A Distribution Separation Method Using Irrelevance Feedback Data for Information Retrieval

PENG ZHANG, Tianjin University
QIAN YU, Tianjin University
YUEXIAN HOU, Tianjin University
DAWEI SONG, Tianjin University and The Open University
JINGFEI LI, Tianjin University
BIN HU, Lanzhou University

In many research and application areas, such as information retrieval and machine learning, we often encounter dealing with a probability distribution which is mixed by one distribution that is relevant to our task in hand and the other that is irrelevant and we want to get rid of. Thus, it is an essential problem to separate the irrelevant distribution from the mixture distribution. This paper is focused on the application in Information Retrieval, where relevance feedback is a widely used technique to build a refined query model based on a set of feedback documents. However, in practice, the relevance feedback set, even provided by users explicitly or implicitly, is often a mixture of relevant and irrelevant documents. Consequently, the resultant query model (typically a term distribution) is often a mixture rather than a true relevance term distribution, leading to a negative impact on the retrieval performance. To tackle this problem, we recently proposed a Distribution Separation Method (DSM), which aims to approximate the true relevance distribution by separating a seed irrelevance distribution from the mixture one. While it achieved a promising performance in an empirical evaluation with simulated explicit irrelevance feedback data, it has not been deployed in the scenario where one should automatically obtain the irrelevance feedback data. In this article, we propose a substantial extension of the basic DSM from two perspectives: developing a further regularization framework and deploying DSM in the automatic irrelevance feedback scenario. Specifically, in order to avoid the output distribution of DSM drifting away from the true relevance distribution when the quality of seed irrelevant distribution (as the input to DSM) is not guaranteed, we propose a DSM regularization framework to constrain the estimation for the relevance distribution. This regularization framework includes three algorithms, each corresponding to a regularization strategy incorporated in the objective function of DSM. In addition, we exploit DSM in automatic (i.e., pseudo) irrelevance feedback, by automatically detecting the seed irrelevant documents via three different document re-ranking methods. We have carried out extensive experiments based on various TREC data sets, in order to systematically evaluate the proposed methods. The experimental results demonstrate the effectiveness of our proposed approaches in comparison with various strong baselines.

This work is supported in part by the Chinese National Program on Key Basic Research Project (973 Program, grant No. 2013CB329304, 2014CB744604), National High-tech R&D Program of China (863 program, grant No. 2015AA015403), and the Natural Science Foundation of China (grant No. 61402324, 61272265, 61373122).

Author's addresses: P. Zhang, Q. Yu, Y. Hou, D. Song, J. Li, Tianjin Key Laboratory of Cognitive Computing and Application, School of Computer Science and Technology, Tianjin University, China, Email: {darcyzzj, yqcloud, krete1941, dawei.song2010}@gmail.com, jingfeili@tju.edu.cn; D. Song is also with Computing and Communications Department, The Open University, United Kingdom; B. Hu, Ubiquitous Awareness and Intelligent Solutions Lab, Lanzhou University, China, Email: bh@lzu.edu.cn. Corresponding authors: Yuexian Hou and Dawei Song.

DOI: http://dx.doi.org/10.1145/0000000.0000000

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms: Theory, Algorithms, Experimentation

Additional Key Words and Phrases: Relevance feedback, mixture distribution, distribution separation method, irrelevant term distribution, regularization

1. INTRODUCTION

In many fields, e.g., information retrieval, natural language processing and machine learning, we often need to deal with a mixture distribution consisting of two distributions. One is a distribution that is relevant to our task in hand, while the other is irrelevant and we want to remove it from the mixture one. For instance, if a term distribution is generated from texts, this distribution may have both relevant terms as useful information and irrelevant terms as noises, with each kind of terms having its own distribution. It is, therefore, an essential research problem to separate the irrelevant distribution from the mixture one and obtain the truly relevant distribution. In this article, we are focused on the investigation of the distribution separation process in the field of Information Retrieval (IR), particularly the relevance feedback area [Rijsbergen 1979; Lavrenko and Croft 2001].

