Information Sciences 393 (2017) 66–81

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A general framework to expand short text for topic modeling

Paulo Bicalho[a], Marcelo Pita[a,c], Gabriel Pedrosa[a], Anisio Lacerda[b], Gisele L. Pappa[a,*]

[a] *Computer Science Department, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil*
[b] *Computing Department, Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte, Brazil*
[c] *Serviço Federal de Processamento de Dados, Belo Horizonte, Brazil*

ARTICLE INFO

Article history:
Received 16 August 2016
Revised 19 December 2016
Accepted 2 February 2017
Available online 8 February 2017

Keywords:
Topic modeling
Short text
Word vector representation
Pseudo-documents

ABSTRACT

Short texts are everywhere in the Web, including messages posted in social media, status messages and blog comments, and uncovering the topics of this type of messages is crucial to a wide range of applications, e.g., context analysis and user characterization. Extracting topics from short text is challenging because of the dependence of conventional methods, such as Latent Dirichlet Allocation, in words co-occurrence, which in short text is rare and make these methods suffer from severe data sparsity. This paper proposes a general framework for topic modeling of short text by creating larger pseudo-document representations from the original documents. In the framework, document components (e.g., words or bigrams) are defined over a metric space, which provides information about the similarity between them. We present two simple, effective and efficient methods that specialize our general framework to create larger pseudo-documents. While the first method considers word co-occurrence to define the metric space, the second relies on distributed word vector representations. The pseudo-documents generated can be given as input to any topic modeling algorithm. Experiments run in seven datasets and compared against state-of-the-art methods for extracting topics by generating pseudo-documents or modifying current topic modeling methods for short text show the methods significantly improve results in terms of normalized pointwise mutual information. A classification task was also used to evaluate the quality of the topics in terms of document representation, where improvements in F1 varied from 1.5 to 15%.

1. Introduction

Short texts are everywhere in the Web, e.g., messages posted in social media, status messages, blog comments, questions in Q&A websites, etc. Automatically understanding the content of this type of message is crucial to a wide range of applications, including context analysis [6,35] and user characterization [18,30]. Among the techniques that can help uncover the content of these messages are topic identification methods.

Topic identification methods were conceived to find semantically meaningful topics from a document corpus. They assume that there are hidden variables (topics) that explain the similarities between observable variables (documents). The

[*] Corresponding author.
E-mail addresses: p.bicalho@dcc.ufmg.br (P. Bicalho), marcelo.pita@dcc.ufmg.br (M. Pita), gabrielmip@dcc.ufmg.br (G. Pedrosa), anisio@decom.cefetmg.br (A. Lacerda), glpappa@dcc.ufmg.br (G.L. Pappa).

http://dx.doi.org/10.1016/j.ins.2017.02.007
0020-0255/© 2017 Elsevier Inc. All rights reserved.

main representative of topic modeling methods is Latent Dirichlet Allocation (LDA) [1]. LDA has been applied to many different contexts to discover topics, including text, image and biological data [7,22]. However, as pointed out by Tang et al. [28], there are scenarios where LDA models are not "data-friendly". These scenarios include those where only a few documents are available, where documents contain too many topics or documents are too short. This paper focuses on the latter scenario.

Extracting topics from short text is difficult because of the dependence of topic modeling methods in words co-occurrence, which in short text is rare and makes conventional algorithms suffer from severe data sparsity [6]. Two different approaches have been proposed to address the problem of topic extraction from short text: (i) methods that propose new probabilistic topic models or modify the traditional LDA algorithm [9,17,35]; and (ii) methods that create larger pseudo-documents from short text documents, and then apply current topic modeling methods to these pseudo-documents [6,11,36]. The latter has the main advantage of being simpler and method-independent, since it only transforms the input data.

One of the main problems with current methods that generate larger pseudo-documents is that, most of the time, they use information about the data source or the context where they are being applied, and cannot be easily generalized for other contexts. For example, in [11], the authors propose different tweet pooling schemes to generate pseudo-documents from tweets. They found out that grouping tweets using a common hashtag to generate larger pseudo-documents is the most effective approach to generate larger pseudo-documents. However, depending on the number of different hashtags present in the data, this approach may reduce the number of documents significantly, generating another type of problem to LDA: dealing with few documents [28]. Furthermore, in scenarios where there is not an available common element to merge the documents (e.g., hashtags), this method cannot be applied.

This paper proposes a general framework for topic modeling in short text that is context-independent, allows one to specify the maximum desired size of the documents and creates pseudo-documents that can be given as input to any topic modeling method. The foundation of our framework is to identify similar components (e.g., words or bi-grams) from document corpora to enrich original short text documents. In our framework these components are defined over a metric space, which provides information about the similarity between pairs of components. We also present two methods that are specializations of this general framework to expand short text documents.

The first method, Co-Frequency Expansion (CoFE), exploits the co-occurrence frequency (co-frequency) of terms in the collection to define a metric space. It was first introduced in [19], and here is presented within the general framework we propose. The main idea behind CoFE is that words with high co-frequency have also high probability of belonging to the same topic, and hence can be used to expand documents. The second, Distributed Representation-based Expansion (DREx), is first introduced in this paper and exploits the powerful word embedding representation to model word similarities [14,20], taking advantage of the semantics and vector algebra captured by this type of representation.

We compare the results of the proposed strategies run with LDA with a tweet pooling scheme proposed in [11], two context-independent document expansion methods – WNTM [36] and Self-Term Expansion [23] – and two other state-of-the-art methods designed for short text: Biterm Topic Modeling [32] and Latent Feature-LDA [17]. All methods are evaluated using the normalized pointwise mutual information (NPMI) topic quality metric [2] and also within the context of a text classification task. Experimental results show that DREx outperforms the baselines, achieving higher values of NPMI and macro F1 score, with gains up to 15% in the latter.

The main contributions of this paper are:

- A new framework based on metric spaces for generating larger pseudo-documents that are more suitable for topic extraction;
- two instances of this framework, one based on word co-occurrence and the second based on word vectors, which can be coupled to any topic model to improve topic extraction;
- the results of the expansion based on word vectors are statistically significantly better than those obtained by the state-of-the-art methods in the original text both using the NPMI metric and when considering the classification task.

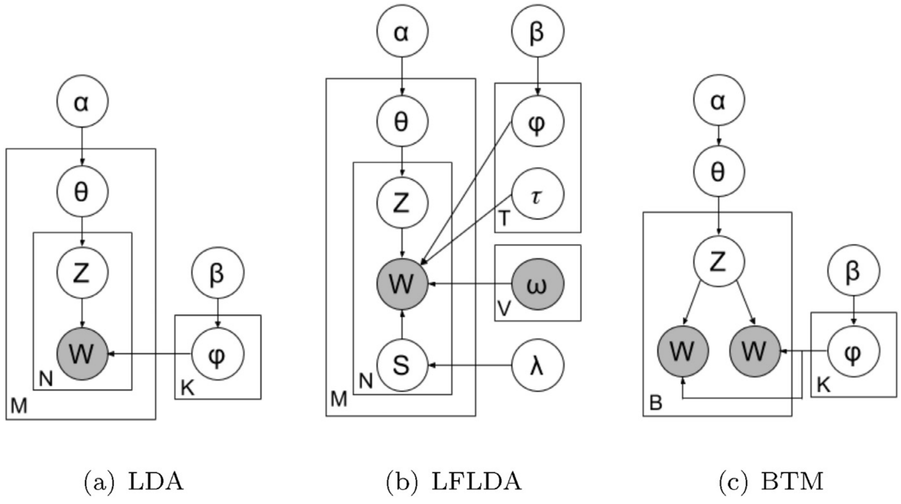
The remainder of this paper is organized as follows. Section 2 introduces related work on topic models for short text. Section 3 describes the general framework and instantiates CoFE and DREx. Section 4 introduces the experimental methodology and shows the results obtained. Finally, Section 5 lists our conclusions and directions of future work.

2. Probabilistic topic modeling for short text

The fast growth of user-generated content has raised the interest of researchers in developing more appropriate methods to extract useful information in short text scenarios [5,31]. Among these methods are those proposed to deal with topic extraction, which can be divided into two categories: (i) those that create larger pseudo-documents from short text documents; (ii) those that propose new topic models or modify the original LDA to better deal with data sparsity.

The first approach was inspired by solutions proposed for other scenarios where dealing with short text is also a challenge. Surveys published by Rosso et al. [26] and Rafeeque et al. [24], for instance, reviewed several works, applied mostly to text classification and clustering scenarios, which propose to overcome data sparsity with document expansion approaches. Similarly, Carpineto and Romano [3] presented different methods for expanding search engine queries.

