Seed-guided topic model for document filtering and classification

Li, Chenliang; Chen, Shiqian; Xing, Jian; Sun, Aixin; Ma, Zongyang

2018

Li, C., Chen, S., Xing, J., Sun, A., & Ma, Z. (2018). Seed-guided topic model for document filtering and classification. ACM Transactions on Information Systems, 37(1), 9-. doi:10.1145/3238250

https://hdl.handle.net/10356/142845

https://doi.org/10.1145/3238250

A

Seed-Guided Topic Model for Document Filtering and Classification

CHENLIANG LI, Wuhan University
SHIQIAN CHEN, Wuhan University
JIAN XING, Hithink RoyalFlush Information Network Co, Ltd
AIXIN SUN, Nanyang Technological University
ZONGYANG MA, Baidu Inc.

One important necessity is to filter out the irrelevant information and organize the relevant ones into meaningful categories. However, developing text classifiers often requires a large number of labeled documents as training examples. Manually labeling documents is costly and time-consuming. More importantly, it becomes unrealistic to know all the categories covered by the documents beforehand. Recently, a few methods have been proposed to label documents by using a small set of relevant keywords for each category, known as *dataless text classification*. In this paper, we propose a seed-guided topic model for the dataless text filtering and classification (named DFC). Given a collection of unlabeled documents, and for each specified category a small set of seed words that are relevant to the semantic meaning of the category, DFC filters out the irrelevant documents and classifies the relevant documents into the corresponding categories through topic influence. DFC models two kinds of topics: *category-topics* and *general-topics*. Also, there are two kinds of category-topics: relevant-topics and irrelevant-topics. Each relevant-topic is associated with one specific category, representing its semantic meaning. The irrelevant-topics represent the semantics of the unknown categories covered by the document collection. And the general-topics capture the global semantic information. DFC assumes that each document is associated with a single category-topic and a mixture of general-topics. A novelty of the model is that DFC learns the topics by exploiting the explicit word co-occurrence patterns between the seed words and regular words (*i.e.,* non-seed words) in the document collection. A document is then filtered, or classified, based on its posterior category-topic assignment. Experiments on two widely used datasets show that DFC consistently outperforms the state-of-the-art dataless text classifiers for both classification with filtering and classification without filtering. In many tasks, DFC can also achieve comparable or even better classification accuracy than the state-of-the-art supervised learning solutions. Our experimental results further show that DFC is insensitive to the tuning parameters. Moreover, We conduct a thorough study about the impact of seed words for existing dataless text classification techniques. The results reveal that it is not using more seed words, but the document coverage of the seed words for the corresponding category that affects the dataless classification performance.

CCS Concepts:•**Information systems** → **Document topic models; Clustering and classification;**

General Terms: Algorithms, Management, Experimentation

Additional Key Words and Phrases: Topic Model, Dataless Classification, Document Filtering

ACM Reference Format:
Chenliang Li, Shiqian Chen, Jian Xing, Aixin Sun, and Zongyang Ma, 2018. Seed-Guided Topic Model for Document Filtering and Classification. *ACM Trans. Inf. Syst.* V, N, Article A (January YYYY), 36 pages.
DOI: 0000001.0000001

1. INTRODUCTION

With the advance of the Information Technology, the tremendous amounts of textual information generated everyday is far beyond the scope that people can manage manually. The recent prevalence of social media further exacerbates this information overload, because rich information about various

---

This paper is an extended version of the paper [Li et al. 2016b] presented at the 25th International ACM CIKM conference (Indianapolis, USA, Oct 24-28, 2016).

Author's addresses: C. Li (corresponding author), S. Chen, State Key Lab of Software Engineering, Computer School, Wuhan University, China 430072; J. Xing, Hithink RoyalFlush Information Network Co, Ltd, Hangzhou, China; A. Sun, School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798; Z.Ma, Baidu Inc. ShenZhen, China 518000.

DOI: 0000001.0000001

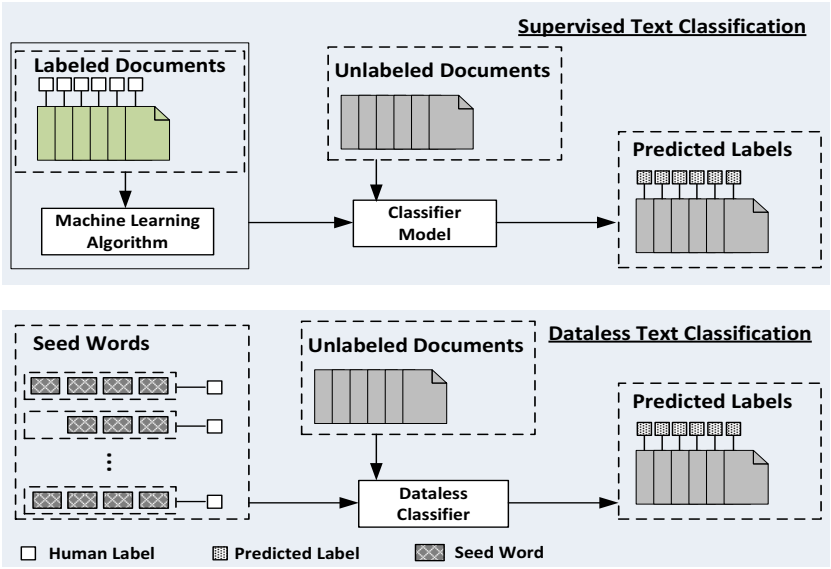


Fig. 1. Supervised vs dataless text classification.

kinds of events, user opinions and daily life activities are generated in an unprecedented speed. These timely information breeds the new and dynamic information needs everywhere. For example, a data analyst might need to track an emerging event by using a few relevant keywords [Ritter et al. 2015]. A data-driven company often needs to conduct a focused and deep analysis on the documents of the specified categories. Within these semantic applications, one fundamental task is to filter out irrelevant information and organize relevant information into meaningful topical categories.

During the last decade, text classifiers have become important tools in managing and analyzing large document collections. Text classification refers to the task of assigning category labels to documents based on their semantics. Due to its wide usage, text classification has been studied intensively for many years. Existing solutions are mainly based on supervised learning techniques which require tremendous human effort in annotating documents as labeled examples, as shown in the upper part of Figure 1. To reduce the labeling effort, many semi-supervised algorithms have been proposed for text classification [Chang et al. 2008; Nigam et al. 2000]. Considering the diversity of the documents in many applications, constructing a relatively small training set required by the semi-supervised algorithms remains very expensive. Recently, a number of *dataless text classification* methods have been proposed [Chang et al. 2008; Chen et al. 2015; Downey and Etzioni 2008; Druck et al. 2008; Gliozzo et al. 2009; Hingmire and Chakraborti 2014; Hingmire et al. 2013; Li et al. 2016b; Liu et al. 2004; Song and Roth 2014]. Instead of using labeled documents as training examples, dataless methods only require a small set of relevant words for each category or labeling the topics learned from a standard LDA model [Blei et al. 2003], to build text classifiers. As illustrated in the lower part of Figure 1, dataless classifiers do not require labeled documents, which saves a lot of human efforts. It has been reported that a speed-up of up to 5 times can be achieved to build a dataless text classifier with indistinguishable performance to a supervised classifier, by assuming that labeling a word is 5 times faster than labeling a document [Druck et al. 2008]. These promising results suggest that dataless text classification is a practical alternative to the supervised approaches, when constructing the training documents is not an easy task. More importantly, the labeled documents produced by a dataless classifier can also be used as training examples to learn supervised text classifiers if necessary [Li et al. 2016b]. However, these existing dataless classification techniques do not consider *document filtering*. That is, we like to retrieve all the documents relevant to a specified

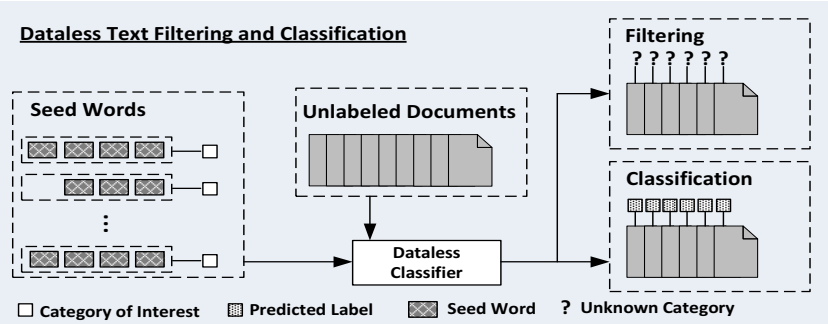


Fig. 2. Illustration for dataless filtering and classification.

set of categories from a given document collection, and organize these relevant documents into the corresponding categories. With the existing dataless classifiers, we need to provide all the categories and the corresponding seed words covered by the document collection. Unfortunately, it is often unrealistic to foresee all possible categories covered by a document collection, since the documents streamed in are likely to cover dynamic topics. In an extreme case, the number of possible categories covered by documents could be potentially limitless.

In this study, we aim to devise a dataless algorithm for the task of *filtering and classifying documents* into categories of interest. Figure 2 provides an illustration for this dataless filtering and classification task. Specifically, given a document collection of D documents and C categories of interest, where each category c is defined by a small set of seed words $\mathbb{S}_c$, the task is to filter out the documents irrelevant to any of the C categories, and to classify the relevant documents into C categories, without using any labeled documents. We also call this task as *dataless classification with filtering*.

Human beings can quickly learn to distinguish whether a document belongs to a category, based on several relevant keywords about a category. This is because that people can learn to build the relevance among the representative words of the category. For example, a human being can successfully identify a relevant word "wheel" to category *automobile*, after browsing several documents in category *automobile*, even if she does not know the meaning of word "wheel". The underlying reason is the high co-occurrence between "wheel" and other relevant words like "cars" and "engines". This relevance learning process is analogous to the unsupervised topic inference process of the standard LDA [Blei et al. 2003], a probabilistic topic model (PTM) that implicitly infers the hidden topics from the documents based on the higher-order word co-occurrence patterns [Thomas and Mark 2004]. However, conventional PTMs like PLSA and LDA are unsupervised techniques that implicitly infer the hidden topics based on word co-occurrences [Blei et al. 2003; Hofmann 1999]. It is difficult or even infeasible to filter and classify documents in such a purely unsupervised manner.

Inspired by the recent success of the PTM-based dataless text classification techniques [Chen et al. 2015; Hingmire and Chakraborti 2014; Hingmire et al. 2013; Li et al. 2016b], in this paper, we propose a seed-guided topic model for **d**ataless text **f**iltering and **c**lassification, named DFC. Given a collection of unlabeled documents, DFC is able to achieve the goal of filtering and classification by taking only a few semantically relevant words for each category of interest (called "seed words"). To enable document filtering , we model two sets of category-topics: *relevant-topics* and *irrelevant-topics*. A one-to-one correspondence between relevant-topics and categories of interest is made. That is, each relevant-topic is associated with one specific category of interest, and vice versa. The relevant-topic is assumed to represent the meaning of that category[1]. Since the documents relevant to the categories of interest could comprise a limited proportion of the whole collection, irrelevant-topics are expected to model other categories covered by the irrelevant documents.

[1]Category and relevant-topic are considered equivalent and exchangable in this work when the context has no ambiguity.

In our earlier work [Li et al. 2016b], a topic model based dataless classification technique (named STM) is proposed by using a set of general-topics to model the general semantics of the whole document collection. Although the modeling of general and specific aspects of documents was studied previously for information retrieval [Chemudugunta et al. 2006], it had been overlooked for dataless text classification in previous studies [Chen et al. 2015; Hingmire and Chakraborti 2014; Hingmire et al. 2013]. Our earlier work has proven that this model setting is beneficial for dataless classification performance. Following this modeling strategy, in DFC, we also utilize a set of general-topics to represent the general semantic information. The task of filtering and classification is achieved by associating each document with a single category-topic[2] and a mixture of general-topics. The posterior category-topic assignment is then used to label the document as a category of interest, or an irrelevant one. DFC also subsumes STM under some particular parameter settings. This means that DFC is also able to conduct dataless text *classification without filtering*.

Seeking useful supervision from the seed words to precisely infer category-topics and general-topics is vital to the efficacy of DFC. In other words, precise relevance estimation between a word and a category-topic via a small set of seed words is crucial for DFC. It is noteworthy to underline that no seed word can be provided for any irrelevant-topic, because the possible semantic categories covered by a document collection are unknown beforehand. However, the precise inference for the irrelevant-topics is an essential factor for classification performance guarantee. Without any supervision from the corresponding seed words, the model is hard to identify the irrelevant-topics successfully, leading to inferior classification performance. Here, we devsie a simple but effective mechanism to identify a set of pseudo seed words for each irrelevant-topic. Specifically, we resort to using standard LDA to extract the hidden topics for the document collection in an unsupervised manner. Then, a relevance measure is proposed to calculate the distance between each LDA hidden topic and all the seed words provided for the relevant-topics. After a heuristic procedure to filter out noisy LDA hidden topics, the top topical words from the least relevant LDA hidden topics are then considered as the pseudo seed words for the irrelevant-topics.

In contrast to the existing dataless classification methods that simply exploit the semantic guidance provided by the seed words in an implicit way (*i.e.,* the word co-occurrence information), we adopt an explicit strategy to estimate the relevance between a word and a category-topic, and also the initial category-topic distribution for each document. The estimated relevance is then utilized to supervise the topic learning process of DFC. In particular, we investigate two mechanisms (*i.e.,* Doc-Rel and Topic-Rel) to estimate the probability of a word being generated by a category-topic, by measuring its correlations to the (pseudo) seed words of that category-topic based on either document-level word co-occurrence or topical-level word co-occurrence information. We call the words that are generated by a category-topic *category words*.

In summary, DFC conducts the document filtering and classification in a weakly supervised manner, just as what humans do in learning to classify documents with just few words: (i) first to identify the highly relevant documents based on the given seed words of a category; (ii) then based on these highly relevant documents, to collectively identify the category words in addition to the seed words; (iii) next to use both the seed words and category words to find new relevant documents and new category words; the last step repeats until a global equilibrium is optimized. We conduct extensive experiments on two datasets Reuters-10 and 20-Newsgroup, and compare DFC with state-of-the-art dataless text classifiers and supervised learning solutions. In terms of classification accuracy measured by F_1, our experimental results show that DFC outperforms all the dataless competitors in almost all the tasks and performs better than the supervised classifiers sLDA and SVM in many tasks for both classification with filtering and classification without filtering. We also conduct a comprehensive performance evaluation to analyze the impact of parameter settings in DFC. The results show that the proposed DFC is reliable to a broad range of parameter values, indicating its superiority in real scenarios.

[2]A category-topic refers to either a relevant-topic or an irrelevant-topic in this work when the context is for DFC.

We need to emphasize that all existing dataless text classification techniques rely solely on the weak supervision provided by the seed words. That is, the quality of seed words plays a crucial factor regarding the classification performance. It is intuitive that fewer seed words could carry less semantic information for the classification. However, the study on seed words in the paradise of dataless text classification is still missing. Here, an open question naturally arises: *will using more seed words lead to better classification accuracy?* If the answer to this question is *no*. Then, *what criterion should we use to build a set of seed words for a category?* In this work, we conduct a thorough study with the aim of answering these two questions. We find that using more seed words may not produce better classification accuracy. Also, we empirically observe that more relevant documents covered by a set of seed words under a category, better classification accuarcy can be obtained. In an extreme case, given two seed words for a category, no performance gain could be obtained by using both seed words over using either one, if the two words always appear together in documents. That is, it is not the number of seed words that matters, but the document coverage for that category. In summary, the main contributions of this paper are listed as follows:

(1) We propose and formalize a new task of dataless text filtering and classifcation. To the best of our knowledge, this is the first work to classify documents into relevant categories of interest, and filter out irrelevant documents in a dataless manner. To enable precise inference of irrelevant-topics, we propose a novel mechanism to identify the pseudo seed words for irrelevant topics in an unsupervised manner.
(2) DFC does not solely rely on the implicit word co-occurrence patterns to guide the category inference process. Instead, we introduce two mechanisms to estimate the probability of a word being generated by a category-topic. The estimation is based on the explicit word co-occurrence patterns derived from the document collection.
(3) We empirically study the impact of seed words for dataless text classification techniques. Our results suggest that using more seed words may not lead to better classification accuracy. Instead, the document coverage of the selected seed words correlates positively with the classification accuracy.
(4) We conduct extensive experiments to evaluate the proposed DFC on two real-world text datasets. The results demonstrate that DFC achieves promising classification and filtering performance, and outperform the existing dataless alternatives. Moreover, compared with supervised classifiers (sLDA and SVM), DFC even achieves better performance in a few tasks.

