Enabling Hierarchical Dirichlet Processes to Work Better for Short Texts at Large Scale

Khai Mai, Sang Mai, Anh Nguyen, Ngo Van Linh, and Khoat Than(✉)

Hanoi University of Science and Technology,
No. 1, Dai Co Viet Road, Hanoi, Vietnam
khaimaitien@gmail.com, magicmasterno1@gmail.com,
{anhnk,linhnv,khoattq}@soict.hust.edu.vn

Abstract. Analyzing texts from social media often encounters many challenges, including shortness, dynamic, and huge size. Short texts do not provide enough information so that statistical models often fail to work. In this paper, we present a very simple approach (namely, *bag-of-biterms*) that helps statistical models such as Hierarchical Dirichlet Processes (HDP) to work well with short texts. By using both terms (words) and biterms to represent documents, bag-of-biterms (BoB) provides significant benefits: (1) it naturally lengthens representation and thus helps us reduce bad effects of shortness; (2) it enables the posterior inference in a large class of probabilistic models including HDP to be less intractable; (3) no modification of existing models/methods is necessary, and thus BoB can be easily employed in a wide class of statistical models. To evaluate those benefits of BoB, we take Online HDP into account in that it can deal with dynamic and massive text collections, and we do experiments on three large corpora of short texts which are crawled from Twitter, Yahoo Q&A, and New York Times. Extensive experiments show that BoB can help HDP work significantly better in both predictiveness and quality.

Keywords: Online HDP · Bag of biterms · Document representation · Short texts

1 Introduction

Recently, with the explosion of social networks, microblogs, instant messages, Question & Answer forums, the number of texts that users generate is extremely massive. For example, according to the statistics from http://www.internetlivestats.com/twitter-statistics/, the number of tweets posted per day is about 500 millions. With the increase in the popularization of internet, this number will be even bigger in the future. These sources of information are really attractive to data scientists because analyzing them would provide a lot of insights into users. From this, companies can come up with many strategies in doing business.

J. Bailey et al. (Eds.): PAKDD 2016, Part II, LNAI 9652, pp. 431–442, 2016.
DOI: 10.1007/978-3-319-31750-2_34

Unfortunately, unlike formal or official documents (academic papers, news articles, etc.), data from these sources are often short texts and most are originated from social sources. Those kinds of data feature three challenges:

- **Short**: Short texts have extremely short length. In fact, in some popular social networks such as Twitter, users are restricted to write a status (tweet) with no more than 140 characters.
- **Massive**: The number of short texts is big and increases tremendously. For example, the number of tweets generated per day is more than 500 milions.
- **Dynamic**: The topics of short texts are highly dynamic and reflect the trends of the society. A topic might emerge and disappear over time.

The limited length of short texts poses various difficulties which have been revealed by existing studies [2,4,6,7,9–13,19]. Statistical models such as Latent Dirichlet Allocation (LDA) [3] and Hierarchical Dirichlet Processes (HDP) [15] often do not work well on short texts. Those models often base on the co-occurrence of terms. However, a short text often contains few co-occurrences and does not provide a clear context. In statistical perspectives [14], we can never recover/learn correctly a model from very short texts even though we may have arbitrarily large collections.

In this paper, we show a very simple approach to dealing with short texts. Instead of using bag-of-words to represent documents, we propose to use both terms (words) and biterms which lead to *bag-of-biterms* (BoB). By this way, BoB brings us many significant benefits:

(1) BoB naturally lengthens documents and thus helps us reduce bad effects of shortness. Therefore, a statistical model might be recovered better [14].
(2) BoB enables the posterior inference in a large class of probabilistic models including HDP to be less intractable [16]. This suggests that BoB helps learn a model better.
(3) No modification of existing models/methods is necessary, and thus BoB can be easily employed in a wide class of statistical models.

To evaluate the benefits of BoB, we take HDP into account as it is a popular model and is the base for many other models. HDP [15] is a nonparametric model which can automatically grow as more data come in. Therefore, HDP can deal with the dynamic of short texts. In combination with an online learning algorithm in [17], HDP can deal with two challenges (massive and dynamic) mentioned above. Data for evaluation are three big short text collections which have been crawled from Twitter, Yahoo Q&A, and New York Times.