Despite being originally designed for different tasks, one should also be able to apply these methods previously proposed in the literature in the topic modeling context [8,23,27]. Among techniques proposed so far, we highlight the work of Pinto



(a) LDA (b) LFLDA (c) BTM

Fig. 1. LDA, LF-LDA, and BTM graphical models in plate notation.

et al. [23]. They proposed Self-Term Expansion (STE), a document expansion method that does not require the use of any external source of data – while many other works rely on sources such as the WordNet or ontologies – and presents a set of similarities to one of the framework's specializations presented in this paper.

STE works by replacing document words with a set of correlated terms, where the correlation score is computed by analyzing co-occurrence of word pairs on the dataset. The method uses PMI score [10], which captures semantic associations between word pairs, to create a ranked correlation list for every word on the dataset, which is used to expand the corpus' documents.

Regarding document expansion methods previously proposed specifically for short text topic modeling, most approaches focused on expanding Twitter data. Hong et al. [6], for instance, proposed two different tweet aggregation schemes: one based on the tweets authors and another based on each word of the corpus vocabulary. Their goal was to infer a topic distribution for both messages and authors in the corpus. Mehrotra et al. [11] presented a similar approach. They evaluated four tweet pooling schemes in order to improve the results of LDA, including pooling tweets wrote by the same author, posted in the same hour, sharing hashtags and by trend topics. They found out that pooling tweets by hashtags yields superior performance when compared to the other proposed approaches.

Zuo et al. [36], in contrast, proposed a word co-occurrence network-based model named WNTM to generate pseudo-documents. In WNTM, an undirected weighted graph of word co-occurrence is derived from the original documents. Words that co-occur at least once in a document are linked, and the edge is weighted by the word co-occurrence frequency. Each word w_i of this graph generates a completely new pseudo-document, which is composed of the adjacent words to w_i in the graph. WNTM explores the fact that, even in short-text scenarios, the word-word space is rather dense, making the algorithm less sensitive to document length or heterogeneity of the topic distribution. Note that CoFE, one of our proposed methods, also uses a word co-occurrence graph to generate pseudo-documents. However, CoFE differs significantly from WNTM. While the former expands each document with vocabulary words that co-occur more frequently with each word of the document, WNTM creates entirely new documents from the graph.

Concerning works that generate or adapt existing methods for topic extraction, Zhao et al. [35] modified LDA to make it more suitable for Twitter. Their method, Twitter-LDA, assumes that a single tweet is usually about a single topic, and every Twitter user has a topic distribution that defines their probabilities of writing a tweet related to each topic. Jin et al. [9], in turn, proposed Dual LDA (DLDA), which enhances topic modeling for short texts via transfer learning from an auxiliary dataset of longer texts. They use URLs present in the short text messages that reference longer documents to produce the auxiliary dataset, and hence assume most documents in the collection do have links to longer documents. Previous works have followed a similar approach, but they ignored the semantic and topical inconsistencies between the target and auxiliary data [21].

The current state-of-the-art methods for short text are Latent Feature LDA (LF-LDA) [17] and Biterm Topic Modeling (BTM) [32]. Fig. 1 shows the generative models in plate notation to explain the differences between LDA, LF-LDA, and BTM. Shaded nodes represent observed variables, while the latent ones (i.e., the variables to be inferred) are represented by the other nodes.

The plate notation simplifies the representation of large probabilistic graphical models by placing replicated structures into numbered rectangles, where the number represents the amount of replications (e.g., rectangles numbered with M and N in LDA and LFLDA represents replicated structures for M documents and N words, respectively). In LDA, topics are defined as a distribution over vocabulary words (φ distribution), documents are mixtures of topics (θ distribution) and words (W)

are derived, one by one, from topics (according to the Z distribution of topics per word). Only words within documents are observable variables and all priors (α and β) are defined to be Dirichlet distributions. The main objective of the algorithm is to infer the hidden (or latent) variables, which are topic proportions of documents and word distributions per topic.

Latent Feature LDA (LF-LDA) is a modification of LDA that includes word vectors (ω) trained on a large external corpus. While in LDA each topic is modeled as a Dirichlet distribution, in LF-LDA the authors introduced a latent feature component (τ) for each topic. This component is a vector representation of the topic learned after each iteration of the Gibbs sampling using the maximum a posteriori estimation. The generative process of LF-LDA works as follows. For each document d, the method draws a multinomial distribution θ_d over all topics. For each ith word w_i in d, it draws a topic indicator z_i and a binary switch s_i. The topic indicator z_i defines from which topic the word w_i is to be generated, and the binary switch s_i determines whether to use the traditional Dirichlet multinomial or the latent feature component.

Biterm Topic Modeling (BTM), in contrast, learns topics by directly modeling the generation of biterms, i.e., pairs of words that co-occur in the same document. The model aggregates all corpus biterms in a big pseudo-document that is used to infer the topic distribution, overcoming the sparsity problem at a document level. Since the inference is done over one pseudo-document, the model has a single topic distribution for the entire corpus, instead of one distribution per document.

3. A framework for document expansion

The main idea behind the framework proposed in this paper is to expand short documents using new words that are similar to the ones that already appear in the document, increasing words co-occurrences, reducing sparsity and generating larger pseudo-documents. Note that we only add to the documents words already present in the collection vocabulary, as the idea is to increase the co-occurrence of semantically similar words that, due to the nature of short text, do not originally co-occur together.

Many different approaches can be used to define the words that should be added to a document. In order to make an extensible framework, we formalize the problem of document expansion by using the concept of metric space. A metric space is a set for which distances between all members of the set are defined. More formally, a metric space is a pair $(\mathcal{V}, g)$ where $\mathcal{V}$ is a set of elements and g a metric function that defines a distance between every pair of points $v_i, v_j \in \mathcal{V}$. Metric spaces guarantee that the minimum properties of distance functions are satisfied: (i) $g(v_i, v_j) \geq 0$ and $g(v_i, v_j) = 0 \iff v_i = v_j$; (ii) $g(v_i, v_j) = g(v_j, v_i)$; (iii) $g(v_i, v_j) \leq g(v_i, v_k) + g(v_k, v_j)$.

By given different definitions to $\mathcal{V}$ and g, we can generate a large set of methods to expand the documents in a collection D. In our case, the metric space will be used to define similarities between n-grams (i.e., a sequence of n words) already in the document and new n-grams.

Definition 1. Let $\mathcal{V}$ be the vocabulary, $v \in \mathcal{V}$ a n-gram belonging to the documents, and g a distance function. A t-nearest neighbor n-gram function based on v, denoted as $\mathcal{NN}(v, t)$, determines the t closest n-grams with respect to v. Formally, $\mathcal{NN}(v, t) = \mathcal{A} : |\mathcal{A}| = t \wedge \forall p \in \mathcal{A}, \forall x \in (\mathcal{V} - \mathcal{A}), g(v, p) \leq g(v, x)$.

The methods proposed here are specializations of a general framework that receives a metric space $(\mathcal{V}, g)$, and generates an intermediate graph representation G_d for each short document d based on this space.

Definition 2. Let $G_d = (L_d \cup R_d, E_d)$ be a bipartite graph representing short text d, where $L_d \cup R_d \subseteq \mathcal{V}$. G_d has two types of nodes: $l \in L_d$, which represents n-grams extracted from d, and $r \in R_d$, where R_d is the set of candidate n-grams t to expand document d. An edge $e_d = (l, r, w)$ determines the relationship between a n-gram l present in d and a candidate n-gram r with weight w. Formally, $e_d = (l, r, w) : l \in d, \ r \in \mathcal{NN}(l, t), w = g(l, r)$ and $\mathcal{NN}(l, t) \subseteq V$.

We present the general expansion framework in Algorithm 1. It has the following parameters: (i) D, the collection of documents to be expanded; (ii) $(\mathcal{V}, g)$, the metric space that contains the representation of document n-grams and the function to compare them; (iii) t, the number of neighbor words considered by g; (iv) M, the maximum expected length of a document. The metric space is the basis to build the graph of candidate words used to create the new short text representation, i.e., the pseudo-document.

Algorithm 1 General expansion framework.

```
Require: $D, (\mathcal{V}, g), t, M$
for $d \in D$ do
    if $|d| < M$ then
        $G_d \leftarrow$ Graph $(L_d \cup R_d, E_d)$ generated from $(\mathcal{V}, g)$ and $t$      ▷ Def.2
        $C_d \leftarrow \emptyset$      ▷ Candidate words
        for $e_d = (l, r, w) \in E_d$ do
            $C_d \leftarrow C_d \cup \{r, w\}$
        while $|d| < M$ do
            $h \leftarrow$ SelectionMethod$(C_d)$      ▷ Selected word
            $d \leftarrow d \cup h$
```

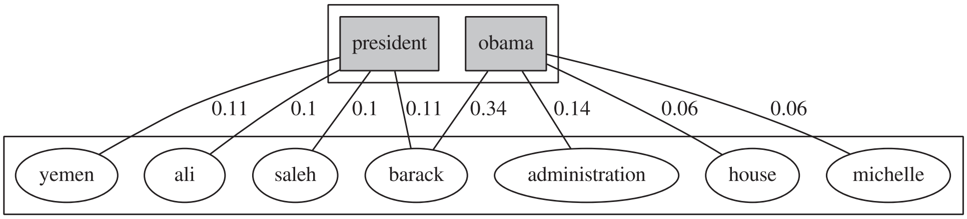


Fig. 2. Example of document distances subgraph for CoFE (“President Obama”). Each n-gram is represented by the set of documents that contains the n-gram.