2. RELATED WORK

Here, we mainly review related work on dataless text classification and topic modeling with auxiliary knowledge.

2.1. Dataless Text Classifiers.

As being the seminal work of dataless text classification, Liu *et al.* investigated the possibility of building a text classifier by simply employing few words relevant to each category in a semi-supervised manner, where these relevant words are used to bootstrap an initial set of training instances [Liu et al. 2004]. Then a semi-supervised naive Bayes classifier based on the Expectation Maximization algorithm (NB-EM) [Nigam et al. 2000] is built based on the training instances. Similarly, Gliozzo *et al.* [Gliozzo et al. 2009] proposed to build an initial set of training instances by using the Latent Semantic Analysis [Deerwester et al. 1990]. Then a support vector machine (SVM) classifier is trained based on these bootstrapped instances. Downey and Etzioni provided a theoretical analysis about the possibility of achieving accurate classification in the absence of training data [Downey and Etzioni 2008]. Their analysis and empirical studies showed that the accurate text classification without the training data is possible under certain assumptions. Druck *et al.* proposed a maximum entropy based dataless text classifier which uses only the labeled words of each category, named GE-FL [Druck et al. 2008]. GE-FL was designed by assuming that the documents containing the seed words of a category are more likely to belong to this category. Hence, the pa-

rameters of GE-FL are optimized by minimizing the distance between the expected category distribution of the documents containing a labeled word under GE-FL and the corresponding reference category distribution of the labeled word. They showed that a speed-up of 5 times can be achieved by GE-FL with indistinguishable performance to an entropy regularization based semi-supervised (ER) method [Grandvalet and Bengio 2004], given labeling a word is 5 times faster than labeling a document [Raghavan et al. 2006].

Chang *et al.* proposed a dataless text classification method by projecting each word and document into the same semantic space of Wikipedia concepts [Chang et al. 2008]. They represent each category with the words used in the category label. The similarity between a document and a category is measure by using Explicit Semantic Analysis (ESA) [Gabrilovich and Markovitch 2007]. Recently, Song and Roth [Song and Roth 2014] studied the task of dataless hierarchical text classification by applying the work of [Chang et al. 2008]. Their experimental results showed that Wikipedia-based ESA still performs the best for this task. Since the large-scale knowledge base like Wikipedia is not always available for many languages or domains, this method may not be applicable in these cases. Note that the proposed DFC does not rely on the external knowledge base at all. Instead, DFC learns the discriminative category information by exploiting the semantic relevance of the seed words to the dataset itself, which can be applied in a much broader range of scenarios.

Several methods based on the standard LDA have been proposed for dataless text classification [Chen et al. 2015; Hingmire and Chakraborti 2014; Hingmire et al. 2013]. Hingmire *et al.* proposed a dataless text classifier model based on the LDA, named ClassifyLDA [Hingmire et al. 2013]. ClassifyLDA first infers the hidden topics by using LDA. Then, an annotator assigns a category to each topic. ClassifyLDA continues the topic inference process by aggregating the topics with the same category label as a single topic. The corresponding category of the topic with the maximum posterior topic proportion is used as the prediction. They showed that ClassifyLDA achieves almost comparable performance with a semi-supervised naive Bayes classifier (NB-EM) proposed in [Nigam et al. 2000]. Hingmire and Chakraborti [Hingmire and Chakraborti 2014] proposed a new model (TLC) by extending ClassifyLDA, which allows to assign more than one category to a topic. Then, TLC was further enhanced to incorporate the relevant words of each category, called TLC++. TLC++ selects the most informative words by using the information gain metric based on the initial category predictions from TLC. They found that TLC++ consistently outperforms ClassifyLDA, TLC and GE-FL by a comprehensive evaluation.

Chen *et al.* proposed a LDA based dataless classification model, called DescLDA [Chen et al. 2015]. DescLDA assumes that each category is associated with a fixed number of topics, and each topic is only associated with a single category. The selected semantically related words (called descriptive words) for each category are used to constrain the topic-word distribution such that these words have a higher probability under the associated topics. DescLDA has a tuning parameter, *i.e.,* the topic number for a category. However, DescLDA is very sensitive to this parameter. According to their experimental results, a significant performance degradation is experienced when a suboptimal number is used across the both datasets used in their work. Our earlier work proposes a seed-guided topic model for dataless text classification, named STM [Li et al. 2016b]. In STM, two separate sets of topics are modeled: *category-topics* and *general-topics*. While category-topics are responsible for extracting the discriminative category information, general-topics are used to organize the general semantic information underlying the whole collection. The experimental results demonstrated that STM outperforms DescLDA, TLC++, GE-FL, and a supervised topic model for classification (sLDA), and is more robust than these competitors. They also showed that STM even achieves very close or better performance than SVM in many tasks. As built on the basis of STM, DFC is also very robust to the parameter settings. Our experimental results show that little performance variations are observed for different parameter settings across different datasets.

Our proposed DFC differs significantly from the above PTM-based solutions. While these methods can classify the documents into their corresponding categories without any training document, they can only be applied for classification without filtering. That is, we need to provide the seed words

for all the categories covered by the document colleciton, which is unrealistic in many real-world applications.

2.2. Topic Models with Auxiliary Knowledge.

Different kinds of prior domain knowledge have been incorporated into PTMs to achieve better performance of different tasks. Mimno *et al.* exploited the corpus-specific word co-occurrence information to enhance the topic coherence of the standard LDA [Mimno et al. 2011]. Besides exploiting the corpus-specific knowledge, many works have proposed to incorporate the semantical relations between word pairs into the topic model [Andrzejewski et al. 2009; Chen and Liu 2014; Chen et al. 2013; Li et al. 2016a]. The semantic relatedness information based on the learnt word embeddings over the large external corpus is incorporated for better short text topic modeling in [Li et al. 2017, 2016a]. A seeded topic model was proposed to extract the aspects and sentiments from the customer comments in [Mukherjee and Liu 2012], named SAS. SAS takes the seed words related to a specific aspect as a seed set, *e.g.,* words related to the aspect *room service*. Then an aspect is considered as a multinomial distribution over the non-seed words and the seed sets. Based on the implicit word co-occurrence information regarding these seed words, SAS can obtain a significant improvement in terms of aspect extraction accuracy. Similarly, Jagarlamudi *et al.* proposed a SeededLDA model to learn better topic-word and document-topic distributions with the seed words selected by using information gain from the labeled documents [Jagarlamudi et al. 2012]. These works exploit the semantic guidance provided by the seed words in an implicit way, *i.e.,* the word co-occurrence information. Differing from these works, we employ an explicit strategy to estimate the initial document relevance and discriminate relevant words of a specific category. These prior knowledge extracted based on the seed words are then used directly to guide the topic inference process, leading to a promising classification performance. Later in Section 4.4, we will show that this strategy indeed brings significant improvement to the classification performance.

Since our work exploits the seed words and the word co-occurrence information to derive the category of each document in a collective manner, the underlying motivation is strongly connected to the pseudo relevance feedback, a widely studied strategy for information retrieval enhancement[Buckley and Salton 1995; Dunlop 1997; Espinosa and Akella 2012; Miao et al. 2016; Ye and Huang 2014]. The aim of pseudo relevant feedback is to expand the query with relevant words covered by the top-retrieved documents in the first pass. The techniques presented in the existing literature mainly count the word frequency in these highly relevant documents regarding the original query. Hence, the words highly co-occur with the query terms are considered to be relevant and extracted as the complement. Several works also incorporate topical information to guide the query expansion process. Caballero and Akella [Espinosa and Akella 2012] incorporate topic-based language model to extract the relevant words. Miao *et al.* [Miao et al. 2016] proposed to weight the top-retrieved documents in terms of their proximity in the latent topic space. Here, our work introduce two kinds of topics: category-topics and general-topics. The word co-occurrence information and the provided seed words are used together to build the prior knowledge (*i.e.,* category word probability). This prior konwledge is then used to supervise the topic inference process. By taking the seed words as the query, the category word probability estimation and topic inference can be considered as a query expansion process, but in a probabilistic manner. That is, the most probable words under a category-topic reflect the meaning of the corresponding category, in addition to the seed words.

3. SEED-GUIDED TOPIC MODEL

In this section, we present the proposed DFC model for dataless text filtering and classification in detail.

3.1. Generative Process and Inference

Unlike the existing related works that directly use a topic to represent a distinct category [Chen et al. 2015; Hingmire and Chakraborti 2014; Hingmire et al. 2013], we assume that there are three kinds of topics underlying the document collection: relevant-topics, irrelevant-topics and general-

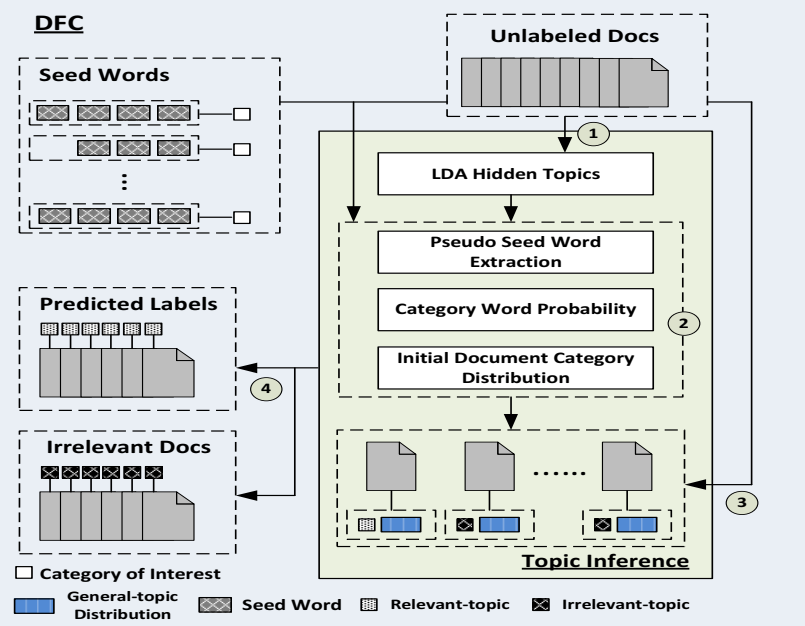


Fig. 3. The architecture of DFC.

topics. As in PLSA and LDA, general-topics in DFC are shared by all the documents and capture the global semantic information of the whole collection. In contrast, each relevant-topic is associated with a single category and be comprised by the relevant words of the category. Also, each category has only one relevant-topic. That is, the category and its related relevant-topic have a one-to-one correspondence in DFC. Since the categories of interest could only cover a part of all documents in the collection, we use irrelevant-topics to cover the semantics of the irrelevant documents. In this sense, we call both relevant-topics and irrelevant-topics as category-topics. In DFC, each document is generated by a category-topic and all general-topics together.

Since general-topics are used in DFC to model the semantic information besides the category information encoded by the category-topics, it is expected that the documents of the same category could share similar document general-topic distributions to some extent. Consequently, we associate each category-topic in DFC with a mixture of general-topics. Let φ_c denotes the general-topic distribution associated with category-topic c, θ_d denotes the general-topic distribution of document d. We assume the general-topic distribution θ_d is sampled from a Dirichlet prior with the concentration parameter α_1 and the category's φ_c as the base measure. This hierarchical Dirichlet prior setting allows the flexibility that the deviation of general-topic distributions of the documents under the same category is controlled by the concentration parameter α_2 [Wallach et al. 2009]. Given R specified categories (*i.e.,* relevant-topics), T irrelevant-topics and B general-topics, a word in a document can be either generated from category-topic c or a general-topic b. The document is an irrelevant document when category-topic c refers to an irrelevant-topic. Otherwise, the document belong to the corresponding category indicated by relevant-topic c. The main notations used in the rest of the paper are summarized in Table I. The graphical representation of DFC is shown in Figure 4 and the generative process is described as follows:

1 For each relevant-topic $r \in \{1...R\}$:
. (a) draw a general-topic distribution $\varphi_r \sim Dir(\alpha_0)$;
. (b) draw a word distribution $\vartheta_r \sim$ Dir(β_0);
2 For each irrelevant-topic $t \in \{1...T\}$:

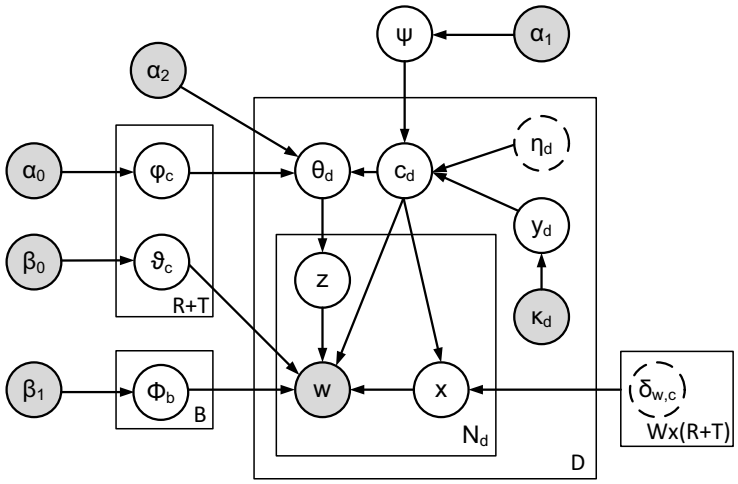


Fig. 4. Graphical representation of DFC. Since the variables η_d and $\delta_{w,c}$ are estimated based on the (pseudo) seed words of DFC, we therefore plot these two sets of variables in dotted circle.

Table I. List of Notations.

D	the total number of documents in the dataset
R	the total number of specified categories/relevant-topics
T	the total number of irrelevant-topics in the dataset
B	the total number of general-topics
W	the size of the vocabulary
s	a seed word of a category
$\mathbb{S}_c$	a set of seed words of category c
c_d	the category-topic assignment for document d
$w_{d,i}$	the observed word at position i in document d
$z_{d,i}$	the general-topic assignment for word $w_{d,i}$
$x_{d,i}$	indicator about whether $w_{d,i}$ is generated by a category-topic
η_d	the initial category (*i.e.,* relevant-topic) distribution of document d
θ_d	the general-topic distribution of document d
φ_c	the prior general-topic distribution of all documents of category-topic c
ϕ_b	the word distribution of general-topic b
ϑ_c	the word distribution of category-topic c
α_2	the concentration parameter of φ_c for document's θ_d of category-topic c
$\delta_{w,c}$	the probability of word w being a category word for category-topic c
$\tau_{w,c}$	the relevance weight between word w and category-topic c
κ_d	prior likelihood for document d being a relevant document
ψ	the irrelevant-topic distribution of the whole dataset
ρ	the tuning parameter for category word probability $\delta_{w,c}$
$\alpha_0, \alpha_1, \beta_0, \beta_1$	Dirichlet Priors

. (a) draw a general-topic distribution $\varphi_t \sim Dir(\alpha_0)$;
. (b) draw a word distribution $\vartheta_t \sim$ Dir(β_0);
3 Draw an irrelevant-topic distribution: $\psi \sim Dir(\alpha_1)$
4 For each general-topic $b \in \{1...B\}$:
. (a) draw $\phi_b \sim$ Dir(β_1);

5 For each document $d \in \{1...D\}$:
. (a) draw $y_d \sim Bernoulli(\kappa_d)$;
. (b) if $y_d = 1$:

. $*$ generate an initial relevant-topic distribution η_d;
. $*$ draw a category-topic $c_d \sim$ Multi(η_d);

. (c) if $y_d = 0$:

. $*$ draw a category-topic $c_d \sim$ Multi(ψ);

. (d) draw a general-topic distribution $\theta_d \sim$ Dir($\alpha_2 \cdot \varphi_{c_d}$);
. (e) for each word $i \in \{1...|d|\}$:

. $*$ draw $x_{d,i} \sim$ Bernoulli($\delta_{w_{d,i},c_d}$);
. $*$ if $x_{d,i} = 0$:

. - draw $w_{d,i} \sim$ Multi(ϑ_{c_d});

. $*$ if $x_{d,i} = 1$:

. - draw $z_{d,i} \sim$ Multi(θ_d);
. - draw $w_{d,i} \sim$ Multi($\phi_{z_{d,i}}$);

In the above generative process, the binary variable y_d indicates whether document d is a relevant document ($y_d = 1$) or an irrelevant one ($y_d = 0$). κ_d works as a prior preference for y_d, which indicates the prior probability that a document is a relevant document without considering the textual content. The binary variable $x_{d,i} = 0$ indicates that the associated word $w_{d,i}$ is generated from category-topic c_d , otherwise word $w_{d,i}$ is generated from a specific general-topic $z_{d,i}$ (*i.e.*, $x_{d,i} = 1$).