From extensive experiments we find that in most cases BoB helps Online HDP [17] work significantly better than bag-of-words. Both predictiveness and quality of the learned models are significantly improved. We also find that BoB helps Online HDP less sensitive to learning rate paramaters, which is an important property in pratice. This suggests that BoB really helps us recover HDP better.

The rest of paper is organzied as follows. In Sect. 2, we present a short review of previous approaches to dealing with short texts. In Sect. 3, we present general

idea about Hierarchical Dirichlet Processes and the online learning algorithm [17]. In Sect. 4, we present BoB and related definitions. In Sect. 5, we describe our experiments and the evaluation of the new representation. In Sect. 6, we draw some conclusions for the paper.

2 Related Work

Besides simply applying traditional models such as LDA, HDP to short texts directly, there have been many other approaches to dealing with analyzing short texts. Here we summarize some of them to have a general view about previous approaches.

The first approach is to find additional context of the short texts by using powerful search engines (Google, yahoo, etc.) [4,12,19]. In this way, top results from searching keywords in each short document are used to evaluate its semantic meaning. One disadvantage of this method is the dependence on external tools, we cannot guarantee the quality of search engine for any keyword, especially those from social networks. Therefore it is hard to use this approach for a general short text corpus.

In another attempt, some researchers utilize external sources to deal with short texts [2,11,13]. More concretely, in both [2] and [13], the authors use Wikipedia as a source of additional information for short texts. For a short document, the authors search the closest articles from Wikipedia and take advantage of this extra information. On the other hand, in [11], the authors use Wikipedia as a universal knowledge to build a model by applying LDA. From built model, the authors do inference for each document and integrate the result into its vector. The weak point of this approach is there would be nothing sure about the correlation between the universal corpus and the short text corpora as they are almost generated from social networks with informal content and noises.

Grouping documents in rational ways is also a technique for dealing with short texts, as described in [6,7,9,10]. For example, the authors group tweets by hashtags or by the authors of tweets, the result is surely better when applying LDA to grouped texts. However, by grouping, one must know the metadata of the corpus, not all corpora of short texts have metadata as tweets. Therefore this technique is not a general approach to dealing with any corpus of short texts.

In [5,18] the authors directly model the word co-occurence pattern instead of a single word. More specifically, they generate all pairs of words (called biterm) for each document and aggregate all the biterms in all documents into one collection, and next they model the generative process for this collection. The good point of this approach is that it requires no additional information or any sources of knowledge. However, this approach does not model the generative process for each document, the authors use a heuristics to infer the topic proportions for each document. This does not guarantee the consistency between training phase and inferring phase.

In our work, we try to devise an approach that does not require any additional metadata or external sources of knowledge and able to take advantage of the

flexibility of HDP. More concretely, we deal with the problem of analyzing short texts by coming up with a new representation of document and applying it to Online HDP. By this way, we can fully deal with three challenges of short texts mentioned above.

3 Hierarchical Dirichlet Processes

In topic modeling, HDP [15] is considered as the nonparametric model of LDA [3], the generative process is

$$G_0 \sim DP(\gamma, Dir(\eta)), \quad G_d \sim DP(\alpha_0, G_0), \quad \phi_{z_{di}} \sim G_d, \quad w_{di} \sim Mult(\phi_{z_{di}})$$

where Dir is Dirichlet distribution, DP is Dirichlet process and $Mult$ is multinomial distribution. γ, η, α_0 are hyperparameters. ϕ_k (for $k \in \{1, 2, \ldots\}$) are corpus-level atoms. z_{di} is topic index of w_{di} and w_{di} is the i^{th} word in document d,

We follow [17] to use stick-breaking construction [15] to have a closed-form coordinate ascent variational inference:

$$G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}, \quad G_d = \sum_{t=1}^{\infty} \pi_{dt} \delta_{\phi_{c_{dt}}}, \quad c_{dt} \sim Mult(\boldsymbol{\beta})$$

$$\beta'_k \sim Beta(1, \gamma), \beta_k = \beta'_k \prod_{l=1}^{k-1}(1 - \beta'_l), \quad \pi'_{dk} \sim Beta(1, \alpha_0), \pi_{dt} = \pi'_{dt} \prod_{l=1}^{t-1}(1 - \pi'_{dt})$$

where c_{dt} are document-level topic indices. $Beta$ is Beta distribution.