The expansion procedure is performed for each document d that has less than M words (lines 1–2 from Algorithm 1). Documents with a minimum predefined size are simply ignored, and their original text kept in the pseudo-document. A similarity graph for d is generated using the given metric space, which changes according to the method being implemented (line 3). Then an initially empty set C_d accumulates, for each potential expansion n-gram r, its respective weight w (lines 4–7). Next, n-grams are probabilistic selected from C_d (without replacement) until $|d| = M$ (lines 8–10). The probability of a new n-gram r be added to d is given by the total sum of its weights in C_d (i.e., sum of its degrees).

The algorithm uses a stochastic selection process in order to guarantee an unbiased presence of terms in the final pseudo-document. In other words, the probability of selecting a word is derived from the previously created word similarity graph, and we assume a multinomial distribution over the set of candidate words. Therefore, in the selection step (line 8 of Algorithm 1), words with higher probability are more likely to be This process prevents the algorithm from raising the co-occurrence frequency of word pairs excessively, once preliminary experiments showed LDA can be very sensitive to it.

In order to show how flexible is the proposed framework, we first formalize an expansion method based on co-frequency among words, which is called *Co-Frequency Expansion* (CoFE) and was previously introduced in [19]. Following, we introduce a new method that exploits word embeddings, named *Distributed Representation-based Expansion* (DREx).

3.1. Co-frequency expansion (CoFE)

The CoFE method is simple and considers that similar words have higher likelihood of occurring in the same context. That is, the conditional probability of one word to occur in a document sliding window, given that a second word was observed, should be higher if the words are similar and lower otherwise. Note that, in this scenario, the document n-grams considered are the words themselves (i.e., we consider $n = 1$). Also note that CoFE was previously introduced in [19], and here we show how it can be rewritten by simply defining a metric space and the methods parameters. The method essence did not change, and it follows the same principles and ideas of the original version. In order to detail CoFE, we first define a metric space that exploits the co-occurrence of words.

In the defined metric space $(\mathcal{V}_{CoFE}, g_{CoFE})$, each word w of the vocabulary is represented by a set $O_w \in \mathcal{V}_{CoFE}$ that contains all documents where w occurs. We define the distance metric g_{CoFE} as:

$$g_{CoFE}(w_i, w_j) = 1 - Jaccard(w_i, w_j) = 1 - \frac{|O_i \cap O_j|}{|O_i \cup O_j|} \tag{1}$$

Having the metric space, the document expansion follows the steps listed in Algorithm 1. Given a document d, we first generate the bipartite graph $G_d = (L_d \cup R_d, E_d)$, where nodes in L_d are the words extracted from d, R_d is the set of all t nearest words of each word in L_d, and there is an edge $e(l, r, w) \in E_d$ between each node $l \in L_d$ and its t nearest nodes $r \in R_d$. As each edge represents the similarity between nodes l and r, the weight w is simply defined as the Jaccard of the words l and r.

Fig. 2 presents an example of a graph derived from CoFE’s metric space. The original document contains the text “President Obama” (represented by squares in the graph), and for each word we present the neighbor candidate words (represented by circles in the graph) considering $t = 4$. Assume that the expanded pseudo-document should contain a maximum number of words $M = 5$ after expansion. Before selecting the expansion words, we add up the weights (edges) of the candidate words connected to more than one word in the original document. For example, *Barack* is connected to both *president* and *Obama*, and hence its final weight is set to 0.45. All other words remain with the same weight, as they are connected to a single word.

The last step of Algorithm 1 is the SelectionMethod, in which we use the weights of words as their probability of appearing in the final pseudo-document. Following the example, the pseudo-document contains the original words: *president* and *obama*; and the selected words: *barack, house* and *administration*. Note that, since the selection method is stochastic, the algorithm does not necessarily select the words with the highest probabilities.

3.2. Distributed representation-based expansion (DREx)

DREx, the second method proposed and one of the main contributions of this paper, defines a metric space $(\mathcal{V}_{DREx}, g_{DREx})$ that exploits the vector representation of words to expand short text documents. Word vector representation allows objective comparison of document words regarding semantics. This is possible because the distance between two word vectors can be interpreted as a metric of semantic relationship between them, as discussed in [4,13,20].

3.2.1. Distributed representation of words

Distributed representation of words were conceived to capture semantics by coding each word and its context in a real vector-space embedding [13]. They are usually expected to be consistent with vector algebra, in the sense that some operations in the vector domain (such as sum or difference) should keep some degree of consistency with similar semantic manipulations. For example, Mikolov et al. [13] present models that produce vector representations v of words (i.e., $v(word)$) where the operation $v(\text{“king”}) - v(\text{“man”}) + v(\text{“woman”})$ outputs a vector that is close to $v(\text{“queen”})$.

Among many previous published works on fundamentals and applications of distributed word representation [4,13,20], we explored Skip-Gram (SG) [13], Continuous Bag of Words (CBoW) [13] and Global Vectors (GloVe) [20], as they are popular state-of-the-art methods for generating these representations. The Skip-Gram (SG) and Continuous Bag of Words (CBoW) models are based on artificial neural networks (ANN). While SG infers the surrounding context for a given word, CBoW infers a word given its surrounding context of size C. For SG, the ANN input is a word (one-hot encoding of size V, the vocabulary size) and the output is a set of words (each word one-hot encoded) inside a context window of size C. All words in the output layer share a weight matrix $W'_{N\times V}$ and produce a multinomial distribution using a softmax function. Word vectors of size N (a parameter that defines the size of the hidden layer) are extracted from the weight matrix $W_{V\times N}$ that connects input and hidden layers, one vector by row. Parameters are learned through backpropagation and stochastic gradient descent. For CBoW, the ANN has a similar architecture, with input and hidden layers connected by a shared matrix of weights $W_{V\times N}$, from which all word vectors of size N are also extracted, one vector per row.

The third distributed representation tested, Global Vectors (GloVe), is based on a word co-occurrence matrix. Let X be the co-occurrence matrix for the whole dataset, and $X_{i,\,j}$ the frequency in which the word j co-occurs with word i in the same observation window. $X_i = \sum_k X_{i,k}$ is the number of times word i co-occurred with the other words from the dataset in the context of i. Therefore, $P_{i,j} = P(j|i) = X_{i,j}/X_i$ is the probability of word j appear in the context of word i. The similarity between two words i and j, given the context word k, is captured by the ratio $P_{i,\,k}/P_{j,\,k}$. Let w_i, w_j and w'_k be the vector representations of word i, word j and context word k, respectively, where $w_i, w_j, w'_k \in \mathbb{R}$. GloVe aims to find a function F over word vectors and context word vectors that is proportional to $P_{i,\,k}/P_{j,\,k}$ (see Eq. (2)) under a set of constraints. For a more detailed description, see [20].

$$F(w_i, w_j, w'_k) = P_{i,k}/P_{j,k} \tag{2}$$

3.2.2. Expansion procedure of DREx

Using the distributed representation of words defined in the previous section, we define a metric space $(\mathcal{V}_{DREx}, g_{DREx})$ that exploits these vectors to expand short text documents. For DREx, the points in the metric space, that become the nodes of the graph, can be either document bigrams or expansion words. Bigrams were chosen as they are better at capturing the document context than individual words.

The set $\mathcal{V}_{DREx}$ contains a vector representation $v_w \in \mathcal{V}_{DREx}$ for each word w of the vocabulary and each bigram $w_i - w_j$ of the documents in the original dataset. We use an external dataset with larger documents to obtain the vector representation of words. For the bigrams, we exploit the arithmetic properties of vector representations and sum the word vectors for each bigram, so that each of them corresponds to an element in the metric space. Note that there are other ways to represent word vectors of bigrams, such as the average value of both vectors. However, as the arithmetic sum of vectors had already shown to be able to effectively capture document context [12], we opted for it. This additive compositionality property of word vectors is related to the learning task involved in their generation process, where each word is represented as a function of its surrounding context.