Here, we infer the hidden parameters $\{\varphi_c, \vartheta_c, \phi_b, \theta_d, z_{d,i}, x_{d,i}, c_d, y_d\}$ in DFC. As with LDA and other PTMs, the exact inference of DFC is intractable. We therefore utilize the Gibbs Sampling to perform the approximate inference and parameter learning [Thomas and Mark 2004]. Specifically, we construct a Markov chain on latent parameters. At each step, a latent parameter or a set of latent parameters are sampled based on the conditional probability given the values of other parameters. In DFC, because parameters $z_{d,i}$ and $x_{d,i}$ are correlated, we jointly sample their values as follows:

$$p(z_{d,i}, x_{d,i} | \mathbf{z}_{\neg(d,i)}, \mathbf{x}_{\neg(d,i)}, \mathbf{c}, \mathbf{w}) \propto \begin{cases} \phi_{b,\neg(d,i)}^{w_{d,i}} \times \theta_{d,\neg i}^{b} \times (1 - \delta_{w_{d,i},c_d}) & z_{d,i} = b, x_{d,i} = 1 \\ \vartheta_{c_d,\neg(d,i)}^{w_{d,i}} \times \delta_{w_{d,i},c_d} & x_{d,i} = 0 \end{cases} \tag{1}$$

where $\phi_{b,\neg(d,i)}^{w_{d,i}}$ is the probability of seeing $w_{d,i}$ under general-topic b excluding the current assignment, $\theta_{d,\neg i}^{b}$ is the probability of seeing general-topic b in document d excluding the current assignment, and $\vartheta_{c_d,\neg(d,i)}^{w_{d,i}}$ is the probability of seeing word $w_{d,i}$ under category-topic c_d excluding the current assignment.

Recall that a category-topic refers to either a relevant-topic or an irrelevant-topic. When $y_d = 0$, we sample an irrelevant-topic t as follows:

$$
\begin{aligned}
&p(c_d = t, y_d = 0 | \mathbf{z}, \mathbf{x}, \mathbf{y}_{\neg d}, \mathbf{c}_{\neg d}, \mathbf{w}) \propto \\
&\prod_{b=1}^{B} \frac{\prod_{w \in d}[(n_b^w + \beta_1 - 1) \cdots (n_{b,\neg d}^w + \beta_1)]}{[\sum_{w=1}^{W}(n_b^w + \beta_1) - 1] \cdots [\sum_{w=1}^{W}(n_{b,\neg d}^w + \beta_1)]} \\
&\times \prod_{b=1}^{B} \frac{(n_d^b + \alpha_2 \cdot \varphi_t^b - 1) \cdots (\alpha_2 \cdot \varphi_t^b)}{[\sum_{k=1}^{B}(n_d^k + \alpha_2 \cdot \varphi_t^k) - 1] \cdots [\sum_{k=1}^{B} \alpha_2 \cdot \varphi_t^k]} \\
&\times \frac{\prod_{w \in d}[(n_t^w + \beta_0 - 1) \cdots (n_{t,\neg d}^w + \beta_0)]}{[\sum_{w=1}^{W}(n_t^w + \beta_0) - 1] \cdots [\sum_{w=1}^{W}(n_{t,\neg d}^w + \beta_0)]} \\
&\times \frac{n_{t,\neg d} + \alpha_1}{\sum_{k=1}^{T}(n_{k,\neg d} + \alpha_1)}(1 - \kappa_d)
\end{aligned} \tag{2}
$$

where n_b^w is the number of times word w is assigned to general-topic b, n_d^b is the number of words that are assigned to general-topic b within document d, n_t^w is the number of times word w is assigned to irrelevant-topic t, n_t is the number of documents that are assigned to irrelevant-topic t. Symbol $\neg d$ means that document d is excluded from the count. When $y_d = 1$, we then sample a relevant-topic r as follows:

$$
\begin{aligned}
&p(c_d = r, y_d = 1 | \mathbf{z}, \mathbf{x}, \mathbf{y}_{\neg d}, \mathbf{c}_{\neg d}, \mathbf{w}) \propto \\
&\prod_{b=1}^{B} \frac{\prod_{w \in d}[(n_b^w + \beta_1 - 1) \cdots (n_{b,\neg d}^w + \beta_1)]}{[\sum_{w=1}^{W}(n_b^w + \beta_1) - 1] \cdots [\sum_{w=1}^{W}(n_{b,\neg d}^w + \beta_1)]} \\
&\times \prod_{b=1}^{B} \frac{(n_d^b + \alpha_2 \cdot \varphi_r^b - 1) \cdots (\alpha_2 \cdot \varphi_r^b)}{[\sum_{k=1}^{T}(n_d^k + \alpha_2 \cdot \varphi_r^k) - 1] \cdots [\sum_{k=1}^{T} \alpha_2 \cdot \varphi_r^k]} \\
&\times \frac{\prod_{w \in d}[(n_r^w + \beta_0 - 1) \cdots (n_{r,\neg d}^w + \beta_0)]}{[\sum_{w=1}^{W}(n_r^w + \beta_0) - 1] \cdots [\sum_{w=1}^{W}(n_{r,\neg d}^w + \beta_0)]} \eta_d(r) \kappa_d
\end{aligned} \tag{3}
$$

where n_r^w is the number of times word w is assigned to relevant-topic r. The category general-topic distributions $\{\varphi_r, \varphi_t\}$, document general-topic distribution θ_d, word-distributions $\{\vartheta_r, \vartheta_t, \phi_b\}$ can be computed based on the point estimation as follows:

$$\varphi_r^b = \frac{n_r^b + \alpha_0}{\sum_{k=1}^{B}(n_r^k + \alpha_0)} \tag{4}$$

$$\varphi_t^b = \frac{n_t^b + \alpha_0}{\sum_{k=1}^{B}(n_t^k + \alpha_0)} \tag{5}$$

$$\theta_d^b = \frac{n_d^b + \alpha_2 \cdot \varphi_{c_d}^b}{\sum_{k=1}^{B}(n_d^k + \alpha_2 \cdot \varphi_{c_d}^k)} \tag{6}$$

$$\vartheta_r^w = \frac{n_r^w + \beta_0}{\sum_{w'=1}^{W}(n_r^{w'} + \beta_0)} \tag{7}$$

$$\vartheta_t^w = \frac{n_t^w + \beta_0}{\sum_{w'=1}^{W}(n_t^{w'} + \beta_0)} \tag{8}$$

$$\phi_b^w = \frac{n_b^w + \beta_1}{\sum_{w'=1}^{W}(n_b^{w'} + \beta_1)} \tag{9}$$

Because all the pairs of $z_{d,i}$ and $x_{d,i}$ of document d are affected by the choice of category-topic c_d and vice versa (Equations 1-3), the sampling order of $z_{d,i}, x_{d,i}$ and c_d becomes a critical factor. Simply sampling a new relevant-topic r or irrelevant-topic t based on Equations 2-3 with all $z_{d,i}, x_{d,i}$ values conditioned on the previous c_d could not converge to the true posterior distribution. This is a common issue of Markov Chain Monte Carlo (MCMC) methods, called autocorrelation phenomenon [Straatsma et al. 1986]. The details of the Gibbs sampling process of DFC are described in Algorithm 1. Here, to avoid the autocorrelation, at the first step we sample each pair of $z_{d,i}, x_{d,i}$ conditioned on each possible c (*i.e.,* $R + T$ possible choices, Lines 3-19 in Algorithm 1). Then, the likely c_d is sampled conditioned on all the corresponding $z_{d,i}, x_{d,i}$ values with Equations 2 and 3. Afterwards, all the values of $z_{d,i}, x_{d,i}$ are set to the updated c_d's corresponding values sampled in the first step (Lines 21-24 in Algorithm 1). For document filtering and classification, we take the sampled c_d of document d at the last iteration as the prediction. Document d belongs to the corresponding category of relevant-topic r when $y_d = 1$ and c_d refers to relevant-topic r. Otherwise, document d is considered as an irrelevant document (*i.e.,* filtering) when c_d refers to an irrelevant-topic.

3.2. Pseudo Seed Word Extraction

In DFC, users only need to specify the seed words associated with the categories of interest. However, without any supervision, it is hard to successfully infer the irrelevant-topics, which in turn hurts the classification performance. Because the LDA model infers the hidden topics from documents based on the higher-order word co-occurrence patterns, the words relevant to a specific category are likely to be grouped together under a LDA hidden topic. Hence, we choose to resort to using standard LDA to extract the hidden topics from the collection in an unsupervised manner. We expect that some hidden topics discovered by LDA are relevant to the irrelevant-topics (*i.e.,* irrelevant categories) to some extent. Specifically, we can calculate the distance between a LDA hidden topic and all the seed words provided for the categories of interest. The least relevant LDA topics are then likely to represent the irrelevant-topics underlying the collection. Let $\mathbb{S} = \cup_r \mathbb{S}_r$ denotes the set of all seed words provided, we calculate the distance between a LDA hidden topic k and $\mathbb{S}$ as follows:

$$dist(k, \mathbb{S}) = 1 - \sum_{s \in \mathbb{S}} \sum_{w \in \mathbb{W}_k} p_{LDA}(w|k) p_{LDA}(k|s) \tag{10}$$

where $\mathbb{W}_k$ is the dominating topical words under LDA hidden topic k, $p_{LDA}(w|k)$ is the word probability under topic k, $p_{LDA}(k|s)$ is the topic probability under seed word s. Here, we take $\mathbb{W}_k$ as the top-10 topical words associated with topic k in Equation 10, since only top topical words can precisely represent the meaning of the topic. Before the distance calculation, we first filter out the noisy LDA hidden topics. In detail, the LDA hidden topics matching the following criterions are filtered out based on their top-10 topical words: 1) the percentage of numeric words is larger than 50%; 2) the percentage of words with the length less than 4 characters is larger than 50%. Then, we take the top-10 topical words under each of T least relevant LDA hidden topics as the pseudo seed words for the irrelevant-topics in DFC.

Given a collection of unlabeled documents and a few seed words for each category, a critical challenge of DFC is to incorporate the supervision of the provided seed words to filter and classify the documents through topic influence. As described in the generative process, η_d works as the prior preference on the specified categories for d. And $\delta_{w,c}$ works as the prior preference that word w is generated from category-topic c (*i.e.,* $x_{d,i} = 0$). We call the words that are generated by a category-topic *category words*. In DFC, we need to derive these prior knowledge solely based on seed words to enable effective document filtering and classification. In the following, we will describe the mechanisms used in DFC to estimate *category word probability* and the *initial document category distribution*.

Algorithm 1: DFC-Sampling Process

input : D documents, R relevant-topics, T irrelevant-topics, B general-topics, topic assignment list $\mathbb{L} = \{\mathbb{L}_1, ..., \mathbb{L}_D\}$ where $\mathbb{L}_d = \{(z_{d,1}, x_{d,1}), ..., (z_{d,|d|}, x_{d,|d|})\}$ is the topic assignment list of all the words in document d.

output: Category-topic c_d for all documents and the updated topic assignment list $\mathbb{L}$

foreach $d \in D$ **do**
  foreach $r \in R$ **do**
    `/* assume that` d `belongs to relevant-topic` r `*/`
    create an empty list $\mathbb{L}_d^r$ for relevant-topic r;
    foreach $i \in d$ **do**
      foreach $b \in B$ **do**
        calculate general-topic likelihood score $p(z_{d,i} = b, x_{d,i} = 1 | c_d = r)$; `/* see Eq.1 */`
      calculate category-topic likelihood score $p(z_{d,i} = r, x_{d,i} = 0 | c_d = r)$; `/* see Eq.1 */`
      sample $z_{d,i}$ and $x_{d,i}$ based on the topic likelihood scores;
      $\mathbb{L}_d^r.append((z_{d,i}, x_{d,i}))$;
    calculate likelihood score $p(c_d = r, y_d = 1 | d)$ of d conditioned on $\mathbb{L}_d^r$; `/* see Eq.3 */`
  foreach $t \in T$ **do**
    `/* assume that` d `belongs to irrelevant-topic` t `*/`
    create an empty list $\mathbb{L}_d^t$ for irrelevant-topic t;
    foreach $i \in d$ **do**
      foreach $b \in B$ **do**
        calculate general-topic likelihood score $p(z_{d,i} = b, x_{d,i} = 1 | c_d = t)$; `/* see Eq.1 */`
      calculate category-topic likelihood score $p(z_{d,i} = t, x_{d,i} = 0 | c_d = t)$; `/* See Eq.1 */`
      sample $z_{d,i}$ and $x_{d,i}$ based on the topic likelihood scores;
      $\mathbb{L}_d^t.append((z_{d,i}, x_{d,i}))$
    calculate likelihood score $p(c_d = t, y_d = 0 | d)$ of d conditioned on $\mathbb{L}_d^t$; `/* see Eq.2 */`
  sample c_d and y_d based on the likelihood scores of d ;
  if $c_d == r$ *and* $y_d == 1$ **then**
    $\mathbb{L}_d = \mathbb{L}_d^r$;
  else if $c_d == t$ *and* $y_d == 0$ **then**
    $\mathbb{L}_d = \mathbb{L}_d^t$;

3.3. Estimating Category Word Probability

The discriminative power carried by the category words essentially determines the classification accuracy of DFC. In DFC, the chance of choosing an occurrence of word w as a category word for category-topic c relies on the prior knowledge constructed by using the seed words, *i.e.,* a category word probability $\delta_{w,c}$ that word w is a category word ($x = 0$) under category-topic c.

Initially, it is simple to assume that each word has the same probability to be a category word for each category-topic. Hence, we can refer a global parameter ρ as the probability that a word is picked as a category word for any category-topic and $1 - \rho$ as the probability that a word is picked as being from a general-topic, *i.e.,* $\delta_{w,c} = \rho$. Note that the category-topics are mainly governed by the higher-

order word co-occurrence patterns. Therefore, the words that frequently co-occur with each other under the documents of the same category can be grouped together in the corresponding category-topic. Because DFC is a probabilistic topic model, a wrong category-topic may be sampled for some documents. Given the underlying collection is severely imbalanced, the documents of the largest category could be allocated with a wrong category-topic. The smaller categories will be dominated by these documents. The resultant incorrect category-topics of the smaller categories in turn will hurt the classification performance very much.

Doc-Rel. Similar to the given seed words of c, category words are expected to represent the semantic meaning of category-topic c. However, without training documents labeled for a category, the statistically informative words could not be easily derived. On the other hand, as to relevant-topics, we believe that category words could be semantically or statistically related to the seed words of that category. Although semantically relevant words can be extracted based on the seed words and an external thesauri or knowledge base, such prior knowledge bases may not always be available. Here, we simply use word co-occurrences to estimate the probability of category words. If a word has high word co-occurrences with the seed words of a category, this word is more likely to be a category word. The degree of co-occurrences between a word w and a seed word s is measured by the conditional probability $p(w|s)$:

$$p(w|s) = \frac{df(w,s)}{df(s)} \tag{11}$$

where $df(s)$ is the number of the documents containing seed word s, and $df(w,s)$ is the number of the documents containing both word w and seed word s. Then, we calculate the relevance score $rel(w,c)$ and weight $\tau_{w,c}$ for each word w and category-topic c as follows:

$$rel(w,c) = \frac{1}{|\mathbb{S}_c|}\sum_{s\in\mathbb{S}_c} p(w|s) \tag{12}$$

$$\nu(w,c) = \max(\frac{rel(w,c)}{\sum_c rel(w,c)} - \frac{1}{A}, 0) \tag{13}$$

$$\nu_c(w,c) = \frac{\nu(w,c)}{\sum_w \nu(w,c)} \tag{14}$$

$$\tau_{w,c} = \max(\frac{\nu_c(w,c)}{\sum_c \nu_c(w,c)}, \epsilon) \tag{15}$$

In Equation 12, $\mathbb{S}_c$ is the collection of seed words of category-topic c. Note that Equations 11-12 needs seed words as input. However, for the irrelevant-topics, the seed words can not be provided since the possible (irrelevant) categories under the collection is unknown beforehand. In Section 3.2, we propose a simple but novel mechanism to extract the pseudo seed words for irrelevant-topics in an unsupervised manner. Then, for irrelevant-topics, $\delta_{w,c}$ values are estimated based on these pseudo seed words for topic inference.