Approximating the posterior distribution of its latent variables bases on the idea of mean-field variational inference [8]. The general idea of this technique is considering a family of distribution over latent variables, which is defined by free parameters and then finding the closest member of this family to the true posterior. Following [17], the variational distribution has the following form:

$$q(\mathbf{c}, \mathbf{z}, \boldsymbol{\phi}, \boldsymbol{\beta}', \boldsymbol{\pi}') = \prod_d \prod_t q(c_{dt} \mid \varphi_{dt}) \prod_d \prod_n q(z_{dn} \mid \zeta_{dn}) \prod_k q(\phi_k \mid \lambda_k)$$
$$\prod_{k=1}^{K-1} q(\beta'_k \mid u_k, v_k) \prod_d \prod_{t=1}^{T-1} q(\pi'_{dt} \mid a_{dt}, b_{dt})$$

where φ_{dt}, ζ_{dn} are the variational parameters of multinomial distribution, λ_k is the variational parameter of Dirichlet distribution, (u_k, v_k) and (a_k, b_k) are variational parameters of Beta distribution, K is topic-level truncation, T is document-level truncation. Using Jensen inequality for the log likelihood of observed data we obtain a lower bound called ELBO. Taking derivatives with respect to each variational parameter, we have the coordinate ascent updates, more details at [17].

Online learning for HDP bases on stochastic optimization [8]. The idea of the technique is based on computing noisy estimates of the gradient of ELBO

instead of the true gradient which requires iterating through the whole dataset. At time t the fast noisy estimates of gradient is computed by subsampling a small set of document (called minibatch). Based on noisy estimates, the intermediate global parameters $(\hat{\boldsymbol{\lambda}}, \hat{\mathbf{u}}, \hat{\mathbf{v}})$ are calculated and then update global parameters with a decreasing learning rate schedule: $\boldsymbol{\lambda}^{(t)} \leftarrow (1-\rho_t)\boldsymbol{\lambda}^{(t-1)} + \rho_t\hat{\boldsymbol{\lambda}}$, $\mathbf{u}^{(t)} \leftarrow (1-\rho_t)\mathbf{u}^{(t-1)} + \rho_t\hat{\mathbf{u}}$, $\mathbf{v}^{(t)} \leftarrow (1-\rho_t)\mathbf{v}^{(t-1)} + \rho_t\hat{\mathbf{v}}$ where $\rho_t \leftarrow (\tau+t)^{-\kappa}$ (κ is *forgetting rate*, τ is *delay*, see [8]). The procedure will converge to an optimum [8].

4 Bag-of-biterms and Representation of Documents

4.1 Bag-of-Words (BoW)

Bag-of-words (BoW) is the conventional representation of documents. In this representation, a document is considered as a container and words as items regardless of the information about the order of words. BoW has been widely used in common models for topic modeling such as LDA and HDP. Though being widely used in practice, BoW exhibits many limitations in modeling short texts [6]. The reason is that statistical models such as HDP often base on the statistical information about the co-occurence of words; when the document is short, the co-occurence is not enough to provide a clear context. This fact inspired us to try to find out a new representation that is applicable to HDP and more suitable than BoW for modeling short documents.

4.2 Bag-of-biterms (BoB)

First, we define a "biterm" is created by a pair of words that co-occur in some document. For example, given a document containing two words "character" and "story", (character, story) is a created biterm and (story, character) is also a created biterm.