To complete the definition of our metric space, we define the distance metric between word vectors, which is a modified version of the cosine distance [34] that satisfies all properties required by the metric distance g:

$$g_{DREx}(v_i, v_j) = 1 - \frac{\cos^{-1}(v_i \cdot v_j / \|v_i\| \left\|v_j\right\|)}{\pi} \tag{3}$$

As we are interested in the angular distance between word vector pairs, Eq. (3) returns the normalized angle between the vectors, which is a formal distance metric and bounded by 0 and 1. The bound values are important, since we used them to mimic probability scores when choosing a candidate word. Fig. 3 presents a graph derived from metric space $(\mathcal{V}_{DREx}, g_{DREx})$. The original document contains the text “president obama visited cuba”, corresponding to the bigrams: (i) president+obama, (ii) obama+visited, and (iii) visited+cuba, represented by rectangles in the graph. We also present, for each bigram, the neighborhood candidate words (the circles in the graph).

Assume that the expanded pseudo-document considers $t = 3$ neighbor words for each word in the original document, and a maximum number of $M = 7$ words in total after expansion. As in CoFE, the weights of each word are inversely proportional to their distance to the original bigrams in the metric space, and the words are probabilistic selected according to these weights. Following the example, the pseudo-document contains the original words: *president, obama, visited*, and *cuba*; and the selected words: *barack, havana* and *visiting*.

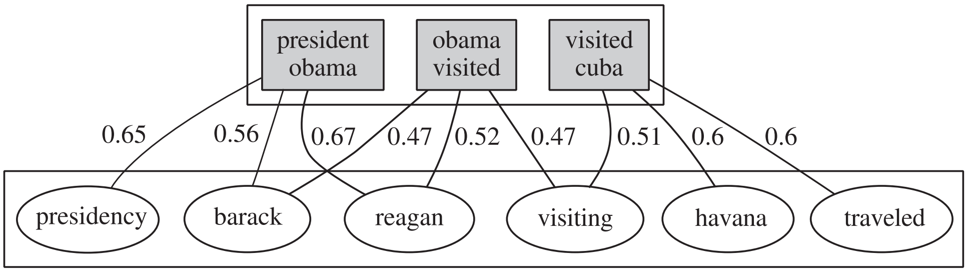


Fig. 3. Example of expansion graph for DREx ("President Obama visited Cuba"). Note that each word and bigram is actually represented in DREx by their embedded vector.

3.3. Complexity analysis

In their implementations, both CoFE and DREx have the function g of their metric spaces calculated and stored in a cache before the document expansion step starts. In terms of computational time complexity, let N be the number of documents in the dataset, V the vocabulary size, M the expected number of words in the expanded document, T the number of closest neighbors per word stored in the cache and L the dimension of the generated word vectors.

CoFE's cache generation process creates an inverted index from the dataset – $O(NV)$, considering that every document has every word in the vocabulary –, and for each pair of words in the vocabulary, it calculates their Jaccard index – $O(NV^2)$. Therefore, CoFE's cache generation time complexity is, in the worst case, of order $O(NV^2)$.

DREx's cache generation process, in turn, creates a vector representation for every bigram present in the dataset, which in the worst case is equals to $O(V^2)$ bigrams. For each bigram, its distance to the other V word vectors is calculated – $O(VL)$ calculations – and the T closest vectors are retrieved using a partial sorting of time complexity $O(V + T \log T)$. Therefore, DREx's cache generation time complexity is of order $O(V^3L + V^3 + V^2T \log T)$ in the worst case.

For the document expansion step, both methods are of order $O(NMT + NM) = O(NMT)$. For each document, T candidate words are retrieved for every word in the document, which has length $M - 1$ in the worst case, and at most $M - 1$ words are selected to be part of the new pseudo-document generated. In general, the time complexity of both algorithms is dominated by the cost of the cache generation procedure, which is highly dependent on the vocabulary size but rarely follows the worst case.

4. Experiments and results

Experiments were performed in four main phases. First we assessed the impact of varying the expected length of documents, represented by the parameter M, for both CoFE and DREx run with LDA. Based on the results of these experiments, we defined the value of M for LDA-CoFE and LDA-DREx. We then compared the results obtained in the first phase with other methods that also generate pseudo-documents to improve topic modeling for short text, namely WNTM, LDA-# and STE, introduced in Section 2. As LDA-# uses a tweet pooling scheme based on common hashtags to generate the pseudo-documents, it was only run for datasets where hashtags were available. STE expands documents with terms correlations based on PMI. Originally, the authors propose a threshold scheme based on the PMI score to determine which words should be added to the documents. Here, to make a fair comparison with CoFE and DREx, we continually add new words to the document until it reaches the target size M.

In a third phase, we explored the results of the proposed expansion methods with other topic modeling besides LDA, namely LF-LDA and BTM. Note that although these methods were conceived to deal with short text, both authors argue they should perform well in datasets with larger text [17,32]. With that in mind, we compared the results obtained by the methods using the original documents and those obtained when extracting the topics from the pseudo-documents.

Besides evaluating the methods using topic assessment metrics, in a fourth phase we measured the quality of topical document representation through a document classification task applied to the labeled datasets available. The next sections present the datasets, the metrics used in the experiments, and show the experimental results obtained.

4.1. Datasets

We used seven short text document corpus as input for the topic modeling algorithms, namely:

1. Tweets NBA (NBA): a sample of tweets about two NBA teams, Golden State Warriors and Los Angeles Lakers, collected from June to August 2015, using the hashtags #warriors and #lakers.[1]
2. Tweets Politics (Politics): a sample of tweets mentioning Democrats and Republicans, collected from June to August 2015, using the hashtags #democrats and #republicans.

[1] Available at: http://www.github.com/gabrielmip/st-topic-modeling.

Table 1
Average and standard deviation for dataset features. For the expanded collections, we consider a target of 60 words per document.

Dataset	No. of docs	Vocab. size	No. of classes	Original dataset		Expanded by CoFE	Expanded by DREx
				w/doc	unique w/doc	unique w/doc	unique w/doc
TMN	30,376	6314	7	4.9 (±1.5)	4.9 (±1.5)	59.9 (±1.2)	30.3 (±13.0)
NBA	70,707	12,504	–	8.6 (±3.0)	8.4 (±3.0)	59.5 (±0.8)	50.1 (±12.2)
Politics	70,712	15,029	–	8.1 (±2.6)	8.0 (±2.5)	59.8 (±0.5)	53.3 (±10.6)
20Nshort	1723	964	20	8.2 (±3.5)	7.1 (±2.9)	58.8 (±1.8)	26.4 (±13.3)
Sanders	3770	1311	4	6.1 (±2.7)	5.8 (±2.5)	59.7 (±0.8)	32.1 (±14.1)
Snippets	12,117	4677	8	14.3 (±4.4)	10.3 (±3.1)	55.9 (±2.9)	49.6 (±8.3)
CLEF	1001	1639	–	6.3 (±2.8)	6.0 (±2.6)	57.2 (±8.1)	25.2 (±12.5)

3. Tweets Sanders (Sanders): tweets related to four different companies: Apple, Google, Microsoft, Twitter.[2]
4. 20 Newsgroups (20Nshort): a collection of newsgroup documents, partitioned across 20 different public newsgroups. We use only the documents with less than 21 words, as done in [17].
5. Tag My News (TMN): a collection of English RSS news items grouped into 7 categories, where only the news' titles are considered [29].
6. Web Snippets (Snippets): a collection of web search snippets, which are summaries of documents presented as results of a query by a search engine [21]. The queries used are related to 8 different domains.
7. CLEF 2012 Tweet Contextualization Dataset (CLEF): tweets in English selected among informative accounts, such as CNN and TennisTweets. This dataset was created for the INEX 2012 Tweet Contextualization track at CLEF.[3]

All datasets were preprocessed before the expansion step by making all the text lower case, removing non-alphabetic characters and stop words. We also removed words shorter than 3 characters, and words appearing less than 10 times in 20Nshort and under 5 times in the TMN and Twitter datasets.

Table 1 shows statistics for the datasets. Note that we have few words per document for all datasets (column w/doc). This is also true when considering only unique words per document (column unique w/doc), which ranges from 5.82 (Sanders) to 10.27 (Snippets). DREx expansion produces, on average, documents shorter than the size defined by parameter M, despite the high standard deviation. This happens because DREx removes from the similarity graph words that are not in the target dataset vocabulary (word vectors are created from an external dataset), decreasing the number of expansion words.

4.2. Experimental setup

This section describes all the parameter configurations used by the methods considered in our experiments and the experimental setup. For DREx, we first need to obtain the vector representation of words. As mentioned before, we use the SG, CBoW and GloVe algorithms to extract the word vectors from the English Wikipedia dump from 06/02/1015. Text data were extracted from XML producing a dataset with 8,102,107 articles and a vocabulary of size 2,120,659 (also the number of word vectors). Experiments used the original implementations of SG, CBoW and GloVe. All methods used a context window of size 10. Both SG and CBOW used word vectors of size 300 and negative sampling (5 negative examples). Initial learning rate was set to 0.025 for SG and 0.05 for CBoW. For GloVe, we used the values suggested by the authors [20], fixing $x_{\max} = 100$ and $\alpha = 0.75$.