In Equation 13, we normalize the relevance score $rel(w,c)$ and subtract it by the average relevance score $1\backslash A$ for each category-topic where A refers to the total number of category-topics under consideration (*i.e.,* $A = R + T$). It is expected that word w is a category word for category-topic c only if w and c have a high $rel(w,c)$ value. Therefore subtracting the relevance scores by the average is necessary to filter out irrelevant categories using Equation 13. At this stage, we can simply take $\nu(w,c)$ as the final weight indicating the relevance between w and c. However, we observe that the absolute value of $\nu(w,c)$ does not really reflect the true relevance between word w and category-topic c, because the values are largely affected by the statistical properties of the seed words as well.

Given a seed word s with very high document frequency (*i.e.,* $df(s)$ is large), $p(w|s)$ would be relatively smaller than other seed words with small document frequencies based on Equation 11. This results in relatively smaller $\nu(w,c)$ values for all words under category c. Hence, we take the

impact of the seed words into account by normalizing $\nu(w, c)$ with respect to c by using Equation 14. Similarly, the $\nu_c(w, c)$ values for different word w could be in different scales. We further normalize $\nu_c(w, c)$ with respect to word w and take it as the final relevance weight $\tau_{w,c}$ between word w and category c, by using Equation 15. A higher $\tau_{w,c}$ means that word w is more likely to be a category word of category c. A small constant ϵ is assigned for the words that have relatively low $rel(w, c)$ values to avoid being zero, and $\epsilon = 0.01$ in this work. With $\tau_{w,c}$ calculated by Equation 15, we can refine the category word probability $\delta_{w,c}$ of word w and category-topic c as follows:

$$\delta_{w,c} = \frac{\tau_{w,c}\rho}{1 - \rho + \tau_{w,c}\rho} \tag{16}$$

In Equation 16, ρ becomes a tuning parameter within $[0, 1]$, specifying the importance of $\tau_{w,c}$ for $\delta_{w,c}$. When $\rho = 0$ (*i.e.,* $\delta_{w,c} = 0$), DFC is downgraded to the standard LDA and classify each document based on the general-topic distribution of the document only. When $\rho = 1$ (*i.e.,* $\delta_{w,c} = 1$), DFC consists of only category-topics, and all word occurrences are assigned to some category-topic, which is similar to TLC++ model proposed in [Hingmire and Chakraborti 2014] with the exception that the association between the topic and category is made by the seed words here instead of manual labeling as in the TLC++.

The above procedure simply estimates the probability of category words based on the document-level word co-occurrences. Therefore, we denote this relevance estimation mechanism as Doc-Rel.

Topic-Rel. Normally, a document could cover multiple topics and contain noisy information (*e.g.,* common words). Two words appearing together may not indicate their semantic relevance because these two words could refer to two distinct semantic topics. Recall that we utilize the standard LDA to extract the pseudo seed words for irrelevant-categories. Here, we also devise a topic-based mechanism to estimate category word probabilities. Let $p_{LDA}(w|k)$ be the word probability obtained for hidden topic k by using LDA, we can therefore derive the topic probability $p_{LDA}(k|w)$ by using Bayes' theorem [Li et al. 2016a]. We denote this topic-level relevance estimation mechanism as Topic-Rel. Given the most relevant L topics $\mathbb{L}_s$ for a seed word s in terms of $p_{LDA}(k|s)$, we can calculate the relevance score $rel(w, c)$ as follows:

$$rel(w, c) = \frac{1}{|\mathbb{S}_c|} \sum_{s \in \mathbb{S}_c} \sum_{k \in \mathbb{L}_s} p_{LDA}(w|k) p_{LDA}(k|s) \tag{17}$$

We then calculate $\delta_{w,c}$ by following Equations 13-16. Instead of measuring the relevance based on coarse document-level co-occurrence information, we take the topical information into consideration in Equation 17. Only the relevant words under the most relevant topics of the seed words are likely to be the category words for the corresponding category. In our study, we observe that a seed word only contains very few dominating LDA topics. That is, $p_{LDA}(k|s)$ is very small for most LDA topics. Here, we use the most relevant 3 topics for the calculation (*i.e.,* $L = 3$) to save the computation cost[3].

Note that both Topic-Rel and pseudo seed word extraction require running LDA over the document collection. The inference results from a single run of LDA can be used for both relevance estimation and pseudo seed word extraction. However, efficiency would still be a potential issue by utilizing LDA to extract the hidden topics. Fortunately, we find that a small iteration number (*i.e.,* 100 iterations) is adequate enough to deliver promising performance and no further performance gain could be obtained with further iterative inference. In this work, we take the results after running LDA over 100 iterations. Moreover, the existing works enabling efficient topic inference would be utilized [Yao et al. 2009; Yuan et al. 2015]. However, this is beyond the focus of this work.

[3]We validated the performance of DFC with different L values. However, litter performance variation is experienced with larger L values.

3.4. Initial Document Category-Topic Distribution Estimation

Since the seed words represent the meaning of a category, it is reasonable to assume that a document containing the seed words of a category is likely to belong to this category. Specifically, given a document d, we derive the initial category distribution η_d as follows:

$$f(d,c) = \sum_{s \in \mathbb{S}_c} tf(s,d) \tag{18}$$

$$\eta_d(c) = \frac{\ln(1+f(d,c))+\gamma}{\sum_c (ln(1+f(d,c)))+R\gamma} \tag{19}$$

where $tf(s,d)$ is the term frequency of seed word s in document d, and γ is a prior parameter for the Dirichlet smoothing, $\gamma = 0.01$ in this work. DFC degrades to become our earlier proposed dataless classification model STM when $\delta_{w,c}$ is estimated by using Doc-Rel mechanism and all the irrelevant-topics are not considered (*i.e.,* either $T = 0$ or $\kappa_d = 1$). That is, DFC can also be used to conduct conventional document classification in a dataless manner (*i.e.,* classification without filtering).

4. EXPERIMENT

In this section, we evaluate the filtering and classification performance of the proposed DFC[4] against other state-of-the-art dataless text classification alterantives and supervised learning methods. Under some parameter settings, DFC can be degraded to be equivalent of the earlier proposed STM for classification without filtering. That is, DFC can also be adapted for dataless classification without filtering. For the completeness of the comparison, we also conduct the performance comparison in terms of conventional classification without filtering. Then, we analyze the impact of parameter settings in DFC. Our experimental results show that DFC outperforms all existing state-of-the-art competitors and is very robust to the parameter settings. At last, we conduct a thorough study about the impact of seed words by using the existing state-of-the-art dataless techniques (including DFC).

4.1. Datasets

Two real-world text collections for the classification are used here for performance evaluation.

20-Newsgroup (20NG): The 20NG is a widely used dataset[5] for document classification research [Chang et al. 2008; Chen et al. 2015; Guan et al. 2009; Kusner et al. 2015; Xie and Xing 2013]. It contains approximately $20,000$ newsgroup documents, evenly distributed across 20 different newsgroups/categories. We use the *bydate* version of the 20NG dataset, where a total of $18,846$ documents are divided into a training set (60%) and a test set (40%). These 20 categories can be further aggregated into 6 major sub-categories. For example, the major category sci consists of 4 categories: sci.crypt, sci.electronics, sci.med, sci.space. This dataset has been previously used in the related works [Chang et al. 2008; Chen et al. 2015; Druck et al. 2008; Hingmire and Chakraborti 2014; Hingmire et al. 2013; Li et al. 2016b; Song and Roth 2014]. When parsing the documents, we keep the text contained in the "Subject", "Keywords", and "Content" fields. The information in the other fields and email addresses are filtered out.

Reuters-10: Reuters-21578 is also a widely used dataset for document classification. It contains $21,578$ documents in 135 categories. Among them, $13,625$ and $6,188$ documents are in the training set and test set respectively. This dataset is very imbalanced and the variation of category size is quite large. We used the 10 largest categories (hence denoted by Reuters-10) in the dataset with Aptè split[6]. We further discard the documents belonging to more than one category. This left us with a total of $7,285$ documents: $5,228$ documents in train and $2,057$ documents in test set. The same

[4]Our implementation will be released after paper acceptance.
[5]http://qwone.com/~jason/20Newsgroups/
[6]http://kdd.ics.uci.edu/database/reuters21578/reuters21578.html

Table II. Statistics of 20NG dataset. #train/#test: the number of training/test documents; Avg($|d|$): the average length of the documents; $|\mathbb{S}^L|$ / $|\mathbb{S}^D|$: the number of seed words obtained from the category label $\mathbb{S}^L$, and from category description $\mathbb{S}^D$.

Category label	#train	#test	Avg($\|d\|$)	$\|\mathbb{S}^L\|$ / $\|\mathbb{S}^D\|$
alt.atheism	480	319	157.30	1/5
comp.graphics	584	389	135.58	2/5
comp.os.ms-windows.misc	591	394	226.08	4/8
comp.sys.ibm.pc.hardware	590	392	95.10	5/7
comp.sys.mac.hardware	578	385	86.98	4/3
comp.windows.x	593	395	146.81	3/5
misc.forsale	585	390	77.27	1/8
rec.autos	594	396	103.10	1/7
rec.motorcycles	598	398	95.41	1/3
rec.sport.baseball	597	397	113.07	2/3
rec.sport.hockey	600	399	144.714	2/3
sci.crypt	595	396	164.34	2/5
sci.electronics	591	393	97.60	2/5
sci.med	594	396	141.16	2/5
sci.space	593	394	148.34	2/6
soc.religion.christian	599	398	175.13	3/6
talk.politics.guns	546	364	166.24	2/5
talk.politics.mideast	564	376	243.53	2/5
talk.politics.misc	465	310	210.61	1/3
talk.religion.misc	377	251	165.61	1/6
politics	1575	1050	207.02	3/13
religion	1456	968	166.79	4/12
sci	2373	1579	137.92	5/31
comp	2936	1955	138.38	13/27
rec	2389	1590	114.10	4/17

Table III. Statistics of the Reuters-10 dataset.

Category label	#train	#test	Avg($\|d\|$)	$\|\mathbb{S}^L\|$ / $\|\mathbb{S}^D\|$
acq	1,435	620	68.13	1/9
coffee	89	21	109.54	1/4
crude	223	98	118.10	1/9
earn	2,637	1,040	57.97	1/7
gold	70	20	78.36	1/4
interest	140	57	89.23	1/8
money-fx	176	69	99.70	2/9
ship	107	35	77.94	1/8
sugar	90	24	104.08	1/2
trade	225	73	130.02	1/7

subset, Reuters-10, has been previously used in the related works as well [Chen et al. 2015; Li et al. 2016b; Xie and Xing 2013].

For both datasets, we further remove the stop words, the words shorter than 2 characters, and the words appearing in fewer than 5 documents. The data statistics after preprocessing are reported in Tables II and III, respectively. The statistics of the 5 major categories of 20NG dataset used in the experiments are reported in the last 5 rows in Table II. Observe from Table III that the Reuters-10 is very imbalanced. While most categories have around $100-300$ documents, the two largest categories (*i.e.,* earn, acq) have $3,677$ and $2,055$ documents respectively.

4.2. Experimental Setup

Methods in Comparison. The proposed DFC is compared against the following state-of-the-art *dataless text classification* methods and *supervised classification* methods:

Topic Labeled Classification with Labeled Words (TLC++). This method learns to classify documents based on the posterior topic proportions and the category labels of the topics [Hingmire and Chakraborti 2014]. The labels of the topics are annotated manually based on the highly probable words in each topic. We present the 30 most probable words in each topic for topic labeling, the same setting recommended by its authors.

Generalized Expectation with Feature Labels (GE-FL). It learns a maximum entropy based text classifier by using the labeled words in each category as soft constraints [Druck et al. 2008]. Here, we use the same seed words that are used for DFC as labeled words of each category for fair comparison. We used the implementation of GE-FL that is provided in the MALLET toolkit.[7]

Descriptive LDA (DescLDA). This method learns the category label of a document by applying document clustering over the learned hidden topics [Chen et al. 2015]. The hidden topic distributions are inferred based on the seed words. DescLDA has a tunable parameter: the number of topics. We report the best results with the optimal setting obtained in our experiments.

Seed-based NB-EM (SNB-EM). It learns dataless text classifier in a semi-supervised manner [Liu et al. 2004], where NB-EM method [Nigam et al. 2000] is used for model building. We report the best performance with the optimal parameter settings obtained in our experiments. For fair comparison, we use the same seed words that are used for DFC to build the initial training instances.

Support Vector Machines (SVM). This is a state-of-the-art supervised learning technique for text classification. We train a linear SVM classifier by using LIBSVM with the default parameter settings and TF-IDF weighting scheme.[8]

Seed-based Support Vector Machines (SSVM). We construct a pseudo training set by labeing a training document with a category if the document contains any seed word of that category. Then, we train a linear SVM classifier by using LIBSVM with the default parameter settings and TF-IDF weighting scheme.

sLDA. It is a supervised text classifier based on the LDA model [Blei and McAuliffe 2007]. We train the model by using the implementation provided by the authors[9]. The best results obtained in our experiments are reported.

MedLDA. It is an LDA-based supervised topic model which exploits max-margin principle for jointly max-margin and maximum likelihood learning [Zhu et al. 2009, 2012]. We use the implementation provided by the authors. The best restuls obtained in our experiments are reported.

Among the above methods in comparison, SVM, SSVM, TLC++, MedLDA and sLDA can be adapted for classification with filtering. For SVM and SSVM, we can adopt the one-class SVM for evaluation[10]. For TLC++, a pseudo category is added to indicate the irrelevance during the topic annotation process. As for MedLDA and sLDA, we introduce a pseudo category to cover all irrelevant documents in the training set. The hyper-parameters for these methods are tuned accordingly for optimal performance.

The rest methods not discussed above are used for performance evaluation in classification without filtering. Note that, Chang *et al.* learns the category label of a document by projecting the document and the category into the same semantic space of Wikipedia concepts [Chang et al. 2008]. The nearest-neighbor based explicit semantic analysis (NN-ESA) is then used for dataless classification without filtering. Since NN-ESA involves parsing the whole Wikipedia, we choose not to include this method for comparison. Nevertheless, it was reported that DescLDA significantly outperforms NN-ESA in earlier study [Chen et al. 2015]. A state-of-the-art dataless classifier (*i.e.,* STM) was proposed in our earlier work [Li et al. 2016b]. However, STM can be applied for classification without filtering only. Note that the proposed DFC subsumes STM under some particular parameter settings. As an extension of STM, for classification without filtering evaluation, we choose to report the classification performance of DFC with the corresponding settings (ref. Section 3.4).

[7] http://mallet.cs.umass.edu

[8] www.csie.ntu.edu.tw/∼cjlin/liblinear

[9] http://www.cs.cmu.edu/∼chongw/slda/

[10] The implementation provided in LIBSVM is used.

Parameter Setting. In DFC, there are several hyper-parameters. They are set to typical values: $\alpha_0 = 50/B$, $\beta_0 = \beta_1 = 0.01$, as used in PTM studies [Thomas and Mark 2004]. We set $\gamma = \epsilon = 0.01$ in our experiments. As to the tunable parameters in DFC, we use the following setting: (1) For classification without filtering (*i.e.,* $T = 0, \kappa_d = 1$), we follow the setting in STM [Li et al. 2016b] to set $\alpha_2 = 100, \rho = 0.85, B = 3 \cdot R$ in the evaluation; (2) For classification with filtering, we set $\alpha_1 = 50/T, \alpha_2 = 100, \rho = 0.95, T = 20, B = 3 \cdot (R + T)$. Also, we use $\kappa_d = 0.5$ such that no prior preference is given to a document for its relevance. The study of the tunable parameters will be presented in Section 4.4. We run DFC for 20 iterations, and then the category-topic assigned to a document during the last iteration is taken as its predicted label.

Performance Metric. In the experiments, we use the standard training/test partitions of the two datasets for the evaluation. For all the dataless classifiers, the classifiers run over both the training and test documents as a single collection of unlabeled documents (*i.e.,not using their labels*). The classification accuracy are evaluated based on the labels of documents in test set, for fair comparison with the supervised methods. For supervised methods, the classifiers are developed using the training documents and then evaluated on the test set, as per normal.

For performance comparison, we report macro-averaged F_1 scores (Macro-F_1) [Manning et al. 2008]. Macro-F_1 is the averaged F_1 scores of all categories. We report the average results over 10 runs for all the methods (excluding SNB-EM, SSVM and SVM) with random model initialization. The same setting are used in previous works [Hingmire and Chakraborti 2014; Li et al. 2016b]. The statistical significance is conducted by using the student *t-test*.