Consider a document d with its set of distinct words $\{w_1, w_2, \ldots, w_n\}$ and associated frequency set $\{f_1, f_2, \ldots, f_n\}$. In BoB, this document is represented by set of created biterms $\{b_{ij}\}$ from words (w_i, w_j) where $i = 1, \ldots n$ and $j = 1 \ldots n$ and associated frequencies defined by:

- Frequency of b_{ii} from (w_i, w_i) is f_i
- Frequency of b_{ij} from (w_i, w_j) where $i \neq j$ is 1

For example, a document d with set of words $\{w_1, w_2, w_3\}$ and set of frequencies $\{2, 2, 4\}$. In BoB, this document is represented by set of biterms $\{b_{11}, b_{22}, b_{33}, b_{12}, b_{21}, b_{13}, b_{31}, b_{23}, b_{32}\}$ and set of frequencies $\{2, 2, 4, 1, 1, 1, 1, 1, 1\}$. About the frequency of biterms, we have the observation that short documents only have several distinct words, and each often appears 1 time, therefore, to guarantee that the influence of additional biterms would not surpass original words, we set 1 for the frequency of each biterm.

Note that our approach is different from that in [18]. About the definition, biterm in our perspective can be created from two identical words while biterm

in [18] is only created by two different ones. Moreover, the purpose of creating biterms in our approach is to find a new representation of documents and apply it to a wide class of statistical models, while in [18], the authors aggregate all biterms derived from all documents to form a collection of biterms and model this collection as a mixture of topics, there is no concern about documents in the course of modeling process, only in inferring step, the authors use a heuristic to infer the topic proportion for each document. This does not guarantee the consistency between modeling phase and infering phase.

Based on the definition presented above, we see that BoB has the same form like BoW, therefore applying BoB in HDP is the same as BoW. Moreover, we can see several properties of BoB:

- The document representations in BoB are improved thanks to additional biterms b_{ij} where $i \neq j$.
- Size of vocabulary in BoB (the number of distinct biterms) is higher than that of BoW. The fact is that each biterm is created from a pair of words.
- BoB is an expansion of BoW: BoB can be viewed as BoW adding additional "ingredients". In fact, in BoB, biterms b_{ii} can be considered as original words w_i and biterms $b_{ij}, i \neq j$ are additional ingredients.

These properties are the foundations for the explanation for the superiority of BoB in comparison with BoW which is presented in Sect. 4.3.

The Size of Vocabulary and Biterm Threshold in BoB. Note that in BoB, because of the symmetry of biterms b_{ij} and b_{ji}($i \neq j$) in every document, in practice, we can reduce the memory for storage by merging b_{ij} and b_{ji}, we only use b_{ij} $(i < j)$ with frequency 2 instead of 1.

The biggest challenge of BoB is the dilation of vocabulary set. In theory, the number of possible biterms might be up to $V_b = V(V-1)/2 + V = V(V+1)/2$ where V is the number of distinct words in corpus. This number is quite large that requires a lot of memory to store and the training time might last so long that makes the algorithm become infeasible. Fortunately, owing to the shortness of short texts, the practical number of distinct biterms is often not so large (see the description of datasets in Sect. 5.1) that storing and running are possible in a regular personal computer.

However, besides the purpose of improving the speed of learning the model and reducing the memory needed during training process, removing low-frequency biterms is also considered as the preprocessing step like removing words with low frequency (noises) in preprocessing text. Therefore, we need to have a threshold called "biterm threshold". The biterms with document frequency lower than this threshold are eliminated from the representation of document. The size of vocabulary in BoB can be adjusted by changing biterm threshold. In our experiments presented at Sect. 5, we also evaluate the influence of V_b on the quality of the model. Note that when biterm threshold is ∞, all biterms are removed, therefore BoB becomes BoW.

4.3 Explanation for the Potential Superiority of BoB for Short Texts

There are several reasons for believing in a better performance of BoB.

Firstly, BoB deals with the shortness of short documents. In BoB, the length of a document is much more longer than that in BoW. Namely, in BoB each document is $d(d-1)$ longer than document in BoW, where d is the number of distinct words in this document. In [14], the authors showed that document length plays an important role in topic modeling, when documents are extremely short, the learned model is expected to have poor performance. The idea of lengthening short documents has also been carried out in previous researches [6, 7,9,10]. In these researches, the authors aggregate short documents in a rational way to make them longer, consequently bring about better results.