Regarding the topic models, LDA, LF-LDA and BTM share four main parameters: the number of topics (k), the hyper-parameters α and β for the Dirichlet distribution and the number of sampling iterations. The values of α and β for LDA were estimated using Minka's fixed point iteration technique [15], and LDA was run for 2000 iterations. The number of topics assumed values 20, 50 and 100. Notice that defining the number of topics of a collection is still an open problem in the literature, and the values tested are similar to the standard values used in the literature and in the original papers of the considered baselines [17,32,36]). LF-LDA has two extra parameters: the word vector representations and a mixture factor λ, which controls whether to use the Dirichlet or the latent feature component of the method. We use the default value of λ suggested by the authors (0.6) [17], and the word vectors learned from Wikipedia.

All experiments involving intrinsic evaluation of topics (i.e., all but the document classification experiment) were repeated 5 times. For all experiments, in order to verify the statistical validity of our conclusions, we used the non-parametric Wilcoxon signed-rank test with 0.05 of significance level over the means for comparison.

4.3. Evaluation metrics

Two types of evaluation were used: first we looked at the Normalized Pointwise Mutual Information (NPMI)-score [2] to evaluate the quality of the topics extracted by the methods. Then, we used a classification task to assess the topical representation of documents.

[2] Available at http://www.sananalytics.com/lab.

[3] Available at http://inex.mmci.uni-saarland.de/data/documentcollection.html#qa.

The PMI-score [16] verifies if the semantic relation between a pair of words suggested by a topic model is also found in an external dataset by evaluating the pointwise mutual information (PMI) of all pairs of its most probable words. The probabilities are evaluated by counting word co-occurrence frequencies in a 10-words sliding window in a large external dataset. Its normalized version was proposed by [2], and removes the score sensibility to frequency and provides more intuitive score values: when w_i and w_j only occur together, $\mathrm{NPMI}(w_i, w_j) = 1$; when they never occur together, $\mathrm{NPMI}(w_i, w_j)$ is defined as -1. The external dataset used for evaluation consisted of a randomly generated sample of 15M documents in English from the WMT11 news corpus.[4] We used Palmetto's NPMI implementation [25].

Given a topic t and its ten most probable words W_{10}, NPMI-score is calculated as:

$$\text{NPMI-Score}(t; W_{10}) = mean\{\mathrm{NPMI}(w_i, w_j), i, j \in 1, \ldots, 10, i \neq j\} \tag{4}$$

$$\mathrm{NPMI}(w_i, w_j) = \left(\ln \frac{p(w_i, w_j)}{p(w_i)p(w_j)}\right) / -\ln p(w_i, w_j) \tag{5}$$

For document classification evaluation we used the macro-average F1 score [33], which is the mean F1 score of all classes. The F1 score for a class is the harmonic mean between the class precision and recall. The precision of a class *c*, $Precision_c$, is defined as the fraction of correct predictions for that class, and its recall, $Recall_c$, the fraction of instances of c that were correctly predicted, as shown by the following equations:

$$Precision_c = \frac{tpr_c}{tpr_c + fpr_c} \tag{6}$$

$$Recall_c = \frac{tpr_c}{tpr_c + fnr_c} \tag{7}$$

where tpr_c is the *true positive rate* of class *c*, fpr_c its *false positive rate* and fnr_c its *false negative rate*.

The F1 score of the class *c*, $F1_c$, is defined according to the following equation:

$$F1_c = \frac{2 \times Precision_c \times Recall_c}{Precision_c + Recall_c} \tag{8}$$

4.4. Impact of parameters in CoFE and DREx

This section evaluates the performance of CoFE and DREx when the pseudo-documents generated are given as input to LDA, and analyzes the impact of the parameters k, the number of topics, and M, which defines the target document length. For k, we tested the methods with 20, 50 and 100 topics. For M, we tested maximum document sizes of 30, 40, 50 and 60. Due to the large number of experiments, Table 2 summarizes the values of NPMI by showing the average results over 20 experiments (4 values of M × 5 replications) for different number of topics. The complete tables are available as supplementary material. For DREx, we also analyzed the impact of different word vector representations (CBOW, SG and GloVe), which in the table follow the name of the method. Values in bold indicate methods that are statistically significantly better than all other methods in the same column for that dataset according to a Wilcoxon signed-rank test.

Note that, with the exception of *NBA* with 20 topics, where DREx with GloVe and SG present no statistical difference, for all other datasets and configurations the results are consistent: DREx with GloVe is always superior regardless of the number of topics. These results show that the word vector representation generated by GloVe is more robust for generating pseudo-documents than those obtained by CBoW and SG for the task of topic modeling.

Comparing DREx's performance with CoFE's, DREx always performs better, no matter which word vector representation is used. We believe the superior performance of DREx, induced by good word vectors, is related to both (i) the generality of the dataset used to generate the vectors, and (ii) its ability to exploit the input vectors. Intuitively, datasets of a specific domain (e.g., economy or tourism) are not good sources to capture the meaning of general words. These characteristics are found in the Wikipedia dataset, used to generate the word vectors used by DREx. This causes the word relations being less context-specific when compared to CoFE, which obtains word similarity information only from the original dataset.

For the number of topics, we observe that NPMI values decrease when we increase the number of topics in all configurations. This behavior may be related to the nature of the datasets and, when available, their number of classes or contexts. Since the number of contexts varies from 2 (NBA, Politics) to 20 (N20short), a very high number of topics may indeed worsen the quality of the inferred topics. Regarding the results of M, no clear pattern appears, but the best overall results were obtained by CoFE and DREx-GloVe with a maximum length of 60 (see supplementary material).

In order to provide a better insight on how our proposed methods influence LDA, we performed a comparison with topics from models trained with the Web Snippets dataset expanded by DREx-GloVe – as it presented the best values of NPMI, and CoFE. Table 3 shows the most representative words from learned topics when the number of topics is set to 20. Topics were paired using cosine similarity in a greedy strategy. A good topic should be interpretable and reflect the dataset

[4] Available at http://www.statmt.org/wmt11/training-monolingual.tgz.

Table 2
Average results of NPMI for CoFE and DREx run with LDA (mean of different values of expected document length).

	20 topics	50 topics	100 topics	20 topics	50 topics	100 topics
	Tweets NBA			Tweets Politics		
CoFE	−0.130	−0.145	−0.149	−0.099	−0.116	−0.130
DREx-CBOW	−0.073	−0.083	−0.099	−0.033	−0.064	−0.084
DREx-GloVe	−**0.033**	−**0.023**	−**0.025**	**0.020**	**0.018**	**0.006**
DREx-SG	−**0.045**	−0.045	−0.051	−0.006	−0.024	−0.041
	Tweets Sanders			N20 Short		
CoFE	−0.124	−0.143	−0.146	−0.191	−0.204	−0.205
DREx-CBOW	−0.019	−0.053	−0.078	−0.076	−0.107	−0.129
DREx-GloVe	**0.040**	**0.011**	−**0.005**	**0.010**	−**0.029**	−**0.060**
DREx-SG	−0.021	−0.051	−0.078	−0.091	−0.141	−0.166
	TMN Title			Web Snippets		
CoFE	−0.035	−0.067	−0.108	−0.024	−0.062	−0.083
DREx-CBOW	0.003	−0.023	−0.047	0.006	−0.018	−0.046
DREx-GloVe	**0.060**	**0.062**	**0.053**	**0.057**	**0.047**	**0.023**
DREx-SG	−0.041	−0.056	−0.076	−0.018	−0.029	−0.062
	CLEF2012					
CoFE	−0.138	−0.152	−0.154			
DREx-CBOW	−0.053	−0.063	−0.085			
DREx-GloVe	**0.011**	**0.001**	−**0.028**			
DREx-SG	−0.091	−0.101	−0.116			

eight categories: (1) business, (2) computers, (3) culture, arts and entertainment (4) education and science, (5) sports, (6) politics and society, (7) engineering, and (8) health.

Note that topics for both methods can be easily labeled considering the document categories. For instance, topic 1 is related to engineering, topic 2 to health and topic 3 to sports. Exceptions are topics 11 and 15, where one of the methods generated topics hard to label. For topic 11, CoFE presents words related to education and culture, while it is hard to correctly define what major concepts DREx's words are related to. In topic 15, the opposite happens. DREx presents an easier topic to categorize.

Overall, words added by DREx in the document expansion step tend to be less context-specific when compared to CoFE's. Comparing two topics related to the same subject, one may find DREx's to have more nouns that represent concepts, such as 'hardware' in topic 4, while CoFE's topic would have more specific words, such as 'cpu' and 'intel'. Similarly, in topic 10 CoFE's words include 'java' and 'code', while DREx's include 'application', 'internet' and 'information'.