Seed Words Selection. The quality of the seed words is a critical factor for all dataless classifiers. Here, we exploit two sets of the seed words selected from the category label (denoted by $\mathbb{S}^L$) and description (denoted by $\mathbb{S}^D$) respectively. Category label means that the seed words are extracted from the label in the given dataset directly. For example, from the category label comp.sys.ibm.pc.hardware in 20NG, five seed words "computer, systems, ibm, pc, hardware" are extracted as $\mathbb{S}^L$. Note that the semantically irrelevant words in the label are excluded here. For example, "talk" is excluded from category talk.politics.guns. The seed words in $\mathbb{S}^D$ are compiled manually with the domain knowledge. For example, the authors of DescLDA followed the labeling procedure used in TLC++ [Hingmire and Chakraborti 2014] (*i.e.,* assisted by standard LDA) to compile the $\mathbb{S}^D$. The two sets of seed words used here are both used in earlier studies [Chang et al. 2008; Chen et al. 2015; Song and Roth 2014]. Further details[11] about these seed words can be found in the work of DescLDA [Chen et al. 2015]. These seed words are included in Appendix. The number of seed words in $\mathbb{S}^L$ and $\mathbb{S}^D$ in each category are listed in Tables II and III.

4.3. Performance Comparison

Here, we first evaluate the classification preformance of the proposed DFC against the existing state-of-the-art dataless and supervised classifiers in terms of classification with filtering. Then, we further report the performance comparison for classification without filtering.

Classification with Filtering. For classificaiton with filtering, we need to filter out the documents irrelevant to any specified category. In this sense, we take the documents from the specified categories as relevant ones, and the rest documents in the collection as irrelevant documents. In this set of experiments, we create overall 15 classification with filtering tasks over the two datasets, ranging from 1 to 4 specified categories. Note that Reuters-10 is a very imbalanced dataset (ref. Table III). The smallest category gold contains only 70 training documents and 20 testing documents, while the largest category earn contains 2,637 training documents and 1,040 testing documents. Hence, the tasks gold and earn are created to choose the documents in these extreme cases. Similarly, the tasks acq-earn and coffee-gold-sugar are created by using the two largest categories and the three

[11] The seed words for these two datasets are available at https://github.com/WHUIR/STM

smallest categories respectively. Tables IV and V report the Macro-F_1 scores of these methods on these 15 tasks by using $\mathbb{S}^L$ and $\mathbb{S}^D$ respectively. We make the following observations:

First, with $\mathbb{S}^D$ as the seed words, DFC +Doc-Rel achieves the best performance on 12 out of 15 classification with filtering tasks among the dataless classifiers. Also, DFC +Topic-Rel achieves the best performance on a single task and the second best on 14 tasks. Both TLC++ and SSVM achieves much worse but comparable performance to each other. Overall on these 15 classification with filtering tasks, DFC +$\mathbb{S}^D$ +Doc-Rel outperforms TLC++ and SSVM by around 170.5% and 136.4% on average, respectively. The relative performance gain for DFC +$\mathbb{S}^D$ +Topic-Rel is 146.6% and 115.5% over the two baseline methods respectively.

Second, the supervised learning methods SVM, MedLDA and sLDA may not always perform better than the dataless counterparts. Specifically, SVM performs consistently worse than DFC +$\mathbb{S}^D$/$\mathbb{S}^L$ +Doc-Rel and DFC +$\mathbb{S}^D$ +Topic-Rel. Also, SVM only manages to deliver better performance than DFC +$\mathbb{S}^L$ +Topic-Rel on 2 classification with filtering tasks: sci and acq-earn. The averaged performance gain over SVM by DFC +Topic-Rel is 54.3% and 85.6% respectively, under $\mathbb{S}^L$ and $\mathbb{S}^D$ settings. It goes up to 81.6% and 103.5% respectively by using DFC +Doc-Rel. Similarly, with $\mathbb{S}^L$, both DFC +Doc-Rel and DFC +Topic-Rel outperforms MedLDA on most classification with filtering tasks. Also, with $\mathbb{S}^D$, DFC +Doc-Rel performs consistently better than MedLDA on all 15 classification with filtering tasks. And DFC +Topic-Rel also manage to achiever better performance than MedLDA on 14 out of 15 tasks.

As to sLDA, under $\mathbb{S}^D$ setting, DFC +Doc-Rel outperforms sLDA on 10 out of 15 classification tasks, and DFC +Topic-Rel is better on 4 classification tasks. With $\mathbb{S}^L$, both DFC +Doc-Rel and DFC +Topic-Rel outperforms sLDA on 4 classification tasks respectively. Specifically, DFC achieves much better performance than sLDA on two tasks related to the smaller categories: gold and coffee-gold-sugar. As reported in Table III, these categories contain relatively fewer training documents. Without adequate amount of training data, supervised classification techniques could be error-prone to handle the classification with filtering in an extreme sparsity. Moreover, DFC +$\mathbb{S}^L$ +Doc-Rel obtains very close performance to sLDA on two classification tasks: politics-religion and politics-rec-religion-sci.

Third, among the three supervised classifiers, sLDA performs significantly much better than SVM and MedLDA. We also observe that sLDA performs much better than MedLDA on all 15 classification with filtering tasks, though they are both built on the basis of probabilistic topic modeling techniques. Moreover, MedLDA performs significantly better than SVM on most tasks. Specifically, MedLDA achieves superior performance than SVM on 11 out of 15 classification with filtering tasks. We believe that there are two main reasons to explain their performance difference. First, relying on word occurrences alone (*e.g.,* SVM) is too limited to address the document filtering, because the training documents for all the categories underlying the document collection is unavailable. Second, both SVM and MedLDA are built on the basis of max-margin principle. When document collection becomes severely imbalanced, the max-margin based optimization would easily lead to model overfitting. Note that the colleciton becomes very imbalanced when the number of specified categories are few or the number of documents covered by the specified categories are small. For example, MedLDA performs relatively worse on tasks like med, gold and coffee-gold-sugar. In comparison, the averaged performance gain by DFC +$\mathbb{S}^D$ +Doc-Rel over sLDA is 3.7%.

This set of comparisons is unfair since MedLDA, sLDA and SVM are supervised methods which access to a lot of training documents. Besides, the training for sLDA with 40 hidden topics takes about 6 hours. And the optimal performance is often obtained by using a larger topic number (*e.g.,* 60 and 80), which needs much more time. This inefficiency extremely hinders its application in the scenario where the information needs are dynamically changing and fast response is required. In contrast, DFC +Doc-Rel takes only 30 and 15 minutes on average for 20NG and Reuters-10 respectively, excluding the LDA hidden topic inference. Moreover, the proposed DFC can be easily deployed in parallel computing settings. The overall superiority suggests that DFC +Doc-Rel is a desirable solution for real-world classification with filtering scenarios with dynamic information needs.

Table IV. Macro-F_1 of the six methods for classification with filtering, where the seed words in $\mathbb{S}^L$ are used. The best and second best results by dataless classifiers are highlighted in boldface and underlined respectively, on each task. † indicates that the difference to the best dataless classifier is statistically significant at 0.05 level. ▲ and ▼ indicate that the supervised classifiers perform better or worse than the best dataless classifier respectively. Avg: the averaged Macro-F_1 over all tasks.

Dataset	Classification task	DFC		TLC++	SSVM	MedLDA	sLDA	SVM
		Doc-Rel	Topic-Rel					
20NG	med	0.580†	**0.692**	0.004†	0.100	0.256†▼	0.701▲	0.190▼
	space	0.329†	**0.756**	0.027†	0.166	0.304†▼	0.763▲	0.255▼
	sci	**0.716**	0.291†	0.007†	0.298	0.426†▼	0.755†▲	0.370▼
	religion	**0.870**	0.432†	0.500†	0.391	0.390†▼	0.857▼	0.409▼
	med-space	0.595†	**0.791**	0.047†	0.133	0.332†▼	0.737†▼	0.222▼
	pc-mac	0.179†	**0.242**	0.019†	0.131	0.350†▲	0.362†▲	0.208▼
	politics-religion	**0.849**	0.506†	0.311†	0.359	0.564†▼	0.861▲	0.423▼
	politics-sci	0.602†	**0.684**	0.285†	0.313	0.581†▼	0.797†▲	0.404▼
	comp-religion-sci	**0.779**	0.585†	0.448†	0.378	0.637†▼	0.850†▲	0.418▼
	politics-rec-religion-sci	**0.839**	0.658†	0.578†	0.341	0.705†▼	0.848▲	0.407▼
	autos-motorcycles-baseball-hockey	0.839	**0.844**	0.262†	0.148	0.487†▼	0.782†▼	0.262▼
Reuters-10	gold	**0.848**	0.682†	0.000†	0.294	0.000†▼	0.620†▼	0.333▼
	earn	0.947	0.887†	0.943	0.419	0.489†▼	0.984†▲	0.822▼
	acq-earn	0.892†	0.274†	**0.940**	0.443	0.653†▼	0.980†▲	0.636▼
	coffee-gold-sugar	**0.915**	0.841†	0.093†	0.563	0.121†▼	0.764†▼	0.574▼
Avg		0.719	0.611	0.298	0.298	0.420	0.777	0.396

Fouth, with $\mathbb{S}^D$, DFC +Doc-Rel, DFC +Topic-Rel and SSVM perform better than their counterparts with $\mathbb{S}^L$ on the majority of the tasks (the study about seed words is presented in Section 4.5). This is expected because more semantic information could be exploited by providing more seed words. The performance gain by using $\mathbb{S}^D$ over $\mathbb{S}^L$ is about 12.1%, 20.3% and 14.3% for DFC +Doc-Rel, DFC +Topic-Rel, SSVM on average, respectively. The performance gap between DFC +$\mathbb{S}^D$ +Doc-Rel and DFC +$\mathbb{S}^L$ +Doc-Rel is the smallest, compared to the other alternatives. When $\mathbb{S}^D$ is used, SSVM performs much better than TLC++. This is reasonable since more seed words could produce more pseudo training documents of high quality. DFC +$\mathbb{S}^L$ +Doc-Rel outperforms TLC++ on 14 tasks, and outperforms SSVM+$\mathbb{S}^L$ and SSVM+$\mathbb{S}^D$ on all the tasks. Similarly, DFC +$\mathbb{S}^L$ +Topic-Rel outperforms TLC++, SSVM+$\mathbb{S}^L$ and SSVM+$\mathbb{S}^D$ on 12, 13 and 13 tasks respectively. As to the two relevance estimation mechanisms, DFC +Doc-Rel outperforms DFC +Topic-Rel on the majority of the classification tasks when either $\mathbb{S}^L$ or $\mathbb{S}^D$ is used. With $\mathbb{S}^L$, DFC +Doc-Rel outperforms DFC +Topic-Rel on 9 out of 15 classification tasks. The situation becomes much worse in $\mathbb{S}^D$ setting where DFC +Doc-Rel achieves better performance on 12 classification tasks.

Classification without Filtering.

For classification without filtering, we evaluate all the methods on both datasets: 20 categories in the 20NG dataset and 10 categories in the Reuters-10 dataset. We also create 7 classification tasks based on the 20NG dataset, by selecting the documents in subsets of all categories. For example, one of the tasks is to classify documents in categories pc and mac, denoted by pc-mac. These 7 classification tasks were used in the works of TLC++ and STM for the evaluation [Hingmire and Chakraborti 2014; Li et al. 2016b]. In total, we have 9 classifications tasks involving different number of categories on the two datasets. Tables VI and VII report the Macro-F_1 scores of these methods on the 9 tasks by using $\mathbb{S}^L$ and $\mathbb{S}^D$ respectively. We make the following observations:

First, with $\mathbb{S}^D$ as the seed words, DFC +Doc-Rel significantly outperforms other state-of-the-art dataless alternatives on 8 out of 9 classification tasks. SNB-EM+$\mathbb{S}^D$ performs the second best on 5 classification tasks, and DFC +Topic-Rel performs the best on a single task and second best on the other 2 tasks. GE-FL, DescLDA and TLC++ achieve worse but close performances to each other. The

Table V. Macro-F_1 of the six methods for classification with filtering, where the seed words in $\mathbb{S}^D$ are used. The best and second best results are highlighted in boldface and underlined respectively, on each task. † indicates that the difference to the best dataless classifier is statistically significant at 0.05 level. ▲ and ▼ indicate that the supervised classifiers perform better or worse than the best dataless classifier respectively. Avg: the averaged Macro-F_1 over all tasks.

Dataset	Classification task	DFC		TLC++	SSVM	MedLDA	sLDA	SVM
		Doc-Rel	Topic-Rel					
20NG	med	**0.863**	0.820†	0.004†	0.159	0.256†▼	0.701†▼	0.190▼
	space	**0.861**	0.758†	0.027†	0.181	0.304†▼	0.763†▼	0.255▼
	sci	**0.716**	0.639†	0.007†	0.349	0.426†▼	0.755†▲	0.370▼
	religion	**0.872**	0.571†	0.500†	0.375	0.390†▼	0.857▼	0.409▼
	med-space	**0.866**	0.823†	0.047†	0.170	0.333†▼	0.737†▼	0.222▼
	pc-mac	**0.426**	0.283†	0.019†	0.174	0.350†▼	0.362†▼	0.208▼
	politics-religion	**0.899**	0.771†	0.311†	0.395	0.564†▼	0.861†▼	0.423▼
	politics-sci	**0.811**	0.693†	0.285†	0.382	0.581†▼	0.797▼	0.404▼
	comp-religion-sci	**0.793**	0.743†	0.448†	0.392	0.637†▼	0.850†▲	0.418▼
	politics-rec-religion-sci	**0.853**	0.777†	0.578†	0.387	0.705†▼	0.848▼	0.407▼
	autos-motorcycles-baseball-hockey	**0.864**	0.850	0.262†	0.266	0.487†▼	0.782†▼	0.262▼
Reuters-10	gold	**0.818**	0.620†	0.000†	0.167	0.000†▼	0.620†▼	0.333▼
	earn	0.908†	0.942†	**0.943**	0.775	0.489†▼	0.984†▲	0.822▼
	acq-earn	0.803†	0.919†	**0.940**	0.636	0.653†▼	0.980†▲	0.636▼
	coffee-gold-sugar	0.731†	**0.820**	0.093†	0.310	0.121†▼	0.764†▼	0.574▼
Avg		0.806	0.735	0.298	0.341	0.420	0.777	0.396

Table VI. Macro-F_1 of the eight methods for classification without filtering, where the seed words in $\mathbb{S}^L$ are used. The best and second best results by dataless classifiers are highlighted in boldface and underlined respectively, on each task. † indicates that the difference to the best dataless classifier is statistically significant at 0.05 level. ▲ and ▼ indicate that the supervised classifiers perform better or worse than the best dataless classifier respectively. Avg: the averaged Macro-F_1 over all tasks.

Dataset	Classification task	DFC		Ge-FL	DescLDA	SNB-EM	TLC++	MedLDA	sLDA	SVM
		Doc-Rel	Topic-Rel							
20NG	med-space	0.967	**0.972**	0.712†	0.877†	0.897	0.938†	0.975▲	0.910†▼	0.976▲
	pc-mac	**0.902**	0.678†	0.491†	0.688†	0.895	0.685†	0.881†▼	0.735†▼	0.925▲
	politics-religion	0.907	0.506†	0.684†	0.888†	0.894	**0.911**	0.949†▲	0.925†▲	0.954▲
	politics-sci	**0.960**	0.746†	0.750†	0.624†	0.846	0.906†	0.941†▼	0.930†▼	0.971▲
	comp-religion-sci	**0.918**	0.857†	0.709†	0.559†	0.907	0.817†	0.930▲	0.900▼	0.936▲
	politics-rec-religion-sci	**0.919**†	0.742†	0.719†	0.514†	0.768	0.834†	0.932▲	0.823†▼	0.941▲
	autos-motorcycles-baseball-hockey	0.936†	**0.957**	0.849†	0.531†	0.715	0.734†	0.962▲	0.894†▼	0.957
	All 20 categories	**0.662**†	0.572†	0.320†	0.632†	0.461	0.510†	0.705†▲	0.633†▼	0.820▲
Reuters-10	All 10 categories	**0.701**†	0.496†	0.667†	0.317†	0.529	0.506†	0.562†▼	0.754†▲	0.932▲
Avg		0.875	0.725	0.656	0.626	0.768	0.760	0.871	0.834	0.935

superiority of SNB-EM is attributed to its semi-supervised nature. After an initial NB-EM classifier is built based on the seed words, SNB-EM retrains itself by using the classification results of high confidence, and this procedure repeats until the probability parameters stabilize. All the dataless classifiers obtain relatively poorer results when all the categories are used in the classification tasks, *i.e.,* 20 categories on 20NG and 10 categories on Reuters-10. DFC +$\mathbb{S}^D$ +Doc-Rel achieves the best performance over the other alternatives in these two tasks. Although TLC++ and GE-FL achieves comparable performance in the other tasks when $\mathbb{S}^D$ is used, GE-FL performs much better here. We observe that when the number of categories is larger, the resultant topics produced by LDA often carry mixed semantic information from more than one category. It even becomes difficult for annotators to manually associate topics to the relevant categories for TLC++. In this sense, TLC++ experiences a significant performance deterioration. Overall on these 9 classification tasks, DFC +$\mathbb{S}^D$

Table VII. Macro-F_1 of the eight methods for classification without filtering, where the seed words in $\mathbb{S}^D$ are used. The best and second best results by dataless classifiers are highlighted in boldface and underlined respectively, on each task. † indicates that the difference to the best dataless classifier is statistically significant at 0.05 level. ▲ and ▼ indicate that the supervised classifiers perform better or worse than the best dataless classifier respectively. Avg: the averaged Macro-F_1 over all tasks.