Secondly, BoB increases the vocabulary size considerably. Recent research in topic modeling has indicated that the MAP inference for topic mixtures of each document can reach a global optimum when the length of document and the size of vocabulary are large enough [16]. Therefore, BoB can help us do posterior inference in topic models better than BoW. As a result, the learned model from BoB is expectedly better than that in BoW.

Finally, BoB is also considered as adding extra words to the original document in a rational way. In fact, biterms $b_{ii} = (w_i, w_i)$ can be treated as the original word w_i and biterms $b_{ij} = (w_i, w_j)$ with $i \neq j$ are the additional "ingredients". In a short document, due to the lack of information, the context becomes ambiguous and unclear. The ingredients being added would reinforce the context, make it clearer.

5 Experimental Results

In this section, we conduct extensive experiments on three different large short text collections to evaluate the effectiveness of our approach for dealing with short texts. We run Online HDP (as described in Sect. 3) in both BoW and BoB and compare their two performance measures and their sensitivity to learning rate parameters. We also investigate the influence of the size of vocabulary in BoB on the quality of the model. We use the source code of Online HDP from: http://www.cs.cmu.edu/~chongw/software/onlinehdp.tar.gz.

5.1 Datasets

To prepare for our experiments, we had crawled three large collections of short texts, as described below:

- *Yahoo Questions*: This dataset is crawled from https://answers.yahoo.com/, which is a forum that users post questions and wait for the answers from other users. Each document is a question.
- *Tweets*: This dataset is a set of tweets crawled from Twitter (http://twitter.com/) with 69 hashtags containing various kinds of topic. Each document is the text content of a tweet.

– *Nytimes Titles*: This dataset is a set of titles of articles from The New York Times (http://www.nytimes.com/) from 01/01/1980 to 29/11/2014. Each document is the title of an article.

These datasets went through a preprocessing procedure including tokenizing, stemming, removing stopwords, removing low-frequency words (appear in less than 3 documents) and removing extremely short documents (less than 3 words). About BoB, for each corpus we set a different biterm threshold (definition of biterm threshold in Sect. 4.2) to make sure that the number of created biterms is not so large. The detailed description for each dataset is on Table 1.

Table 1. Description of three datasets used in experiments

	Corpus size	Average length per doc	V	Biterm threshold	V_b
Yahoo Questions	537,770	4.73	24,420	2	722,238
Tweets	1,485,068	10.14	89,474	10	764,385
Nytimes Titles	1,684,127	5.15	55,488	5	756,700

5.2 Performance Measures

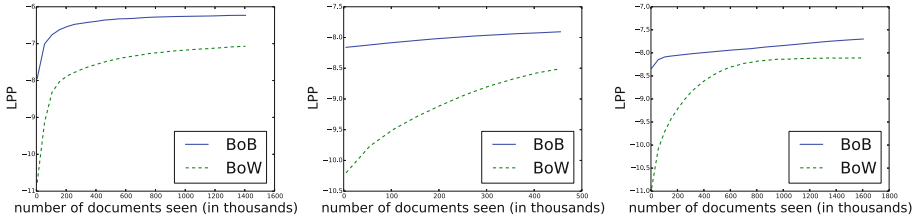
To compare the two representations, we use two very common performance measures in topic modeling: the LPP and NPMI.

Log Predictive Probability (LPP). LPP measures the predictiveness and generalization of a learned model to new data. The procedure for this measure is similar to [8]. For each dataset, we randomly separate into training set and test set. In test set, we only pick out documents with length greater than 4. For each chosen document, we divide it randomly into two part $(\mathbf{w}_1, \mathbf{w}_2)$ with ratio 4:1, we did inference for $\mathbf{w}_1$ and estimated the distribution of $\mathbf{w}_2$ given $\mathbf{w}_1$. In BoB, when estimating the distribution of $\mathbf{w}_2$ given $\mathbf{w}_1$, we need to convert the topic-over-biterms (distribution over biterms) to topic-over-words (distribution over words), the conversion fomula is described in Appendix.