4.5. Evaluating the expanded documents

Strategies that create pseudo-documents can potentially change the original meaning of the expanded document. This can occur when the document loses its original meaning, becoming skewed or almost "random". Particularly, the latter scenario can arise when we use a large value for parameter M. In this case, very short documents (e.g., documents with 2 or 3 words) with few topic discriminative words end up generating expansion graphs where edge weights may have low values and almost uniform probabilities of being selected. In this case, the expansion process may be close to random, with words unrelated to the document topic being selected.

We performed a manual analysis of the pseudo-documents generated. Here we show the results for datasets *TMN* and *Web Snippets*. As documents in these datasets are labeled according to their subjects, one would expect that CoFE and DREx would add words related to these documents categories. Table 4 shows the list of words more frequently added to the documents of the respective dataset with category label *health* and *business*. Each word is followed by the frequency it was added to documents in that dataset.

As expected, both methods added words highly correlated with the categories, weakening the hypothesis of changing the original meaning of the documents. One major difference between CoFE and DREx regards the frequency they add words to documents. Interestingly, although DREx usually generates smaller pseudo-documents (see Table 1), it often adds the same more relevant word to a larger number of documents. This set of words is also usually more general than the words added by CoFE, as mentioned in the previous section. For example, from the set of 1453 documents in category *business* of *Web Snippets*, CoFE added the word *business* to 1109, which corresponds to 1.8% of all words added by DREx in this category, while CoFE added it to 118 (0.2% of the words added by CoFE). Apart from the word *business*, the two methods only agree in other three out of the 10 most frequently added words: *finance, financial* and *information*.

Table 3
Topics discovered by LDA for Web snippets expanded by CoFE and DREx-GloVe. Column *Class* indicates the respective dataset class(es) the topic refers to. Column *Sim* indicates the cosine similarity for topics in the same row.

#	CoFE	DREx	Class	Sim
1	car engine electrical motor wheels electric cars gear fuel automatic	car engine cars equipment manufacturing electrical vehicle motor components	7	0.75
2	health cancer medical disease healthy nutrition information diet treatment hiv	health medical care treatment cancer patients disease patient medicine clinical	8	0.74
3	sports football games news game soccer com league team scores	sports football soccer league teams game basketball games sport team	5	0.74
4	intel computer memory chip processor device cpu cache core pentium	computer hardware computers intel software processor memory computing	2	0.72
5	political democracy party democratic social politics parties communist	political government politics party democratic election democracy	6	0.71
6	research edu science university school department graduate program students	university graduate edu faculty education college student students school harvard	4	0.70
7	news information online yahoo web directory com search sites links	information web online internet links external google search websites blog	2,4	0.65
8	business trade services management marketing gov development international	business industry financial market companies company investment finance	1	0.65
9	movie movies film imdb awards actor video director academy tom	music movie film video movies feature best films shows released	3	0.64
10	software programming computer web data java systems linux code parallel	computer software systems application applications internet information based	2	0.61
11	wikipedia encyclopedia wiki culture history article American ancient category	wikipedia articles article wiki https pages org page encyclopedia doesn	3,4	0.61
12	theory physics quantum philosophy theorem mathematical newton	theory theoretical analysis methods mathematical instance physics concepts	4	0.55
13	journal theoretical journals biology natural paper papers research theory evolution	research science study scientific technology studies institute development	4	0.40
14	music art rock band pop classical artists lyrics arts album	art work works gallery museum arts photo collection artist painting	3	0.40
15	amazon com books fashion online selection design book shopping manga	published book books publications journal publication literature work	4	0.35
16	system gov house government president presidential republic united congress	development public business government education economic information	6	0.27
17	war military navy force air army nuclear revolution civil weapons	culture history american world part united Europe modern america first	3,4	0.17
18	tickets tennis golf ski buy chicago grand diego maradona woods	news media coverage cnn chicago broadcast bbc interview york washington	5/3	0.17
19	market stock finance financial exchange bank investment income quotes money	food health healthy diet nutrition calorie eating fitness eat foods	1/8	0.05
20	network internet security wireless bandwidth test mobile speed access	education students teaching learning school learn work help experience	2/4	0.04

Something similar occurs to *TMN health* category, where *health* itself is the most added word by DREx. Again note that while CoFE adds more specific terms to the documents, including the name of diseases (e.g., cancer and diabetes), DREx simply adds *diseases*.

Finally, Table 5 also presents a few examples of the original *TMN* documents and the pseudo-documents generated by DREx-GloVe to illustrate the semantic agreement between the words in the original and expanded documents. We present one document for each of the eight categories of *TMN*.

4.6. Comparison with baselines

Previous results show that, when comparing CoFE and DREx run with LDA, DREx presents the best results of NPMI while expanding the documents using GloVe's vector representation. We also showed that using DREx with GloVe and setting the maximum number of words to 60 has led to pseudo-documents and topics characterized by less context-specific words. This section compares this configuration with LDA run with other approaches previously proposed which generate pseudo-documents to enhance topic extraction and also state-of-the-art methods designed specifically for topic modeling in short text. The first comparison is the one we consider our true baseline, while the second, besides showing the method generality, also shows its superior performance regardless of the topic model considered.

4.6.1. Comparison with document expansion methods

We compare the performance of DREx with LDA-# [11], which generates pseudo-documents by grouping Twitter hashtags when they are available, WNTM [36], which generates pseudo-documents using the word co-occurrence network, and STE [23], which expands documents with correlated terms according to their PMI score. As previously mentioned, we

Table 4
List of words most frequently added to documents labeled as *Health* and *Business*. For both CoFE and DREx, the target document size *m* is to 60. Each word is followed by the number of times it was added to that class documents.

CoFE	DREx-GloVe	CoFE	DREx-GloVe
Snippets			
Business		Health	
finance (150)	business (1109)	medical (133)	health (803)
financial (136)	financial (856)	health (133)	medical (646)
services (128)	industry (809)	disease (124)	care (586)
business (118)	development (712)	care (119)	research (527)
information (114)	information (683)	nih (110)	education (518)
market (113)	investment (659)	nutrition (107)	patients (499)
research (111)	management (602)	treatment (105)	treatment (495)
online (108)	companies (586)	symptoms (105)	information (484)
money (107)	finance (573)	diseases (104)	study (464)
news (106)	based (550)	research (101)	patient (421)
TMN			
Business		Health	
stocks (429)	time (1380)	study (195)	health (373)
prices (392)	financial (1296)	risk (188)	patients (344)
billion (390)	business (1120)	cancer (184)	due (338)
oil (362)	investment (1026)	heart (163)	care (321)
rise (357)	increase (977)	drug (158)	treatment (300)
profit (356)	money (924)	linked (156)	disease (299)
shares (354)	market (894)	diabetes (147)	time (294)
report (347)	make (867)	drugs (147)	medical (240)
sales (346)	due (818)	disease (132)	make (238)
higher (343)	brought (679)	kids (131)	life (233)

Table 5
Examples of TNM documents expanded by DREx-GloVe. The last column shows the selected words during the document expansion step.

Class	Original text	Words added by DREx
Business	delta air lines q1 loss grows to $318 million	flight due grown base force billion line lose grow losing jet growing service reaches aircraft lost operations makes
Entertainment	like a Rolling Stone Dylans best song	miller bob love rolling called singer rock written songs album band wall ones recording times gave live tribute features music
Health	fda to regulate e-cigarettes as tobacco products	produce smoking restricting drug approved company increase drugs manufacturing drink marijuana food smoke reduce industry alcohol regulatory
Science & Technology	apple co-founder Wozniak: computers can teach kids	company students early learn ceo computer programs allowed teaching founders dedicated read business teachers based computing studying lessons children named
Sports	Nadal cruises past Ljubicic into quarters	future open beginning rafael federer tennis djokovic day half cruise sharapova sets years runner roddick finals ferrer time days
US	Arizona supreme court stays execution	oregon cases appeals leave states case rest court appeal justice judge ruling kansas texas state months takes oklahoma courts remain nevada united california finally

adapted STE to expand the documents until they achieved a target size of M, to make a fair comparison with our proposed methods. The value of M used is the same chosen for CoFE and DREx: $M = 60$. We also show the results of LDA when run with the original documents, as a reference for comparison.[5]

Table 6 shows the values of NPMI obtained by each method. Notice that the results of LDA-# are only available for the Twitter datasets *Politics, NBA* and *CLEF*.

For *Politics*, the 70,712 original documents were grouped into 4184 pseudo-documents and the 70,702 documents of *NBA* into 3924 pseudo-documents. For *CLEF*, the 1001 original documents were grouped into 218 pseudo-documents. Note that,

[5] We implemented the referred methods LDA-#, WNTM, STE and LDA (with hyperparameter optimization) according to their description in the original papers.

Table 6
Results of NPMI for methods that generate pseudo-documents.