Dataset	Classification task	DFC		Ge-FL	DescLDA	SNB-EM	TLC++	MedLDA	sLDA	SVM
		Doc-Rel	Topic-Rel							
20NG	med-space	0.972	**0.979**	$0.935^{\dagger}$	0.977	0.967	$0.938^{\dagger}$	$0.975_{\blacktriangle}$	$0.910^{\dagger}{}_{\blacktriangledown}$	$0.976_{\blacktriangledown}$
	pc-mac	**0.936**	$0.416^{\dagger}$	$0.705^{\dagger}$	$0.694^{\dagger}$	0.876	$0.685^{\dagger}$	$0.881^{\dagger}{}_{\blacktriangledown}$	$0.735^{\dagger}{}_{\blacktriangledown}$	$0.925_{\blacktriangledown}$
	politics-religion	**0.952**	0.935	$0.883^{\dagger}$	$0.900^{\dagger}$	0.939	$0.911^{\dagger}$	$0.949_{\blacktriangledown}$	$0.925^{\dagger}{}_{\blacktriangledown}$	$0.954_{\blacktriangle}$
	politics-sci	**0.962**	$0.912^{\dagger}$	$0.889^{\dagger}$	$0.912^{\dagger}$	0.941	$0.906^{\dagger}$	$0.941^{\dagger}{}_{\blacktriangledown}$	$0.930^{\dagger}{}_{\blacktriangledown}$	$0.971_{\blacktriangle}$
	comp-religion-sci	**0.923**	$0.861^{\dagger}$	$0.828^{\dagger}$	$0.498^{\dagger}$	0.919	$0.817^{\dagger}$	$0.930_{\blacktriangle}$	$0.900^{\dagger}{}_{\blacktriangledown}$	$0.936_{\blacktriangle}$
	politics-rec-religion-sci	**0.941**	$0.914^{\dagger}$	$0.827^{\dagger}$	$0.782^{\dagger}$	0.917	$0.834^{\dagger}$	$0.932_{\blacktriangledown}$	$0.823^{\dagger}{}_{\blacktriangledown}$	0.941
	autos-motorcycles-baseball-hockey	**0.977**	0.957	$0.673^{\dagger}$	$0.713^{\dagger}$	0.938	$0.734^{\dagger}$	$0.962_{\blacktriangledown}$	$0.894^{\dagger}{}_{\blacktriangledown}$	$0.957_{\blacktriangledown}$
	All 20 categories	**0.739**	0.728	$0.590^{\dagger}$	$0.663^{\dagger}$	0.678	$0.510^{\dagger}$	$0.705^{\dagger}{}_{\blacktriangledown}$	$0.633^{\dagger}{}_{\blacktriangledown}$	$0.820_{\blacktriangle}$
Reuters-10	All 10 categories	**0.822**	$0.791^{\dagger}$	$0.776^{\dagger}$	$0.800^{\dagger}$	0.778	$0.506^{\dagger}$	$0.562^{\dagger}{}_{\blacktriangledown}$	$0.754^{\dagger}{}_{\blacktriangledown}$	$0.932_{\blacktriangle}$
Avg		0.914	0.833	0.790	0.771	0.884	0.760	0.871	0.834	0.935

+Doc-Rel outperforms GE-FL+$\mathbb{S}^D$, DescLDA+$\mathbb{S}^D$, SNB-EM+$\mathbb{S}^D$, and TLC++ by around 15.7%, 18.5%, 3.4% and 20.2% on average, respectively.

Second, the supervised learning methods SVM, MedLDA and sLDA may not always perform better than the dataless counterparts. With $\mathbb{S}^D$, dataless classifier outperforms SVM in 3 out of 9 classification tasks: DescLDA+Topic-Rel and DescLDA on med-space, DFC +Doc-Rel on both pc-mac and autos-motorcycles-baseball-hockey. Moreover, DFC +$\mathbb{S}^D$ +Doc-Rel consistently outperforms sLDA on all the classification tasks. Even DFC +$\mathbb{S}^L$ +Doc-Rel performs significantly better than sLDA on 7 out of 9 classification tasks, although sLDA is a supervised classification method. MedLDA outperforms sLDA in 8 out of 9 classification without filtering tasks. However, DFC +$\mathbb{S}^D$ +Doc-Rel still outperforms MedLDA in 7 out of 9 tasks. Note that sLDA only achieves a Macro-F_1 of 0.735 on *pc-mac*, where a large number of words are shared by the two categories [Chakraborti et al. 2008]. DFC +$\mathbb{S}^D$ +Doc-Rel outperforms sLDA, MedLDA and SVM on this task by 27.3%, 6.2% and 1.1% respectively, given that $1,168$ labeled documents are used to train the the supervised classifiers. Further, DFC +$\mathbb{S}^D$ +Doc-Rel achieves very close or even comparable performance with SVM on 4 classificaiton tasks: politics-religion, politics-sci, comp-religion-sci, politics-rec-religion-sci. Although SVM performs better than DFC +$\mathbb{S}^D$ when all the categories are used in the classification tasks on both datasets, the performance gap is not very large. Shown in Tables II and III, only a small number of seed words are used to "train" the DFC classifier. Compared with the much larger number of labeled documents used to train SVM, minimum human efforts are necessary to learn dataless classifiers like DFC. In this sense, the proposed DFC +Doc-Rel can work as an important complement to the existing supervised solutions. This is important in the scenario where the existing system only requires labeled documents rather than classifiers.

Third, with $\mathbb{S}^D$, DFC, GE-FL, DescLDA and SNB-EM all perform better than their counterparts with $\mathbb{S}^L$ on the majority of the tasks (the study about seed words is presented in Section 4.5). The performance gain by using $\mathbb{S}^D$ over $\mathbb{S}^L$ is about 4.5%, 14.8%, 20.4%, 23.2%, and 15.1% for DFC +Doc-Rel, DFC +Topic-Rel, GE-FL, DescLDA, and SNB-EM on average, respectively. Observe that the performance gap between DFC +$\mathbb{S}^D$ +Doc-Rel and DFC +$\mathbb{S}^L$ +Doc-Rel is also the smallest, compared to the other alternatives. This is consistent with the observation made in classification with filtering. When $\mathbb{S}^L$ is used, TLC++ outperforms GE-FL and DescLDA on most tasks. This is reasonable since each topic in TLC++ is manually examined and labeled based on its 30 most probable words, which is equivalent to labeling 30 relevant words for each topic. It is expected that more human efforts leads to better classification accuracy, all else being equal. As to DFC, DFC +$\mathbb{S}^L$ +Doc-Rel outperforms GE-FL+$\mathbb{S}^L$/$\mathbb{S}^D$, DescLDA+$\mathbb{S}^L$, SNB-EM+$\mathbb{S}^L$ on at least 8 classification tasks, and outperforms TLC++, DescLDA+$\mathbb{S}^D$ on 8 and 6 tasks respectively. As reported in the last columns

of Tables II and III, almost all categories studied here have fewer than 5 seed words in $\mathbb{S}^L$, except for the major category *comp* which has 13 $\mathbb{S}^L$ seed words. With just a few semantically relevant words, DFC +$\mathbb{S}^L$ +Doc-Rel can deliver promising classification performance. The superiority suggests that DFC +Doc-Rel is not very sensitive to the number of seed words.

Fouth, DFC +Doc-Rel outperforms DFC +Topic-Rel on the majority of the classification tasks when either $\mathbb{S}^L$ or $\mathbb{S}^D$ is used. DFC +Topic-Rel only obtains better performance on 3 classification tasks. However, the performance gain on these tasks are marginal or small. Also, DFC +$\mathbb{S}^D$ +Topic-Rel performs much better than DFC +$\mathbb{S}^L$ +Topic-Rel on almost all the calssification tasks, as being discussed above. Compared with the performance reported for classification with filtering in Tables IV and V, it is obvious that Topic-Rel is more suitable for classification with filtering. Note that DFC +Topic-Rel estimates the category word probabilities with the LDA hidden topics for both kinds of tasks. The only difference is that, in classification with filtering, the pseudo seed words are extracted for the irrelevant-topics based on the same LDA hidden topics. That is, both the pseudo seed words and the category word probility are infered from the same source. In this sense, the category word probilitise for the irrelevant-topics could be estimated more precise since the pseudo seed words match well with the LDA hidden topics. We believe that the mismatch between the LDA topics and provided seed words leads to much noisy information for DFC +Topic-Rel in classification without filtering tasks. Therefore, we can incorporate the provided seed words into LDA hidden topic inference process for better category word probability estimation, leading to better classification without filtering performance for DFC +Topic-Rel. Several existing works discussed in Section 2.2 can be utilized here by making the word-to-word pairs contraint [Andrzejewski et al. 2009; Chen et al. 2013]. We leave this exporation in our future work.

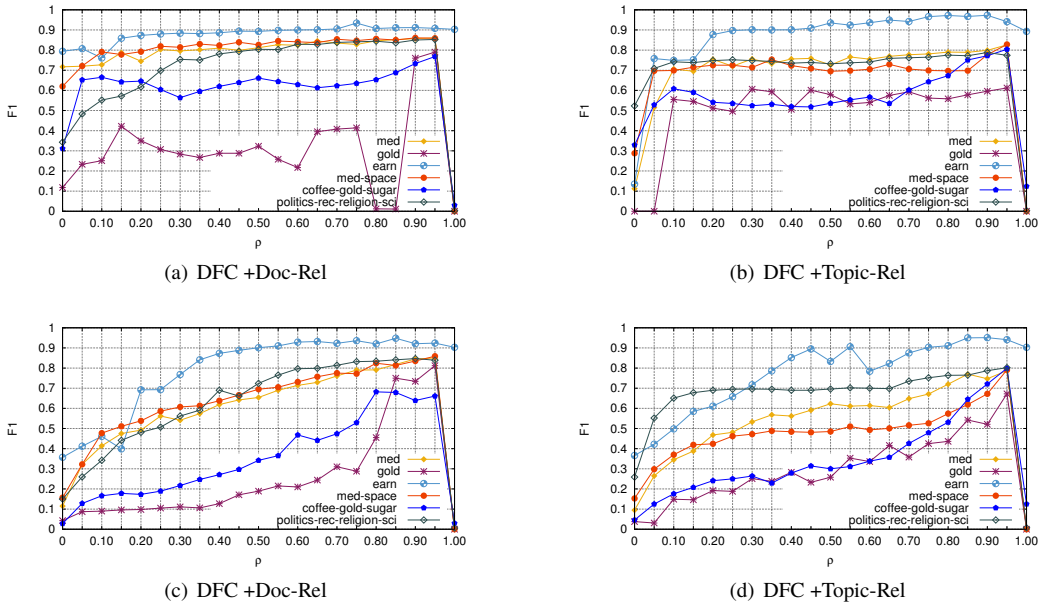
Overall, the experimental results show that DFC delivers promising classificaton accuracy with a few seed words from either the category label or its descripton, for both classification with filtering and classification without filtering.

4.4. Analysis of STM

We now investigate the impact of the parameter settings (*i.e.,* ρ, T, B, α_2) in DFC by using the $\mathbb{S}^D$ setting and classification with filtering. Both two relevance estimation mechanisms (*i.e.,* Doc-Rel and Topic-Rel) are investigated. As to parameter $\tau_{w,c}$, our earlier work validates its impact in classification without filter [Li et al. 2016b]. Since it works exactly the same for DFC in classification with filtering, we omit the repeation of the analysis. Similarly, the parameter analysis for classification without filtering is excluded since the analysis is conducted in our earlier work as well. Here, we also study the convergence rate of DFC in terms of classification performance. Specifically, we report the experimental results by varying one specific parameter value on these 6 classification with filtering tasks: med, gold, earn, med-space, coffee-gold-sugar, and politics-rec-religion-sci, ranging from 1 to 4 specified categories. Similar performance patterns are also observed for other classification tasks studied in Section 4.3. Note that when studying a specific parameter, we set the other parameters to the values used in Section 4.3.

Impact of ρ value. Recall in Equation 16, ρ controls the importance of the relevance weight $\tau_{w,c}$ for category word probability estimation. This relevance weight is measured based on the word occurrence information between word w and the (pseudo) seed words of category-topic c. As discussed in Section 3.1, $\rho = 0$ is equivalent to using a standard LDA with B general-topics. In this setting, DFC only relies on the general-topic distribution θ_d and category general-topic distribution φ_c to filter and classify document d in this setting. On the other hand, DFC has no general-topic at all when $\rho = 1$, and is equivalent to TLC++ model proposed in [Hingmire and Chakraborti 2014]. The only difference is that the category-topic is solely guided by the seed words in this setting. Figure 5(a) and Figure 5(b) plot the performance patterns when using different ρ values in the range of $[0, 1]$ with a step of 0.05 for DFC +Doc-Rel and DFC +Topic-Rel respectively.

Observe that Both DFC +Doc-Rel and DFC +Topic-Rel achieve stable performance on 4 classification tasks: med, earn, med-space, politics-reigion-sci, in the range of $[0.40, 0.95]$. On the tasks



(a) DFC +Doc-Rel (b) DFC +Topic-Rel

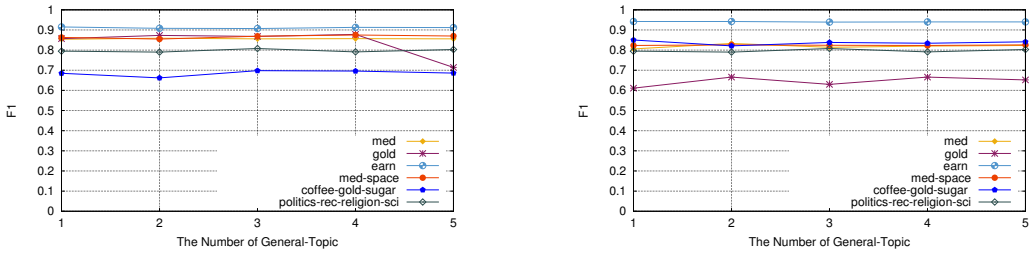
(c) DFC +Doc-Rel (d) DFC +Topic-Rel

Fig. 5. Performance of DFC +Doc-Rel and DFC +Topic-Rel with varying ρ values when $\alpha_2 = 100$ ((a) and (b)), and when $\alpha_2 = 1$ ((c) and (d)).

for very smaller categories (*i.e.,* gold, coffee-gold-sugar), the performance becomes unstable. For coffee-gold-sugar, the classification performance increases as ρ becomes larger under both Doc-Rel and Topic-Rel settings. However, for task gold, the situation becomes quite different for the two relevance estimation mechanisms. DFC +Doc-Rel experiences a large performance fluctuation with varying ρ values. The performance even becomes very low in the range of $[0.80, 0.85]$. In contrast, DFC +Topic-Rel experiences relatively modest performance variations with ρ values, in the range of $[0.30, 0.95]$ for ρ values. Category gold is the smallest category with only 20 testing documents. Therefore, the decision change for a gold document could incur a relatively big change in Macro-F_1 score. In this sense, we could consider that DFC +Topic-Rel already achieves stable performance in a wide range of ρ values. We believe that the large performance variation by Doc-Rel is attributed to the severe sparsity of category gold. Since the number of documents belonging to gold is very small, the relevance $rel(w, c)$ estimated by Doc-Rel based on the conditional probability is not reliable and much noisy information is introduced (ref. Equation 11). On the other hand, the fine-grained LDA hidden topics relevant to gold contains less noisy information, leading to much more stable performance across a wide range of ρ values.