Relevance feedback is a typical post-query process, from which we can build a refined query model (often a term distribution) in order to better represent users' underlying information need [Rijsbergen 1979; Lavrenko and Croft 2001]. The process of relevance feedback can be explicit, implicit or pseudo, each of which has its own advantages and disadvantages. Explicit relevance feedback, where users explicitly indicate which documents are relevant, has been shown useful to improve the IR performance [Buckley and Salton 1995]. However, users may be constrained and reluctant to provide explicit relevance feedback [Dumais et al. 2003; Jansen et al. 2000]. Due to users' reluctance and carelessness, the explicit feedback documents, are not always truly relevant and may contain irrelevant terms. Implicit relevance feedback aims to infer the user's preferences based on his/her interactions with the system, such as the user's click-through record [White et al. 2005; Chapelle and Zhang 2009; Liu et al. 2010]. However, it is still under exploration on what interactions should be taken into account and how good they are as relevance factors [Melucci 2012]. Pseudo relevance feedback (PRF) simply assumes a number of top ranked documents returned by an IR system as relevant. It is simple, fully automatic, and in general can improve the IR effectiveness, but may suffer from problems caused by the irrelevant documents in the pseudo relevant document set. For the above three scenarios, we can observe that users may be reluctant or careless to provide explicit relevance feedback, while implicit and automatic relevance feedback are not reliable enough. As a result, in all the three kinds of relevance feedback, the feedback document set may be mixed by relevant and irrelevant documents. Based on feedback documents, the refined query model is often a mixture of the relevance and irrelevance models, which damages the retrieval effectiveness. Thus, how to approximate the true relevance distribution from the mixture model is an important research problem.

Our approach will conquer the problem of relevance modeling, by making use of its counterpart, i.e., irrelevant information. The motivation is that, while the truly relevant information is difficult to obtain, we may relatively easily get some irrelevance feedback information, i.e., a small number of irrelevant documents. For example, when the relevant documents are sparse, the cognitive overhead for a user to find the truly relevant ones would be high. The user may have to click as many as possible document links, even though some of them may not look relevant enough. Some documents clicked by the user may not be relevant at all, while the documents ignored, or clicked but skipped quickly by the user, may be regarded as irrelevant [Bennett et al. 2012; Fox et al. 2005; Gao et al. 2009]. We can draw an analogy in the language of machine learning: if two classes are unbalanced (e.g., one class has much fewer samples than another), standard supervised or unsupervised learners tend to be overwhelmed by the larger class and ignore the smaller one [Chawla et al. 2004]. Similarly, in IR, given a query, the number of relevant documents is often much smaller than the number of irrelevant ones in the document collection. Therefore, from both cognitive and learning perspectives, it would seem more difficult to identify the relevant feedback documents, than to find out the irrelevant ones.

Recent research in negative relevance feedback has already attempted to make use of irrelevant documents to improve the retrieval performance [Dunlop 1997; Singhal et al. 1997; Wang et al. 2008]. By assuming that a set of *seed* irrelevant documents is available, a Distribution Separation Method (DSM) has been proposed in our earlier work [Zhang et al. 2009]. Essentially, given a mixture distribution and a seed irrelevance distribution, DSM aims to derive an approximation of the true relevance distribution, in other words, to separate the irrelevance distribution from the mixture one. It has been shown in [Zhang et al. 2009] that, compared with directly removing irrelevant documents, separating the irrelevance term distribution (derived from the assumingly available irrelevant documents) from the mixture term distribution (derived from the whole feedback document set including both relevant and irrelevant feedback documents) is theoretically more general and practically has led to a better performance.

Nevertheless, the basic DSM has a number of notable limitations. In this article, we aim to propose a substantial extension to the basic DSM from two perspectives, i.e., the algorithm design and implementation scenario.

On the Regularization Algorithms The original DSM's effectiveness relies on the quality of the seed irrelevance distribution. An important limitation is that: when the seed irrelevant information is limited or "impure" (i.e., containing relevant documents/terms), the output distribution of DSM may not be a good estimate of true relevance term distribution. Especially, if there is no irrelevance documents/distribution explicitly available, one should automatically derive a seed irrelevance distribution, which is inevitably mixed with some relevance information. Separating such an impure irrelevance distribution may make DSM's output term distribution drift away from the true relevance distribution.

To overcome the above limitation, in this article, we propose a regularization framework, which includes three specific algorithms, each with a different regularization term in the objective function. The regularization term acts as a penalty term to constrain the output distribution of DSM. Based on the regularization term, such a framework is expected to avoid the above-described relevance drifting problem and make DSM's estimated relevance distribution approach the true relevance distribution as closely as possible.

