Professor at the University of Southern Queensland (USQ) during 2011–2013. He obtained his Ph.D. in Computer Science from USQ in 2004.

Yanchun Zhang is Professor and Director of the Centre for Applied Informatics in Victoria University. His current research interests include databases, data mining, health informatics, web information systems and web services.

Qianqian Xie is a Ph.D. candidate in School of Computer Science at Wuhan University, Wuhan, China. She received her bachelors degrees from Jiangxi Normal University. Her current research areas include natural language processing, machine learning and deep learning.



Gang Tian received the Ph.D. degree from the Wuhan University, Wuhan, China, in 2011. He is currently a Lecturer at School of Computer Science, Wuhan University. His main research interests include big Data techniques and mining, machine vision, and machine learning.