Another observation is that DFC +Doc-Rel/Topic-Rel performs significantly worse when $\rho < 0.10$ or $\rho = 1.0$ on most tasks. That is, without general-topics ($\rho = 1$) or category-topics ($\rho = 0$), DFC loses its ability to learn the discriminative information for each category. This validates the legitimacy of using both general-topics and category-topics in DFC for dataless filtering and classification. This finding is consistent with our earlier work in [Li et al. 2016b] for classification without filtering. Note that the optimal ρ value is almost identical across the tasks from the two datasets. Both DFC +Doc-Rel and DFC +Topic-Rel achieve the optimal performance when $\rho = 0.95$. This is essentially valuable for real applications. In our experiments, we use this setting.

Impact of B value. The B value specifies the number of general-topics used in DFC. In the above experiments, we fix B to be 3 times of the number of category-topics. Here, we evaluate the effect of B value by setting it to be 1 to 5 times of the number of category-topics. Figure 6(a) and Figure 6(b)



(a) DFC +Doc-Rel (b) DFC +Topic-Rel

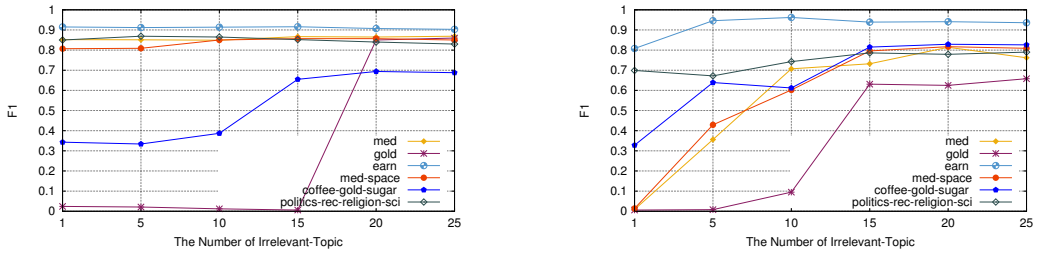
Fig. 6. Performance of DFC +Doc-Rel and DFC +Topic-Rel with varying B values.

plot the performance of DFC with different B values under Doc-Rel and Topic-Rel respectively. We observe that DFC is insensitive to the choice of parameter B. There is little performance variations in DFC when we use different B values across the tasks. Based on the results, we set $T = 3 \cdot R$ for DFC in our experiments.

Impact of T value. The T value specifies the number of irrelevant-topics used in DFC to model irrelevant documents. Since 20NG and Reuters-10 contains 20 and 10 categories respectively, we evaluate the impact of using different T values in the range of $\{1, 5, 10, 15, 20, 25\}$. Note that setting $T = 0$ is equivalent to the setting of $\rho = 1$. The inferiority with no general-topic in DFC is already validated in the analysis of ρ. Figure 7(a) and Figure 7(b) plot the performance patterns for DFC +Doc-Rel and DFC +Topic-Rel respectively. Except for the tasks for the smallest categories, DFC achieves stable performance for other 4 tasks under Doc-Rel setting across all T values. In contrast, DFC experiences a large performance deterioration for all 6 tasks under Topic-Rel setting when T is very small (*i.e.,* $T \in \{1, 5, 10\}$). This observation is reasonable since Topic-Rel estimates the category word probility based on the LDA hidden topics and the pseudo seed words which are also extracted according to the former. In this sense, the pseudo seed words under Topic-Rel are strongly restricted to few irrelevant categories covered by the corresponding LDA hidden topics. When T is smaller than the true number of irrelevant categories covered by the document collection, some unknown categories may not be modeled well by the irrelevant-topics, leading to significant performance loss. In comparison, Doc-Rel does not couple the category word probability estimation and the pseudo seed word extraction with the LDA hidden topics together. The document-level word co-occurrence information utilized in Doc-Rel is more coarse-grained than the LDA hidden topics. Thus, the category word probitlity estimated in Doc-Rel could cover a much broader range of irrelevant categories.

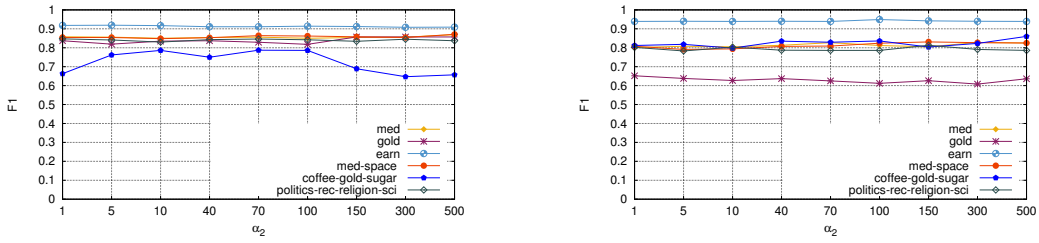
Another observation is that both tasks gold and coffee-gold-sugar experience a lot of performance loss when $T < 20$ under both Doc-Rel and Topic-Rel settings. The performance loss by using Topic-Rel can be well explained by the reason discussed above. It is obvious that DFC +Topic-Rel obtains much better performance than DFC +Doc-Rel on these two tasks when T is set to 10 and 15. The inferiority in these cases is attributed to the conditional probability used in Doc-Rel (ref. Equation 11). As being discussed in the analysis of ρ, the number of documents belonging to category gold is very small. Hence, the relevance $rel(w, c)$ estimated by the conditional probability is error-prone. Based on the results, we set $T = 20$ in our experiments.

Impact of α_2 value. The concentration parameter α_2 controls the degree that the general-topic distribution θ_d of document d of category c can deviate from the general-topic distribution φ_c of that category. When α_2 is very large, each document of category c has almost the identical general-topic distribution θ_d. On the other hand, $\alpha_2 \to 0$ is equivalent to assigning each document a general-topic distribution without the category constraint. Here, we investigate the performance of DFC by varying α_2 values in the range of $[1, 500]$. The experimental results are plotted in Figure 8(a) and Figure 8(b)



(a) DFC +Doc-Rel (b) DFC +Topic-Rel

Fig. 7. Performance of DFC +Doc-Rel and DFC +Topic-Rel with varying T values.

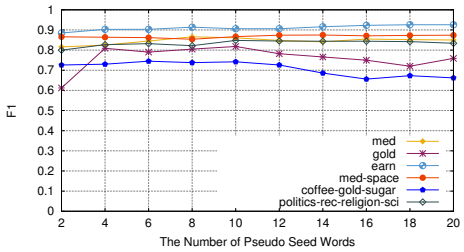


(a) DFC +Doc-Rel (b) DFC +Topic-Rel

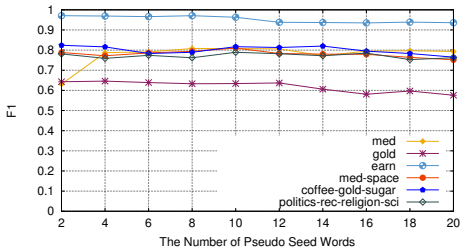
Fig. 8. Performance of DFC +Doc-Rel and DFC +Topic-Rel with varying α_2 values.

for Doc-Rel and Topic-Rel respectively. It shows that the performance of DFC is stable in a broad range of α^2 values across most tasks. Most classification tasks experience very little performance fluctuation when using different α_2 values. It seems that the category constraint is less important. However, when α_2 is set to 1, we observe significant performance degradation with varying ρ values. Figure 5(c) and Figure 5(d) plot the performance patterns by using different ρ values and $\alpha_2 = 1$ for DFC +Doc-Rel and DFC +Topic-Rel respectively. Compared with the results demonstrated in Figure 5(a) and Figure 5(b), it is obvious to see that both DFC +Doc-Rel and DFC +Topic-Rel deliver relatively much lower performance in a wide range of ρ values when $\alpha_2 = 1$. The optimal performance is only achieved by using $\rho = 0.95$. This suggests that the category constraint is indeed helpful in DFC. Based on the results, we set $\alpha_1 = 100$ in our experiments.

Impact of the number of pseudo seed words. In Section 3.2, we extract top-10 topical words of each least relevant LDA hidden topis with respect to the provided seed words as the pseudo seed words for an irrelevant-topic. Here, we investigate the impact of the number of pseudo seed words to the classification performance of DFC. Figure 9(a) and Figure 9(b) plot the performance of DFC with different number of iterations under Doc-Rel and Topic-Rel settings respectively. We can see that DFC can experience little performance variation in the range of $[2, 12]$ for most tasks. With more topical words, the performance of DFC degrades significantly when classifying and filtering the smaller categories like gold,coffee,sugar. In contrast, no performance deterioration is experienced with more pseudo seed words for the other categories. This is reasonable since smaller categories covering few documents, resulting in the unreliable word co-occurrence information. In this sense, taking more pseudo seed words for the irrelevant-topics would misclassify the documents belonging to the specified categories. Based on the results, we take top-10 topical words as the pseudo seed words in our experiments.

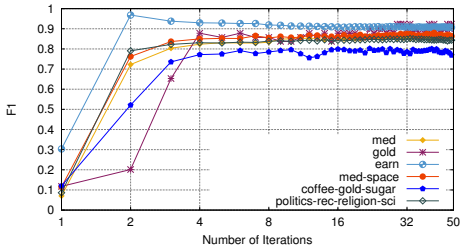


(a) DFC +Doc-Rel

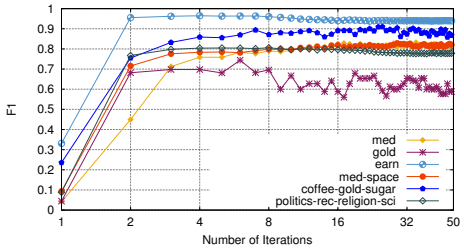


(b) DFC +Topic-Rel

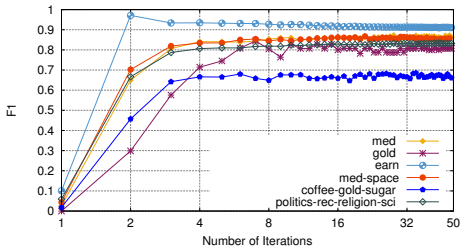
Fig. 9. Performance of DFC +Doc-Rel and DFC +Topic-Rel with different number of pseudo seed words per irrelevant-topic



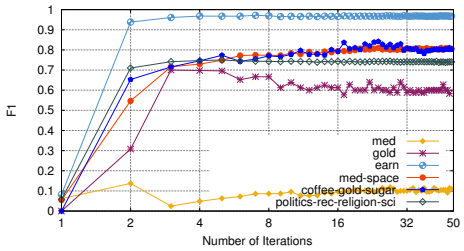
(a) DFC +Doc-Rel



(b) DFC +Topic-Rel



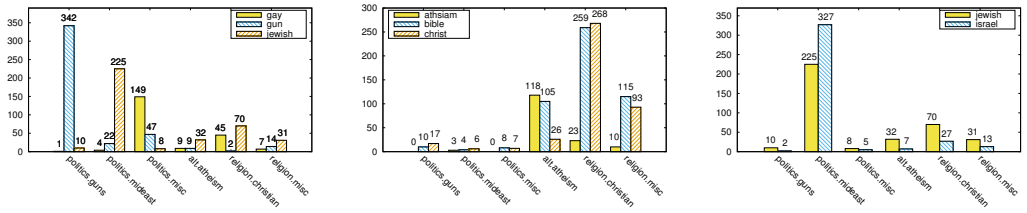
(c) DFC +Doc-Rel



(d) DFC +Topic-Rel

Fig. 10. Performance of DFC +Doc-Rel and DFC +Topic-Rel with different number of iterations when η_d is used ((a) and (b)), and when η_d is not used ((c) and (d)).

Impact of the number of iterations. We like to investigate the impact of the number of iterations to the classification performance of DFC. Figure 10(a) and Figure 10(b) plot the performance of DFC with different number of iterations under Doc-Rel and Topic-Rel settings respectively. we can see that DFC can achieve near-optimal performance after only 4 iterations. The stable performance is reached with about 8 iterations on most tasks. This suggests that DFC can successfully exploiting the semantic information provided by the seed words in an efficient manner, just like what humans are capable of. We further investigate the impact of estimating the initial category distribution (*i.e.,* η_d in Equation 19) for each document based on the seed words. Figure 10(c) and Figure 10(d) plot the performance of DFC under Doc-Rel and Topic-Rel settings respectively, when η_d is set to be a uniform distribution. We find that the classification performance is not affected significantly if this initial distribution estimation is not provided. However, DFC takes more iterations to achieve the stable performance.



(a) politics: "gay", "gun" and "jewis" (b) religion: "atheism", "bible" and "christ" (c) politics: jewish and israel

Fig. 11. The number of documents that a seed word appears in for two major categories: politics and religion.

4.5. Study on Seed Words

As being the only supervision for dataless text classifiers, the seed words play an important role in determining the classification accuracy. Unfortunately, the impact of seed words is still unknown. The existing works take the seed words either from the category labels/description or by human annotation. An illusive hypothesis is that the number of seed words is positively correlated to the classification accuracy. That is, more seed words would bring more supervision information, and hence result in better classification performance. In this section, we conduct a comprehensive study with the aim of answering the following two questions:

Q1 Will using more seed words lead to better classification accuracy?
Q2 What criterion should we use to build a set of seed words for a category?

Here, we take the classification without filtering task POLITICS-RELIGION as a running example to answer the above two questions. As listed in Appendix, there are 13 and 12 seed words from $\mathbb{S}^D$ for categories POLITICS and RELIGION respectively. By using Doc-Rel mechanism, Macro-F_1 score achieved by DFC for this task is 0.952 (ref. Tabel VII). We then iteratively remove seed words one by one for the two categories with the constraint that no performance deterioration is incurred with the updated seed word set. At the end, we obtain three seed words for each category: "gay, jewish, gun" for politics, and "christ, atheism, bible" for religion. That is, only 6 seed words are required for DFC to classify $5,049$ documents into two categories with a Macro-F_1 of 0.952. Note that both POLITICS and RELIGION contain three sub-categories respectively (ref. Table II). Hence, we can take these sub-categories as the coarse-grained topics covered by the two categories. We then check the coverage of each seed word over these topics. Figures 11(a) and 11(b) depict the number of documents supported by these three seed words for POLITICS and RELIGION respectively, zooming in on their sub-categories. It is clear that each seed word covers mainly a specific sub-category for the two major categories. Also, these seed words have strong discriminative power since their coverage in the competing category is small. For example, seed word "gun" appears in 342 documents of sub-category POLITICS.GUNS, and only covers 25 documents for category RELIGION. All topics are well covered by these three seed words for both two major categories respectively. Table VIII lists the classification performance when one or two seed words are removed further from these 6 words. We can observe that the classification performance deteriorates significantly when one or two seed words are removed. For example, without seed word "gun", "gay" and "jewish" cover just a small number of documents in sub-category POLITICS.GUNS. It becomes hard for DFC to classify the documents of POLITICS.GUNS, resulting in a Macro-F_1 of 0.895 only. The same observation is made when the coverage of a specific sub-category of RELIGION shrinks sharply by removing the corresponding two seed words.

Word "israel" is a seed word for category POLITICS. Figure 11(c) plots the number of documents supported by "israel". It has a document coverage similar to seed word "jewish". However, we obtain an unchanged Macro-F_1 of 0.952 when "israel" is added as the fourth seed words for POLITICS. Also,

Table VIII. Macro-F_1 of DFC for politics-religion with different seed words.

Seed words	-	-{gun}	-{gay}	-{jewish}	-{bible,christ}	-{bible,atheism}
gay jewish gun christ atheism bible	0.952	0.895	0.930	0.940	0.830	0.940

Table IX. Macro-F_1 of the 4 methods for politics-religion with different seed words. new-seed: new seed words are used. The best and second best results by dataless classifiers are highlighted in boldface and underlined respectively, on each task.

Method	politics-religion			comp-religion-sci		
	$\mathbb{S}^L$	$\mathbb{S}^D$	new-seed	$\mathbb{S}^L$	$\mathbb{S}^D$	new-seed
GF-FL	0.684	**0.883**	0.843	0.709	**0.828**	0.811
DescLDA	0.888	0.900	**0.936**	0.559	0.498	**0.893**
SNB-EM	0.894	0.939	**0.943**	0.907	0.919	**0.932**
DFC	0.907	**0.952**	**0.952**	0.918	**0.927**	0.923

no performance change is experienced when we replace "jewish" with "israel". It seems the answers to Q1 and Q2 is clear now. However, the above empirical study is conducted by using DFC only. We further conduct experiments by using other 3 dataless text classifiers: GE-FL, DescLDA and SNB-EM, and over two classification without filtering tasks: POLITICS-RELIGION and COMP-RELIGION-SCI. For COMP-RELIGON-SCI, the procedure mentioned above is applied to refine the seed words. There are 71 original seed words in $\mathbb{S}^D$ for these three major categories in total. After the seed word refinement, we obtain only 12 new seed words instead. Table XIV in Appendix reports the number of documents supported by these 12 seed words in terms of sub-categories. Similarly, we can see that each seed word mainly cover a specific sub-categories. Table IX reports the performance comparison when new seed words are used. We can see that the new seed words deliver even better classification performance on the majority of the settings. The same conclusion can be reached on other tasks in our study. By using few seed words with a larger target coverage, the potential noisy information by using more seed words could be reduced significantly. This can explain the better classification performance obtained with the new seed words in these tasks. The same observation is also made in the experimental comparison in Section 4.3, where using $\mathbb{S}^L$ delivers better performance than using $\mathbb{S}^D$.