Specifically, the first regularization algorithm utilizes the idea of sparse representation, which is a standard way for regularization methods used in machine learning. Intuitively, it pushes more term probabilities to be zero and make the probabilities of relevant terms be more dominant. The second algorithm involves a regularization term that enforces the estimated relevance distribution to be close to a reference relevance distribution. Such a reference distribution reflects certain relevance evidences in a different way (e.g., by using contextual information). Therefore, if the output distribution of DSM is closer to the reference distribution, it is more likely that DSM can estimate the relevance distribution accurately. The third algorithm not only uses the reference distribution, but also adopts the mixture distribution to control the possible estimation of the relevance distribution. This is to avoid the unexpected situation that the output distribution of DSM has too few relevance information, even less than the original mixture distribution.

On Deployment in the Automatic Irrelevance Feedback Scenario In the implementation of the original DSM, the seed irrelevant distribution was derived from irrelevant documents, which are manually selected based on ground-truth relevance judgements (available in TREC collections used for our experiments). This is actually a simulated explicit “irrelevance” feedback scenario. However, it limits the model’s practicality when deployed in broader IR scenarios.

Thus, in this article, we will investigate how the seed irrelevant documents can be detected by automatic methods. To this end, we implement DSM in the automatic irrelevance feedback, by automatically detecting the irrelevant documents from the pseudo-relevance feedback document list, via three document re-ranking methods. These re-ranking methods are designed in order to rank the irrelevant documents in the low positions of the document list. After the re-ranking, it is reasonable to assume that, the lowly ranked documents in the re-ranked list are more likely to be irrelevant, compared with those in the original rank list.

We also propose a metric, called as Penalized Weighted Precision of Irrelevance (PWPI), to measure the quality of seed irrelevant documents (either explicitly selected or automatically detected). Based on a number of TREC data sets, we have carried out extensive experiments in the scenario of explicit and automatic ‘irrelevance” feedback. Experimental results have shown the effectiveness of our approaches.

Although this paper focuses on the application of DSM in the irrelevance feedback task, the basic framework of DSM can be generalized and can also benefit other research areas where the irrelevance/noise distribution needs to be removed from a mixture distribution. For example, a modified DSM may be useful for query reformulation [Diaz 2016] and filtering tweets for online reputation monitoring [Spina et al. 2013].

2. RELATED WORK

A classical relevance feedback approach is the Rocchio’s method [Rocchio 1971], which aims to boost terms from relevant documents and suppress terms from irrelevant documents. Recently, a widely used relevance feedback method is Relevance Model (RM) [Lavrenko and Croft 2001]. RM and its variants utilize top ranked documents D to construct a relevance term distribution R, from which the terms with highest probabilities can be used as the expanded query model. One limitation of RM-based methods is that the feedback document set D is often a mixture of relevant and irrelevant documents, so that R is likely a mixture distribution rather than true relevance distribution that is supposed to be derivable from the truly relevant documents only.

Recently, negative relevance feedback techniques have been studied to deal with difficult queries. For difficult queries, there are relatively few relevant documents in the top-ranked document list. In [Wang et al. 2008], it is assumed that all the top-ranked l documents are irrelevant, from which a negative language model is derived. After that, the ranking score (such as KL-divergence value) of each document will be revised based on the similarity between the document language model and the negative language model. The ranking scores of those documents close to the negative language model will be penalized. Rocchio’s model can also perform negative feedback by ignoring the component $w.r.t.$ the relevant documents [Zhang et al. 2009]. In [Karimzadehgan and Zhai 2011], a generalized negative language model is developed to penalize/prune the non-relevant documents that are close to the negative language model.

In [Zhang et al. 2009], we proposed a Distribution Separation Method (DSM), which is distinct from other approaches in the following aspects. First, DSM develops an idea of distribution separation by utilizing a small amount of seed irrelevant data, in order to approximate the true relevance distribution. Second, it automatically estimate

Table I. Notations

Notation	Description
M	Mixture term distribution
R	Relevance term distribution
I	Irrelevance term distribution.
I_S	Seed Irrelevance distribution
$I_{\overline{S}}$	Unknown Irrelevance distribution
$F(i)$	Probability of the i^{th} term in any distribution F
$l(F, G)$	Linear combination of distributions F and G

the linear combination coefficient of the desired relevance distribution in the mixture distribution.

Another well-regarded feedback method is the Mixture Model proposed in [Zhai and Lafferty 2001a]. It was further developed in [Zhai et al. 2004; Tao and Zhai 2006], and recently had a fast solution [Zhang and Xu 2008]. In [Zhang et al. 2016], we for the first time analytically show that Mixture Model can be regarded as a distribution separation process as in DSM, and Mixture Model's EM algorithm can be simplified by the linear separation algorithm of DSM. We also generalize DSM's theoretical analysis by providing additional KL-divergence analysis in [Zhang et al. 2016].