Normalized Pointwise Mutual Information (NPMI). NPMI [1] measures the coherence of topics of a model. This measure indicates the quality of the topics learned from dataset by evaluating words with highest probability of each topic (called top words). Here we compute this measure by considering 10 top words for each topic. In case of BoB, we first need to convert the topic-over-biterms (distribution over biterms) to topic-over-words (distribution over words) the conversion fomula is described in Appendix.

5.3 Result

Performance Comparison. To evaluate thoroughly the performance of the new representation, we run Online HDP in various settings for each representation in each dataset. More concretely, The settings of learning rate parameters



(a) Tweets (b) Yahoo Questions (c) Nytimes Titles

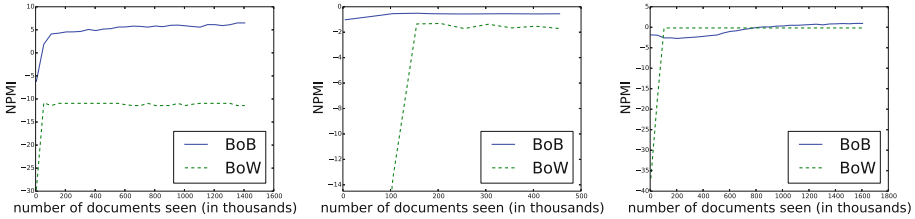
Fig. 1. Predictiveness on three datasets in both BoW and BoB.

(τ, κ) form a grid: $\tau \in \{1, 20, 40, 60, 80, 100\}$, $\kappa \in \{0.6, 0.7, 0.8, 0.9\}$. For each setting of (κ, τ), we fix the minibatch size = 5000, truncation for corpus K = 100, truncation for document T = 20 and $\alpha_0 = 1.0$, $\gamma = 1.0$.

We first compare the predictive capability of two representations. Figure 1 shows the median of 24 LPPs associated with 24 settings of (κ, τ) on three datasets in both representations. As we can see, When the number of documents seen increases, both Bow and BoB attain a better predictive capability regardless of learning rate parameters. However, in all three datasets, BoB outperforms BoW significantly. Moreover, BoB has good starting points while BoW always has poor ones and needs much time to be stable. This superiority of model learned from BoB can be explained by better inference for each document. As mentioned in Sect. 4.3, the goodness of this inference may originate from the longer length in each document, larger vocabulary size and clearer context by additional ingredients. As the inference for each document is more precise, the model is recovered better.

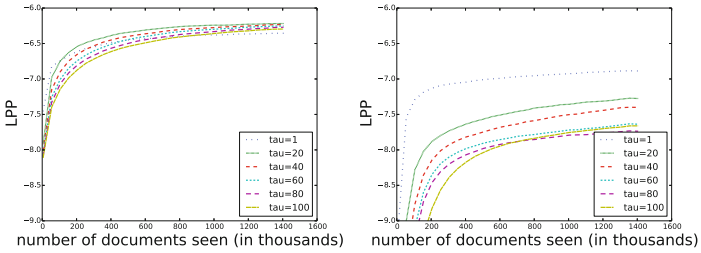
About the NPMI score, as we can see in Fig. 2 showing the median of 24 NPMIs associated with 24 settings of (κ, τ) in both BoB and BoW , there is also a huge distinction between BoB and BoW like LPP, especially in *Tweets*. Apart from outperforming BoW significantly, BoB also has much better starting points. All of these observations indicates the superiority of BoB to BoW in the quality of learned topics in Online HDP. Besides the reasons explained above, the goodness of BoB in terms of NPMI can also be explained by the conversion from topic-over-biterms to topic-over-words (presented in appendix). Biterm b_{ii} in BoB can be considered as w_i in BoW. However in BoB there are additional ingredients $b_{ij}, i \neq j$, these items would contribute their probability to w_i in BoB. Therefore, a word with more biterms containing it would be stressed more than a word with less biterms containing. By this way, the probability of words in each topic would converge to the true one more quickly and more precisely.