Overall, the experimental results demonstrate that the number of seed words is not positively correlated with the classification accuracy. Instead, the document coverage of the seed words for a category of interest plays a critial role in determining the dataless classification accuracy. This finding could lead us to devise several strategies to improve the seed word selection process. One simple solution is to calculate the distance between two seed words in terms of document distribution. Then the optimal set of seed words can be built by adopting optimization techniques, *e.g.,* MMR based approaches for search result diversification [Carbonell and Goldstein 1998; Santos et al. 2011]. We leave this line of work as a part of future work.

5. CONCLUSION

In this paper, we propose a seed-guided topic model for dataless text filtering and classification, named DFC. Without any labeled documents, DFC takes only a few seed words to retrieve the documents of the specified categorise through topic influence. By modeling the documents using both category-topics and general-topics, DFC successfully captures the diverse semantic information of the dataset by separating category-specific information and general semantic information. We investigate the two mechanisms to extract the discriminative category information by explicitly calculating the relevance between the seed words and regular words based on the word co-occurrences. To facilitate the accurate filtering, we devise a simple but effective mechanism to identify a set of pseudo seed words for each irrelevant-topic based on the LDA hidden topics extracted from the given document collection. The experimental results show that DFC outperforms existing state-of-the-art dataless text classifiers for both classification with filtering and classification without filtering. It is interesting to

observe that DFC even surpasses the state-of-the-art supervised classifier like sLDA and SVM on many tasks. The parameter analysis also validates the robustness of DFC against the different parameter settings. Also, we conduct an empirical study about how to choose the seed words for dataless text classification techniques. The experimental results based on several existing dataless classifiers suggest that the document coverage of the seed words under a category correlates positively with the classification accuracy.

Nevertheless, there is still room to improve our model in several directions. The first is the efficiency issue. As described in Section 3.1, DFC needs to sample the hidden parameters for all the category-topics (*i.e.,* relevant-topics and irrelevant-topics) for a document in one iteration. This sampling process comprises the main computation cost in DFC. Similar to the pruning strategy used in [Li et al. 2017], we can calculate the likelihood $p(c_d = t, y_d = 0|d)$ and $p(c_d = r, y_d = 1|d)$ based on Equations 2 and 3 for each document d after h iterations. Then we restrict the category-topic sampling space to the top-m category-topics (*i.e.,* $m \ll R + T$) for each document in the next h iterations. Secondly, as discussed in Section 4.3, DFC +Topic-Rel does not perform well for classification without filtering. The mismatch between the LDA hidden topics and the provided seed words incurs much noisy information in the category word probability estimation. Hence, we like to investigate the existing topic models incorporating the word-level constraints into the topic inference for DFC in classification without filtering. Thirdly, in Section 4.5, we learn the selection criterion of the seed words for the dataless text classification techniques. We plan to devise the seed word selection strategies by following the Maximal Marginal Relevance (MMR) based practices in the search result diversification area.

APPENDIX

In this appendix, we list the seed words in $\mathbb{S}^L$ and $\mathbb{S}^D$ for the two datasets respectively in Tables X-XIII. The number of documents that a seed word appear in for three major categories in 20NG is reported in Table XIV.

ACKNOWLEDGMENTS

This research was supported by National Natural Science Foundation of China (No.61502344), Natural Science Foundation of Hubei Province (No.2017CFB502), Natural Scientific Research Program of Wuhan University (No.2042017kf0225, No.2042016kf0190), Academic Team Building Plan for Young Scholars from Wuhan University (No. Whu2016012) and Singapore Ministry of Education Academic Research Fund Tier 2 (MOE2014-T2-2-066).

REFERENCES

David Andrzejewski, Xiaojin Zhu, and Mark Craven. 2009. Incorporating domain knowledge into topic modeling via Dirichlet Forest priors. In *ICML*. 25–32.

David M. Blei and Jon D. McAuliffe. 2007. Supervised Topic Models. In *NIPS*. 121–128.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3 (2003), 993–1022.

Chris Buckley and Gerard Salton. 1995. Optimization of Relevance Feedback Weights. In *SIGIR*. 351–357.

Jaime G. Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *SIGIR*. 335–336.

Sutanu Chakraborti, Ulises Cerviño Beresi, Nirmalie Wiratunga, Stewart Massie, Robert Lothian, and Deepak Khemani. 2008. Visualizing and Evaluating Complexity of Textual Case Bases. In *ECCBR*. 104–119.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of Semantic Representation: Dataless Classification. In *AAAI*. 830–835.

Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. 2006. Modeling General and Specific Aspects of Documents with a Probabilistic Topic Model. In *NIPS*. 241–248.

Table X. Seed words in $\mathbb{S}^L$ for 20NG.

Category	Seed Words
alt.atheism	atheism
comp.graphics	computer, graphics
comp.os.ms-windows.misc	computer, os, microsoft, windows
comp.sys.ibm.pc.hardware	computer, systems, ibm, pc, hardware
comp.sys.mac.hardware	computer, systems, hardware, macintosh, apple, mac,
comp.windows.x	computer, windows, windowsx
misc.forsale	sale
rec.autos	autos
rec.motorcycles	motorcycles
rec.sport.baseball	sport, baseball
rec.sport.hockey	sport, hockey
sci.crypt	science, cryptography
sci.electronics	science, electronics
sci.med	science, medicine
sci.space	science, space
soc.religion.christian	society, religion, christian
talk.politics.guns	politics, guns
talk.politics.mideast	politics, mideast
talk.politics.misc	politics
talk.religion.misc	religion
politics	politics, guns, mideast
religion	religion, atheism, christian, society, christianity
sci	medicine, space, science, cryptography, electronics
comp	graphics, os, microsoft, ibm, pc, computer, systems, apple, hardware, windows, mac, macintosh, windowsx
rec	cars, motorcycles, baseball, hockey

Table XI. Seed words in $\mathbb{S}^L$ for Reuters-10.

Category	Seed Words
acq	acquisition
coffee	coffee
crude	crude
earn	earnings
gold	gold
interest	interest
money-fx	foreign, exchange
ship	ship
sugar	sugar
trade	trade

Xingyuan Chen, Yunqing Xia, Peng Jin, and John A. Carroll. 2015. Dataless Text Classification with Descriptive LDA. In *AAAI*. 2224–2231.

Zhiyuan Chen and Bing Liu. 2014. Mining topics in documents: standing on the shoulders of big data. In *SIGKDD*. 1116–1125.

Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malú Castellanos, and Riddhiman Ghosh. 2013. Leveraging Multi-Domain Prior Knowledge in Topic Models. In *IJCAI*. 2071–2077.

Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by Latent Semantic Analysis. *JASIS* 41, 6 (1990), 391–407.

Doug Downey and Oren Etzioni. 2008. Look Ma, No Hands: Analyzing the Monotonic Feature Abstraction for Text Classification. In *NIPS*. 393–400.

Gregory Druck, Gideon S. Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In *SIGIR*. 595–602.

Mark D. Dunlop. 1997. The Effect of Accessing Nonmatching Documents on Relevance Feedback. *ACM Trans. Inf. Syst.* 15, 2 (1997), 137–153.

Table XII. Seed words in $\mathbb{S}^D$ for 20NG.

Category	Seed Words
alt.atheism	atheist, christian, atheism, god, islamic
comp.graphics	graphics, image, gif, animation, tiff
comp.os.ms-windows.misc	windows, dos, microsoft, ms, driver, drivers, card, printer
comp.sys.ibm.pc.hardware	motherboard, bios, board, computer, dos, bus, pc,
comp.sys.mac.hardware	mac, apple, powerbook
comp.windows.x	window, motif, xterm, sun, windows
misc.forsale	sale, offer, shipping, forsale, sell, price, brand, obo
rec.autos	car, ford, auto, toyota, honda, nissan, bmw
rec.motorcycles	bike, motorcycle, yamaha
rec.sport.baseball	baseball, ball, hitter
rec.sport.hockey	hockey, wings, espn
sci.crypt	encryption, key, crypto, algorithm, security
sci.electronics	circuit, electronics, radio, signal, battery
sci.med	doctor, medical, disease, medicine, patient
sci.space	space, orbit, moon, earth, sky, solar
soc.religion.christian	christian, god, christ, church, bible, jesus
talk.politics.guns	gun, fbi, guns, weapon, compound
talk.politics.mideast	israel, arab, jews, jewish, muslim
talk.politics.misc	gay, homosexual, sexual
talk.religion.misc	christian, morality, jesus, god, religion
politics	gun, fbi, weapon, compound, israel, arab, jews, jewish, muslim, gay, homosexual, sexual, guns
religion	atheism, atheist, christian, god, islamic, christ, church, bible, jesus, morality, jesus, religion
sci	crypto, encryption, key, algorithm, electronics, circuit, radio, signal, battery, medicine, doctor, medical, disease, patient, space, orbit, moon, earth, sky, solar, security
comp	computer, graphics, image, gif, animation, tiff, os, microsoft, windows, dos, ms, driver, card, printer, pc, bus, motherboard, bios, board, apple, mac, powerbook, sun, xterm, motif, window, drivers
rec	toyota, honda, nissan, bmw, motorcycle, bike, motorcycle, yamaha, baseball, ball, hitter, hockey, wings, espn, car, ford, auto

Table XIII. Seed words in $\mathbb{S}^D$ for Reuters-10.

Category	Seed Words
acq	acquisition, merger, cash, takeover, sale, agreement, asset, purchase, buy
coffee	coffee, export, ico, quota
crude	crude, petroleum, energy, barrel, opec, pipeline, bp, oil, gas,
earn	earning, net, income, loss, cost, profit, gain
gold	gold, mining, ounce, resource
interest	interest, bank, rate, money, debt, loan, bill
money-fx	foreign, exchange, currency, exchange, bank, rate, monetary, finance, budget, currency
ship	ship, port, cargo, river, seamen, refinery, water, vessel
sugar	sugar, tonne
trade	trade, foreign, agreement, export, goods, import, industry

Karla L. Caballero Espinosa and Ram Akella. 2012. Incorporating statistical topic information in relevance feedback. In *SIGIR*. 1093–1094.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *IJCAI*. 1606–1611.

Alfio Gliozzo, Carlo Strapparava, and Ido Dagan. 2009. Improving Text Categorization Bootstrapping via Unsupervised Learning. *ACM Trans. Speech Lang. Process.* 6, 1 (Oct. 2009), 1:1–1:24.

Yves Grandvalet and Yoshua Bengio. 2004. Semi-supervised Learning by Entropy Minimization. In

Table XIV. The number of documents that a seed word appears in for three major categories: comp, religion and sci. The largest number for each seed word is highlighted in boldface.

Category	Sub-category	graphics	windows	pc	apple	motif	atheism	christ	bible	encryption	circuit	medical	orbit
comp	comp.graphics	**304**	100	96	77	24	0	0	0	0	2	16	1
	comp.os.ms-windows.misc	60	**662**	139	31	3	1	0	1	1	10	6	0
	comp.sys.ibm.pc.hardware	31	156	**202**	26	0	0	0	0	1	11	2	0
	comp.sys.mac.hardware	25	20	48	**343**	1	0	0	2	1	5	4	2
	comp.windows.x	73	184	60	21	**184**	0	0	1	1	0	4	0
religion	alt.atheism	0	0	2	16	0	**118**	26	105	0	1	6	9
	soc.religion.christian	3	1	1	6	0	23	**268**	259	0	0	9	0
	talk.religion.misc	0	3	0	3	1	10	93	**115**	0	0	3	1
sci	sci.crypt	2	12	28	17	0	0	2	2	**399**	6	0	1
	sci.electronics	18	13	52	0	1	0	0	0	6	**137**	3	0
	sci.med	4	5	6	3	1	1	0	0	0	0	**186**	0
	sci.space	4	5	15	27	0	0	2	1	0	2	10	**203**

NIPS. 529–536.
Hu Guan, Jingyu Zhou, and Minyi Guo. 2009. A class-feature-centroid classifier for text categorization. In *WWW*. 201–210.
Swapnil Hingmire and Sutanu Chakraborti. 2014. Topic Labeled Text Classification: A Weakly Supervised Approach. In *SIGIR*. 385–394.
Swapnil Hingmire, Sandeep Chougule, Girish K. Palshikar, and Sutanu Chakraborti. 2013. Document Classification by Topic Labeling. In *SIGIR*. 877–880.
Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *SIGIR*. 50–57.
Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. 2012. Incorporating Lexical Priors into Topic Models. In *EACL*. 204–213.
Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. 2015. From Word Embeddings To Document Distances. In *ICML*. 957–966.
Chenliang Li, Yu Duan, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. 2017. Enhancing Topic Modeling for Short Texts with Auxiliary Word Embeddings. *ACM Trans. Inf. Syst.* 36, 2 (2017), 11:1–11:30.
Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. 2016a. Topic Modeling for Short Texts with Auxiliary Word Embeddings. In *SIGIR*. 165–174.
Chenliang Li, Jian Xing, Aixin Sun, and Zongyang Ma. 2016b. Effective Document Labeling with Very Few Seed Words: A Topic Model Approach. In *CIKM*. 85–94.
Bing Liu, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. 2004. Text Classification by Labeling Words. In *AAAI*. 425–430.
Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.
Jun Miao, Jimmy Xiangji Huang, and Jiashu Zhao. 2016. TopPRF: A Probabilistic Framework for Integrating Topic Space into Pseudo Relevance Feedback. *ACM Trans. Inf. Syst.* 34, 4 (2016), 22:1–22:36.
David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *EMNLP*. 262–272.
Arjun Mukherjee and Bing Liu. 2012. Aspect Extraction through Semi-Supervised Modeling. In *ACL*. 339–348.
Kamal Nigam, Andrew Kachites MacCallum, Sebastian Thrun, and Tom Mitchell. 2000. Text classification from labeled and unlabeled documents using EM. *Machine learning* 39, 2-3 (2000), 103–134.
Hema Raghavan, Omid Madani, and Rosie Jones. 2006. Active Learning with Feedback on Features and Instances. *Journal of Machine Learning Research* 7 (2006), 1655–1686.
Alan Ritter, Evan Wright, William Casey, and Tom M. Mitchell. 2015. Weakly Supervised Extraction of Computer Security Events from Twitter. In *WWW*. 896–905.
Rodrygo L.T. Santos, Craig Macdonald, and Iadh Ounis. 2011. Intent-aware Search Result Diversification. In *SIGIR*. 595–604.
Yangqiu Song and Dan Roth. 2014. On Dataless Hierarchical Text Classification. In *AAAI*. 1579–1585.
T.P. Straatsma, H.J.C. Berendsen, and A.J. Stam. 1986. Estimation of statistical errors in molecular simulation calculations. *Molecular Physics* 57, 1 (1986), 89–95.
Griffiths Thomas and Steyvers Mark. 2004. Finding scientific topics. In *PNAS*.
Hanna M. Wallach, David M. Mimno, and Andrew McCallum. 2009. Rethinking LDA: Why Priors Matter. In *NIPS*. 1973–1981.
Pengtao Xie and Eric P. Xing. 2013. Integrating Document Clustering and Topic Modeling. In *UAI*.
Limin Yao, David M. Mimno, and Andrew McCallum. 2009. Efficient methods for topic model inference on streaming document collections. In *KDD*. 937–946.
Zheng Ye and Jimmy Xiangji Huang. 2014. A simple term frequency transformation model for effective pseudo relevance feedback. In *SIGIR*. 323–332.
Jinhui Yuan, Fei Gao, Qirong Ho, Wei Dai, Jinliang Wei, Xun Zheng, Eric Po Xing, Tie-Yan Liu,

and Wei-Ying Ma. 2015. LightLDA: Big Topic Models on Modest Computer Clusters. In *WWW*. 1351–1361.

Jun Zhu, Amr Ahmed, and Eric P. Xing. 2009. MedLDA: maximum margin supervised topic models for regression and classification. In *ICML*. 1257–1264.

Jun Zhu, Amr Ahmed, and Eric P. Xing. 2012. MedLDA: maximum margin supervised topic models. *Journal of Machine Learning Research* 13 (2012), 2237–2278.