In this article, we present a substantial extension to the original DSM, in order to make it more general and more applicable. First, in Section 3, we describe the original DSM with a reorganized presentation (including the structure, descriptions, examples and notations, etc.). Second, in Section 4, we propose a regularization framework to regularize DSM's estimated relevance distribution. This framework includes three algorithms, each with a different regularization term in the objective function. Third, we describe how to obtain the seed irrelevance distribution in two irrelevance feedback scenarios (i.e., explicit and automatic) in Section 5. Particularly, in Section 5.3.1, we develop three document re-ranking based methods to automatically obtain the irrelevant documents from the set of feedback documents. Finally, we carry out a systematic evaluation to demonstrate the effectiveness of the basic DSM and the regularized DSM, in the scenarios of both explicit and automatic irrelevance feedback.

3. THE ORIGINAL DISTRIBUTION SEPARATION METHOD (DSM)

This section describes the original DSM proposed in [Zhang et al. 2009]. We first describe the notations and the task. Let M represent the mixture term distribution derived from all the feedback documents. M is a mixture of relevance term distribution R and irrelevance term distribution I. We assume that only part of the irrelevance distribution I_S (also called as *seed* irrelevance distribution) is available, while the other part of irrelevance distribution is unknown (denoted as $I_{\overline{S}}$). Specific formulations of distributions M, R, I and I_S based on RM will be shown in Section 5.

The task of DSM can be defined as follows: given the mixture distribution M and a seed irrelevance distribution I_S, to derive an output distribution that can approximate R as closely as possible. Specifically, as shown in Figure 1, the task of DSM can be divided into two problems: (1) How to separate I_S from M, and then derive $l(R, I_{\overline{S}})$, which is less noisy but is still a mixture of the true relevance distribution (R) and the unknown irrelevance distribution ($I_{\overline{S}}$)? (2) How to further refine the derived distribution $l(R, I_{\overline{S}})$ to approximate R as closely as possible?

To solve the above two problems, we assume that a term distribution derived from a feedback document set D is a linear combination of two term distributions, which are derived from two document subsets that form a partition of D. This is a linear combination condition of DSM. Under such a condition, the mixture distribution M derived from all the feedback documents can be a linear combination of relevance distribution R (derived from relevant documents) and irrelevance distribution I (derived from ir-

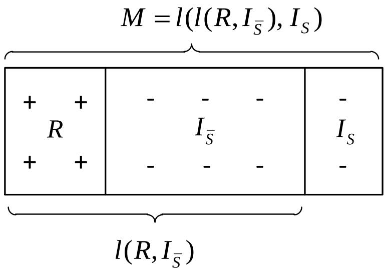

Fig. 1. An illustration of the linear combination $l(\cdot,\cdot)$ between two term distributions.

Table II. Examples of Estimation of λ and $l(R, I_{\overline{S}})$

M	$[0.16, 0.12, 0.18, 0.22, 0.06, 0.26]^T$
I_S	$[0.20, 0.00, 0.10, 0.30, 0.10, 0.30]^T$

$\hat{\lambda}$	$\hat{l}(R, I_{\overline{S}})$	$\rho(\hat{l}(R, I_{\overline{S}}), I_S)$
1	$[0.16, 0.12, 0.18, 0.22, 0.06, 0.26]^T$	0.7734
0.8	$[0.15, 0.15, 0.20, 0.20, 0.05, 0.25]^T$	0.5641
0.6	$[0.13, 0.20, 0.23, 0.17, 0.03, 0.23]^T$	0.1448
0.4	$[0.10, 0.30, 0.30, 0.10, 0.00, 0.20]^T$	-0.3636

relevant documents) (see also Section 5 for details). As shown in Figure 1, M can also be a linear combination of two distributions I_S and $l(R, I_{\overline{S}})$, where $l(R, I_{\overline{S}})$ is also a linear combination of R and $I_{\overline{S}}$. Bear in mind that both R and I are unknown for DSM. Therefore, determining the linear combination coefficient is key for solving the above problems, which will be detailed next.