Sensitivity of Learning Rate Parameters. To compare the sensitivity of τ in BoB and BoW, we fix $\kappa = 0.9$ and explore different values of $\tau \in \{1, 20, 40, 60, 80, 100\}$, the result is described in Fig. 3. Similarly, to compare the sensitivity of κ in BoB and BoW, we fix $\tau = 20$ and explore different values of $\kappa \in \{0.6, 0.7, 0.8, 0.9\}$, the result is described in Fig. 4. Observing two figures, we



(a) Tweets (b) Yahoo Questions (c) Nytimes Titles

Fig. 2. Quality of the topics learned from BoW and BoB on three datasets. The higher is the better

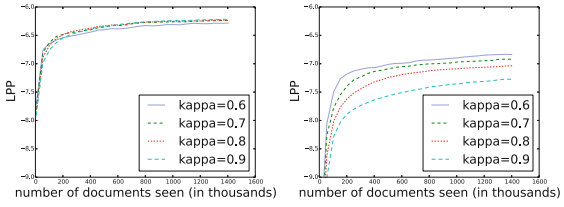


(a) BoB (b) BoW

Fig. 3. Sensitivity of κ in BoB and BoW when fixing $\kappa = 0.9$ in *Tweets*

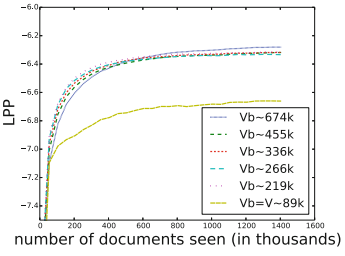
see that BoB is much less sensitive to both κ and τ than BoW. Less dependence on learning rate parameters is a good property of BoB in practice, one does not have to consider exhaustive model selection for learning rate parameters.

The Influence of Size of Vocabulary in BoB (V_b). To evaluate the influence of size of vocabulary in BoB (V_b), we fix $\kappa = 0.8, \tau = 80$ (this is the best setting of learning rate in BoB with dataset *Tweets*) and carry out experiments with varying V_b by changing biterm threshold in set $\{10, 15, 20, 25, 30, \infty\}$. Remember



(a) BoB (b) BoW

Fig. 4. Sensitivity of κ in BoB and BoW when fixing $\tau = 20$ in *Tweets*



(a) V_b

Fig. 5. LPP with varying sizes of V_b in BoB in *Tweets*

that as biterm threshold is ∞, BoB becomes BoW. The experimental result is described in Fig. 5. We observe that when $V_b = V$ (BoB becomes BoW), the model has the poorest predictive capability. The model attains best predictive performance when V_b is not the smallest or the biggest, it seems that there is a tradeoff in V_b. Namely when biterm threshold is large, there are many biterms being removed from document in BoB, as a result, the documents become shorter and we return to the problem of short text. On the other hand, if biterm threshold is small, there are many biterms being added to documents even they only appear in a small number of documents. These biterms can be considered as noises that might degrade the model quality.

6 Conclusion

We investigated a new representation which is called bag-of-biterms (BoB) for documents. We found that BoB has many interesting properties, which overcome some severe limitations of bag-of-words representation. It can help us learn significantly better statistical models such as HDP from short texts. It can be easily employed in a large class of statistical models and methods without any need of modification. It also compliments to existing approaches to deal with short texts. Therefore, we believe that BoB is a potential approach, and can be used in a wide context.

Acknowledgments. This work was partially supported by Vietnam National Foundation for Science and Technology Development (NAFOSTED Project No. 102.05-2014.28), and by AOARD (US Air Force) and ITC-PAC (US Army) under agreement number FA2386-15-1-4011.