	20 topics	50 topics	100 topics	20 topics	50 topics	100 topics
Topic model	**Tweets NBA**			**Tweets Politics**		
LDA - Original	−0.158	−0.156	−0.154	−0.072	−0.090	−0.095
LDA-DREx-GloVe-60	−**0.037**	−**0.019**	−**0.021**	**0.024**	**0.018**	**0.006**
LDA-Hashtag	−0.158	−0.151	−0.153	−0.124	−0.116	−0.117
WNTM	−0.135	−0.141	−0.135	−0.086	−0.089	−0.099
STE	−0.087	−0.070	−0.069	−0.043	−0.045	−0.055
	Tweets Sanders			**20-News Short**		
LDA - Original	−0.087	−0.099	−0.116	−0.184	−0.188	−0.193
LDA-DREx-GloVe-60	**0.047**	**0.015**	−**0.002**	**0.009**	−**0.029**	−**0.056**
WNTM	−0.085	−0.113	−0.125	−0.194	−0.194	−0.198
STE	−0.156	−0.164	−0.170	−0.232	−0.241	−0.239
	TMN			**Web Snippets**		
LDA - Original	−0.062	−0.056	−0.085	−0.061	−0.102	−0.106
LDA-DREx-GloVe-60	**0.056**	**0.062**	**0.058**	**0.061**	**0.050**	**0.030**
WNTM	−0.026	−0.047	−0.067	0.004	−0.034	−0.064
STE	−0.245	−0.217	−0.220	−0.115	−0.141	−0.153
	CLEF2012					
LDA - Original	−0.137	−0.140	−0.135			
LDA-DREx-GloVe-60	**0.008**	**0.004**	−**0.026**			
LDA-Hashtag	−0.114	−0.125	−0.126			
WNTM	−0.149	−0.139	−0.141			
STE	−0.176	−0.178	−0.175			

after the pseudo-document generation, the total number of documents in the collections decreased drastically. Since the number of documents is as important as their size to the success of topic modeling techniques [28], this reduction may impact negatively on the results found by LDA-#. For *Sanders*, information about hashtags was not available.

Observe that the results obtained by LDA-DREx were statistically significantly better than those obtained by all baselines in all datasets. Note that LDA-# was only able to perform better than LDA with the original documents in one out of three datasets, while STE and WNTM showed improvements over LDA for two and three out of the seven datasets, respectively.

In summary, considering the datasets used in our experiments, previously proposed methods that generate pseudo-documents did not even improve the results of LDA with the original datasets in a large number of cases, while the results obtained by LDA-DREx were statistically significantly better than those obtained by both LDA and the two baselines in all cases. This shows the robustness of combining word vector representations trained in external datasets to generate improved larger pseudo-documents.

4.6.2. Comparison with other topic models developed for short text

The previous section showed that DREx's pseudo-documents generate better topics than those created by other expansion methods. This section, in contrast, compares the results of DREx with methods that have changed the LDA model to overcome the problems of short text scenarios. Two of the main representatives of this category are LF-LDA[6] and BTM[7] (see Section 2 for details). It is important to emphasize that their authors claim these methods can also be used to learn topics in datasets with larger text. Considering this and the fact that our expansion framework can be used by any topic modeling algorithm, we compare the performance of these methods using the original version of the datasets and the expanded version generated by DREx-GloVe.

Table 7 shows the values of NPMI obtained by each method followed by the percentage of improvement over the use of the original dataset. Because of the high time complexity of these methods, we only conduct experiments with 20 topics, as previous experiments showed NPMI degraded as we increased the number of topics.

We observed that, for all topic models and datasets, the pseudo-documents generated by DREx were able to improve the quality of the topics learned. The improvements range from 76% (LDA on *Tweets NBA*) to 295% (BTM on *Web Snippets*). However, note that these very high improvement values occur because the values of NPMI in the original datasets were really low.

We also highlight the fact that LDA-DREx performs better than BTM and LF-LDA with the original datasets in all cases, showing that our expansion framework can be used to overcome the sparsity problem in short text scenarios without the need of a specific method. However, note that the best overall results were obtained when using the expanded dataset with

[6] Implementation available at: https://github.com/datquocnguyen/LFTM (2016/12/16).

[7] Implementation available at: https://github.com/xiaohuiyan/BTM (2016/12/16).

Table 7
NPMI values for LDA, LF-LDA and BTM methods with 20 topics considering both the original and expanded versions (DREx-GloVe) of the dataset. Improvements of expansion are in parenthesis.

Dataset	Original	DREx-GloVe	Original	DREx-GloVe
	Tweets NBA		**Tweets Politics**	
LDA	−0.158	−**0.037 (76.58%)**	−0.072	**0.024 (133.33%)**
LF-LDA	−0.149	−**0.014 (90.60%)**	−0.059	**0.027 (145.76%)**
BTM	−0.168	−**0.036 (78.57%)**	−0.085	**0.022 (125.88%)**
	Tweets Sanders		**20-News short**	
LDA	−0.087	**0.047 (154.02%)**	−0.184	**0.009 (104.89%)**
LF-LDA	−0.079	**0.055 (169.62%)**	−0.179	**0.019 (110.61%)**
BTM	−0.085	**0.038 (144.71%)**	−0.202	**0.005 (102.48%)**
	TMN		**Web Snippets**	
LDA	−0.062	**0.056 (190.32%)**	−0.061	**0.061 (200.00%)**
LF-LDA	−0.039	**0.055 (241.03%)**	−0.061	**0.069 (213.11%)**
BTM	−0.048	**0.070 (245.83%)**	−0.042	**0.082 (295.24%)**
	CLEF2012			
LDA	−0.137	**0.008 (106.01%)**		
LF-LDA	−0.129	**0.013 (110.07%)**		
BTM	−0.149	**0.001 (110.07%)**		

Table 8
Classification results of macro-average F1 score when representing documents by topics extracted from original and expanded documents. Improvements of expansion are in parenthesis.

	LDA		BTM		LF-LDA	
	Original	DREx-GloVe	Original	DREx-GloVe	Original	DREx-GloVe
20Nshort	0.216	**0.24 (+11.1%)**	0.252	0.267 (+5.8%)	0.239	0.235 (-1.6%)
TMN	0.599	0.62 (+3.4%)	0.652	**0.689 (+5.7%)**	0.618	0.624 (+0.9%)
Sanders	0.842	**0.901 (+6.9%)**	0.88	**0.924 (+4.9%)**	0.852	**0.899 (+5.5%)**
Snippets	0.757	**0.836 (+10.4%)**	0.857	**0.872 (+1.7%)**	0.729	**0.841 (+15.4%)**

either BTM or LF-LDA. BTM presented the best results for the datasets *TMN* and *Web Snippets*, while LF-LDA was the best in the five remaining datasets.

4.7. Document classification approach

So far all experiments evaluated the topics generated with the original and expanded datasets according to NPMI. This section, in contrast, evaluates the representative power of topics to classify documents into different categories. From the seven datasets considered, four were manually labeled and took part of this experiment, namely *20-News Short, TMN, Sanders* and *Web Snippets*. The number of categories of each dataset is shown in Table 1.

In this experiment, each document was represented by its posterior topical distribution instead of their words. Therefore, the feature set f_i of a document d_i is defined as:

$$f_i = [p(z_1|d_i), p(z_2|d_i), \dots, p(z_k|d_i)]$$

Two different datasets were generated for each of the topic models considered: the first includes the topics extracted by the method from the original dataset, and the second the topics from the expanded dataset generated by DREx-GloVe. In order to compare the results of the classification task with those obtained when using the NPMI metric, both LDA, LF-LDA and BTM were used as topic models. All experiments used 20 topics as document features (i.e., $|f_i| = 20$). These datasets were given as input to a SVM classifier with a Gaussian kernel[8] to classify the documents. The document classification experiment was performed using 5 executions of a 5-fold cross-validation (1 fold for tuning SVM parameters with a grid search, 3 folds for training and 1 fold for testing), totaling 25 repetitions for each configuration. Multiclass classification was performed using the one-against-all strategy.

Table 8 shows the results of the mean macro-average F1 score comparing pairs of original and expanded datasets for each topic modeling technique, i.e., LDA, BTM and LF-LDA. Bold values indicate statistically significant best versions for each pair comparison. Results of F1 show that the proposed expansion method improves the quality of document representation (i.e., topics) from the perspective of document classification. With the exception of 20Nshort using LF-LDA, all other mean

[8] We used the R wrapper (package "e1071") for the libSVM library (https://www.csie.ntu.edu.tw/~cjlin/libsvm/).

results revealed improvements of up to 15.4%. When compared to the original dataset, a total of 8 out of 12 results (in bold) show that classification results with expanded datasets are statistically significantly better than those obtained with the topics extracted from the original datasets. The other four results show no evidence of statistical difference.