3.1. Deriving $l(R, I_{\overline{S}})$ by Analyzing the Lower Bound of Linear Combination Coefficient

Recall that M is a nested linear combination $l(l(R, I_{\overline{S}}), I_S)$, which can be represented as:

$$M = \lambda \times l(R, I_{\overline{S}}) + (1 - \lambda) \times I_S \tag{1}$$

where λ $(0 < \lambda \leq 1)$ is the real linear coefficient.

The problem of estimating $l(R, I_{\overline{S}})$ does not have a unique solution generally. This is due to the fact that the value of the coefficient λ is also unknown. Therefore, the key is to estimate λ. Let $\hat{\lambda}(0 < \hat{\lambda} \leq 1)$ denote an estimate of λ, and correspondingly let $\hat{l}(R, I_{\overline{S}})$ be the estimation of the desired distribution $l(R, I_{\overline{S}})$. According to Eq. 1, we have

$$\hat{l}(R, I_{\overline{S}}) = \frac{1}{\hat{\lambda}} \times M + (1 - \frac{1}{\hat{\lambda}}) \times I_S. \tag{2}$$

There can be infinite possible choices of $\hat{\lambda}$ and its corresponding $\hat{l}(R, I_{\overline{S}})$. Table II shows some examples of different estimated values of $\hat{\lambda}$ and $\hat{l}(R, I_{\overline{S}})$, given M and I_S. To obtain a $\hat{\lambda}$ which can estimate the real coefficient λ as closely as possible, we introduce a constraint

$$\hat{l}(R, I_{\overline{S}}) \succeq 0, \tag{3}$$

which means that all the values in the distribution $\hat{l}(R, I_{\overline{S}})$ are not less than 0. Based on Eq. 2 and Eq. 3, we have

$$\hat{\lambda} \times \mathbf{1} \succeq (\mathbf{1} - M./I_S) \tag{4}$$

Table III. Simplified Notations

Original	**Simplified**	**Linear Coefficient**
$l(R, I_{\overline{S}})(i)$	$l(i)$	λ
$\hat{l}(R, I_{\overline{S}})(i)$	$\hat{l}(i)$	$\hat{\lambda}$ (estimate of λ)
$l_L(R, I_{\overline{S}})(i)$	$l_L(i)$	λ_L (lower bound of $\hat{\lambda}$)

where $\mathbf{1}$ stands for a vector in which all the entries are 1, and ./ denotes the entry-by-entry division of M by I_S. Note that if there is zero value in I_S, then $\hat{\lambda} > 1 - \infty$. It is still valid since $\hat{\lambda} > 0$. Effectively, Eq. 4 sets a lower bound of $\hat{\lambda}$:

$$\lambda_L = \max\left(\mathbf{1} - M./I_S\right) \tag{5}$$

where $\max(\cdot)$ denotes the max value in the resultant vector $\mathbf{1} - M./I_S$. Let us look at the example in Table II again. Based on Eq. 5, λ_L is 0.4. The lower bound λ_L itself also determines an estimate of $l(R, I_{\overline{S}})$, denoted as $l_L(R, I_{\overline{S}})$.

The lower bound λ_L is essential to the estimation of λ. Now, we present an important property of λ_L in Lemma 3.1. For simplicity, we use some simplified notations listed in Table III. Lemma 3.1 guarantees that if there exists zero value (e.g., for a term i, $l(i) = 0$) in $l(R, I_{\overline{S}})$, then $\lambda = \lambda_L$. In relevance feedback, if there is no distribution smoothing step involved for feedback model, zero values often exist in $l(R; I_S)$.

LEMMA 3.1. *If there exists a zero value in $l(R, I_{\overline{S}})$, then $\lambda = \lambda_L$, leading to $l(R, I_{\overline{S}}) = l_L(R, I_{\overline{S}})$.*

The proof can be found in [Zhang et al. 2009]. Intuitively, if $l(i) = 0$, we can get $\lambda = 1 - M(i)/I_S(i)$ based on Eq. 1. This λ is actually the lower bound λ_L in Eq. 5. If not, increasing or reducing λ will force $l(i)$ to be a positive or negative value, respectively. This contradicts the precondition $l(i) = 0$.

Let us see the example in Table II. Let $l(R, I_{\overline{S}}) = [0.1, 0.3, 0.3, 0.1, 0, 0.2]^T$, where a zero value exists. Given the known M and I_S in Table II, $\lambda = 0.4$, based on Eq. 1. Now, according to Eq. 5, we can get $\lambda_L = 0.4$, which equals to λ. Based on this coefficient and Eq. 2, we can then have $l_L(R, I_{\overline{S}})$ which is actually $l(R, I_{\overline{S}})$.