Appendix: Conversion of topic-over-biterms (distribution over biterms) to topic-over-words (distribution over words)

In BoB, after we finish training the model, we obtain topics, each is the multinomial distribution over biterms and we want to convert to the topics with distribution over words. Assume that $\boldsymbol{\phi}_k$ is the distribution over biterms of topic k. According to probability:
$p(w_i \mid z = k) = \sum_{j=1}^{V} p(w_i, w_j \mid z = k) = \sum_{j=1}^{V} p(b_{ij} \mid z = k) = \sum_{j=1}^{V} \phi_{kb_{ij}}$,
As discussed in Sect. 4.2, in implementing BoB, we can merge b_{ij} and b_{ji} into b_{ij} with $i < j$. Because of identical occurence in every document, after finishing training process, the value of $p(b_{ij} \mid z = k)$ will be expectedly the same as $p(b_{ji} \mid z = k)$. Therefore, in grouping these biterms into one, the conversion version of this implementation is: $p(w_i \mid z = k) = \sum_{j=1}^{V} p(b_{ij} \mid z = k) = \phi_{kb_{ii}} + \frac{1}{2} \sum_{\text{b: biterms contain } w_i} \phi_{kb}$.

References

1. Aletras, N., Stevenson, M.: Evaluating topic coherence using distributional semantics. In: Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013), pp. 13–22 (2013)
2. Banerjee, S., Ramanathan, K., Gupta, A.: Clustering short texts using wikipedia. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 787–788. ACM (2007)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
4. Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring semantic similarity between words using web search engines. WWW **7**, 757–766 (2007)
5. Cheng, X., Yan, X., Lan, Y., Guo, J.: BTM: topic modeling over short texts. IEEE Trans. Knowl. Data Eng. **26**(12), 2928–2941 (2014)
6. Ye, C., Wen Wushao, P.Y.: TM-HDP: an effective nonparametric topic model for tibetan messages. J. Comput. Inf. Syst. **10**, 10433–10444 (2014)
7. Grant, C.E., George, C.P., Jenneisch, C., Wilson, J.N.: Online topic modeling for real-time twitter search. In: TREC (2011)
8. Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J.: Stochastic variational inference. J. Mach. Learn. Res. **14**(1), 1303–1347 (2013)
9. Hong, L., Davison, B.D.: Empirical study of topic modeling in twitter. In: Proceedings of the First Workshop on Social Media Analytics, pp. 80–88. ACM (2010)
10. Mehrotra, R., Sanner, S., Buntine, W., Xie, L.: Improving lda topic models for microblogs via tweet pooling and automatic labeling. In: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 889–892. ACM (2013)
11. Phan, X.H., Nguyen, L.M., Horiguchi, S.: Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In: Proceedings of the 17th International Conference on World Wide Web, pp. 91–100. ACM (2008)
12. Sahami, M., Heilman, T.D.: A web-based kernel function for measuring the similarity of short text snippets. In: Proceedings of the 15th International Conference On World Wide Web, pp. 377–386. ACM (2006)
13. Schönhofen, P.: Identifying document topics using the wikipedia category network. Web Intell. Agent Syst. **7**(2), 195–207 (2009)
14. Tang, J., Meng, Z., Nguyen, X., Mei, Q., Zhang, M.: Understanding the limiting factors of topic modeling via posterior contraction analysis. In: Proceedings of The 31st International Conference on Machine Learning, pp. 190–198 (2014)
15. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical dirichlet processes. J. Am. Stat. Assoc. **101**(476), 1566–1581 (2006)
16. Than, K., Doan, T.: Dual online inference for latent dirichlet allocation. In: Proceedings of the Sixth Asian Conference on Machine Learning (ACML), pp. 80–95 (2014)
17. Wang, C., Paisley, J.W., Blei, D.M.: Online variational inference for the hierarchical dirichlet process. In: International Conference on Artificial Intelligence and Statistics, pp. 752–760 (2011)
18. Yan, X., Guo, J., Lan, Y., Cheng, X.: A biterm topic model for short texts. In: Proceedings of the 22nd International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, pp. 1445–1456 (2013)
19. Yih, W.T., Meek, C.: Improving similarity measures for short segments of text. AAAI **7**, 1489–1494 (2007)