The use of topic distributions extracted by BTM and LF-LDA from the original datasets as features for short text classification revealed to be good predictors of classes. These results are consistent with those obtained in [32]. However, DREx further enhances classification performance, with statistically significant improvements that vary from 1.7 to 15.4%.

These results also reinforce the independence of the expansion method from subjacent topic modeling techniques and the potential for consistent improvement. Furthermore, intrinsic evaluation of topics, like NPMI, could raise the hypothesis of eventual misrepresentation of the original latent topic structures by the expansion procedure (even when good results are obtained). The results shown undermine this hypothesis, since if it were true this would worsen classification performance, once classes carry information about topics. Instead, classification performance was systematically improved.

In summary, we observed that for the intrinsic evaluation of topics with NPMI, the general best method in our results is LF-LDA-DREx using the GloVe representation. On the other hand, for the task of document classification, BTM-DREx with GloVe showed to be the best method for all datasets. This means that, although there are more appropriate topic modeling methods for particular tasks (e.g., BTM for document classification), DREx with Glove can generally be applied to improve the quality of topics extracted from short text.

5. Conclusions and future work

This paper introduced a framework for generating large pseudo-documents based on the definition of metric spaces, used to calculate similarities between document components. We presented two instances of the framework, namely CoFE and DREx, and showed their robustness and effectiveness in topic modeling for short text. While CoFE uses a simple word co-occurrence to expand the original text, DREx relies on distributed representation of word vectors.

The methods were evaluated in seven datasets and compared to other state-of-the-art algorithms for topic modeling in short text datasets. We first compared CoFE and DREx against each other and with other algorithms for pseudo-documents generation. The results point out that DREx outperforms all methods, achieving higher values of NPMI in all datasets. In a second phase, the collections expanded with DREx were given as input to LDA, BTM and LF-LDA, the last two being methods specifically designed to learn topics from short-texts. Finally, the methods were also evaluated in a document classification task. The experiments performed indicate that datasets expanded with DREx using GloVe's word vector representation improved significantly the results of all topic modeling methods, both in terms of NPMI and when performing text classification, showing that the proposed expansion framework can be used to enhance the quality of the topics found by any topic modeling algorithm.

As future work, we intend to explore other metric spaces for the framework, considering other document components. Trying out other ways to combine word vectors for bigrams and even extend it to higher sets of words is another interesting direction. Finally, a qualitative evaluation of the topics would also benefit the comparisons, although preliminary results already point out the differences between the proposed expansion methods.

Acknowledgments

The authors would like to thank CAPES, CNPq and FAPEMIG-PRONEX-MASWeb under Proc. APQ-01400-14., CEFET-MG under Proc. PROPESQ-10314/14, W3C/NIC.br and SERPRO, for their financial supp ort.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ins.2017.02.007 .

References

[1] D. Blei, A. Ng, M. Jordan, Latent Dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.

[2] G. Bouma, Normalized (pointwise) mutual information in collocation extraction, in: Proceedings of GSCL, 2009, pp. 31–40.

[3] C. Carpineto, G. Romano, A survey of automatic query expansion in information retrieval, ACM Comput. Surv. 44 (1) (2012) 1:1–1:50.

[4] M. Faruqui, Y. Tsvetkov, D. Yogatama, C. Dyer, N.A. Smith, Sparse overcomplete word vector representations, in: Proceedings of ACL, 2015.

[5] L. Gao, S. Zhou, J. Guan, Effectively classifying short texts by structured sparse representation with dictionary filtering, Inf. Sci. 323 (2015) 130–142.

[6] L. Hong, B.D. Davison, Empirical study of topic modeling in Twitter, in: Proceedings of the First Workshop on Social Media Analytics, ACM, 2010, pp. 80–88.

[7] E. Hörster, R. Lienhart, M. Slaney, Image retrieval on large-scale image databases, in: Proceedings of CIVR, 2007, pp. 17–24.

[8] A. Hotho, S. Staab, G. Stumme, Ontologies improve text document clustering, in: Proceedings of the Third IEEE International Conference on Data Mining, IEEE, 2003, pp. 541–544.

[9] O. Jin, N.N. Liu, K. Zhao, Y. Yu, Q. Yang, Transferring topical knowledge from auxiliary long texts for short text clustering, in: Proceedings of CIKM, 2011, pp. 775–784.

[10] C.D. Manning, H. Schütze, Foundations of Statistical Natural Language Processing, MIT Press, Cambridge, MA, 1999.

[11] R. Mehrotra, S. Sanner, W. Buntine, L. Xie, Improving LDA topic models for microblogs via tweet pooling and automatic labeling, in: Proceedings of SIGIR, 2013, pp. 889–892.

[12] T. Mikolov, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proceedings of NIPS, 2013b, pp. 1–9.

[13] T. Mikolov, G. Corrado, K. Chen, J. Dean, Efficient estimation of word representations in vector space, in: Proceedings of ICLR, 2013a, pp. 1–12.

[14] T. Mikolov, W. tau Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: Proceedings of PNAACL-HLT, 2013.

[15] T. Minka, Estimating a Dirichlet distribution, Technical report, M.I.T (2000) 1–13.

[16] D. Newman, Y. Noh, E. Talley, S. Karimi, T. Baldwin, Evaluating topic models for digital libraries, in: Proceedings of JCDL, 2010, pp. 215–224.

[17] D.Q. Nguyen, R. Billingsley, L. Du, M. Johnson, Improving topic models with latent feature word representations, Trans. Assoc. Comput. Linguist. 3 (2015) 299–313.

[18] A. Pal, A. Herdagdelen, S. Chatterji, S. Taank, D. Chakrabarti, Discovery of topical authorities in instagram, in: Proceedings of the 25th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2016, pp. 1203–1213.

[19] G. Pedrosa, M. Pita, P. Bicalho, A. Lacerda, G.L. Pappa, Topic modeling for short texts with co-occurrence frequency-based expansion, in: Proceedings of the Brazilian Conference on Intelligent Systems, IEEE, 2016.

[20] J. Pennington, R. Socher, C.D. Manning, GloVe: global vectors for word representation, in: Proceedings of EMNLP, 2014, pp. 1532–1543.

[21] X.-H. Phan, L.-M. Nguyen, S. Horiguchi, Learning to classify short and sparse text & web with hidden topics from large-scale data collections, in: Proceedings of WWW, 2008.

[22] P. Pinoli, D. Chicco, M. Masseroli, Latent Dirichlet allocation based on Gibbs sampling for gene function prediction, in: Proceedings of the IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, IEEE, 2014, pp. 1–8.

[23] D. Pinto, P. Rosso, H. Jiménez-Salazar, A self-enriching methodology for clustering narrow domain short texts, Comput. J. 54 (7) (2011) 1148–1165.

[24] P. Rafeeque, S. Sendhilkumar, A survey on short text analysis in web, in: Proceedings of the Third International Conference on Advanced Computing, IEEE, 2011, pp. 365–371.

[25] M. Röder, A. Both, A. Hinneburg, Exploring the space of topic coherence measures, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, ACM, 2015, pp. 399–408.

[26] P. Rosso, M. Errecalde, D. Pinto, Analysis of short texts on the web: introduction to special issue, Lang. Resour. Eval. 47 (1) (2013) 123.

[27] J. Sedding, D. Kazakov, Wordnet-based text document clustering, in: Proceedings of the 3rd Workshop on Robust Methods in Analysis of Natural Language Data, Association for Computational Linguistics, 2004, pp. 104–113.

[28] J. Tang, Z. Meng, X. Nguyen, Q. Mei, M. Zhang, Understanding the limiting factors of topic modeling via posterior contraction analysis, in: Proceedings of ICML, 2014, pp. 190–198.

[29] D. Vitale, P. Ferragina, U. Scaiella, Classification of short texts by deploying topical annotations, Advances in Information Retrieval, Springer, 2012.

[30] J. Weng, E.-P. Lim, J. Jiang, Q. He, Twitterrank: finding topic-sensitive influential twitterers, in: Proceedings of WSDM, 2010, pp. 261–270.

[31] L. Wenyin, X. Quan, M. Feng, B. Qiu, A short text modeling method combining semantic and statistical information, Inf. Sci. 180 (20) (2010) 4031–4041.

[32] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proceedings of WWW, 2013, pp. 1445–1456.

[33] Y. Yang, X. Liu, A re-examination of text categorization methods, in: Proceedings of the 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval, ACM, 1999, pp. 42–49.

[34] J. Zhang, R.R. Korfhage, A distance and angle similarity measure method, J. Assoc. Inf. Sci. Technol. 50 (9) (1999) 772.

[35] W.X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, X. Li, Comparing Twitter and traditional media using topic models, in: Proceedings of the Advances in Information Retrieval, 2011, pp. 338–349.

[36] Y. Zuo, J. Zhao, K. Xu, Word network topic model: a simple but general solution for short and imbalanced texts, Knowl. Inf. Syst. (2015) 1–20.