Remark 3.2. If there is no zero value, but there exist a few very small values in $l(R, I_{\overline{S}})$, i.e., $0 < l(i) \leq \delta$, where δ is a very small value, then $l_L(R, I_{\overline{S}})$ will be approximately equal to $l(R, I_{\overline{S}})$.

The detailed description of this remark can be found in [Zhang et al. 2009].

3.2. Minimum Correlation Analysis

Now, we go in-depth to see another property of the combination coefficient and its lower bound. Specifically, we analyze the correlation between $\hat{l}(R, I_{\overline{S}})$ and I_S, along with the decreasing coefficient $\hat{\lambda}$. Pearson product-moment correlation coefficient [Rodgers and Nicewander 1988], denoted as ρ $(-1 \leq \rho \leq 1)$, is used as the correlation measurement.

PROPOSITION 1. *If $\hat{\lambda}$ ($\hat{\lambda} > 0$) decreases, the correlation coefficient between $\hat{l}(R, I_{\overline{S}})$ and I_S, i.e., $\rho(\hat{l}(R, I_{\overline{S}}), I_S)$, will decrease.*

The proof of Proposition 1 can be found in [Zhang et al. 2009]. According to Proposition 1, among all $\hat{\lambda} \in [\lambda_L, 1]$, λ_L corresponds to $\min(\rho)$, i.e., the minimum correlation coefficient between $\hat{l}(R, I_{\overline{S}})$ and I_S. This can be illustrated by the example in Table II.

We can also change the *minimum correlation coefficient* (i.e., $\min(\rho)$) to *minimum squared correlation coefficient* (i.e., $\min(\rho^2)$). This idea can be formulated as the following optimization problem:

$$\begin{aligned} &\min_{\hat{\lambda}} [\rho(\hat{l}(R, I_{\overline{S}}),\ I_S)]^2 \\ &s.t. \quad \lambda_L \leq \hat{\lambda} \leq 1 \end{aligned} \tag{6}$$

To solve this optimization problem, we need to first find a $\hat{\lambda}$ such that the corresponding $\rho(\hat{l}(R, I_{\overline{S}}),\ I_S) = 0$. According to the proof of Proposition 1 in [Zhang et al. 2009], this $\hat{\lambda} = -\frac{a}{b}$, where $a = \sum_i^m (I_S(i) - \frac{1}{m})(M(i) - I_S(i))$, $b = \sum_i^m (I_S(i) - \frac{1}{m})^2$, and m is the number of terms. Then, we need to check whether $\lambda_L \leq -\frac{a}{b} \leq 1$ holds. If it holds, the optimal linear coefficient $\hat{\lambda}$ for the optimization problem in Eq. 6 is $-\frac{a}{b}$. Otherwise, we just compare the values of $[\rho(\hat{l}(R, I_{\overline{S}}),\ I_S)]^2$ $w.r.t.$ $\hat{\lambda} = 1$ and $\hat{\lambda} = \lambda_L$, in order to get the optimal $\hat{\lambda}$ of the objective function in Eq. 6.

Practically, in our experiments, we observed that the negative correlation value seldom exists. Then, the solutions of $\min(\rho)$ and $\min(\rho^2)$ usually equal to the lower bound λ_L. As discussed in Section 3.1, λ_L's corresponding distribution $l_L(R, I_{\overline{S}})$ can equal to $l(R, I_{\overline{S}})$, which is still a mixture distribution.

3.3. Distribution Refinement in the Original DSM

So far, we obtain an estimation of $l(R, I_{\overline{S}})$, which is still a mixture distribution of R and $I_{\overline{S}}$ (see Figure 1). In order to further compute a distribution that can get closer to R, we need to investigate the difference between $l(R, I_{\overline{S}})$ and R. As described in [Zhang et al. 2009], we observe a difference in terms of the combination coefficient. Specifically, the linear coefficient of R $w.r.t.$ M is less than that of $l(R, I_{\overline{S}})$ $w.r.t.$ M (see a detailed example in [Zhang et al. 2009]).

Recall that the lower bound of the coefficient $\hat{\lambda}$ is λ_L (see Eq. 5), and the corresponding distribution of λ_L is $l_L(R, I_{\overline{S}})$), which has been shown to be equivalent to or approximately equivalent to $l(R, I_{\overline{S}})$) (see Lemma 3.1 and Remark 3.2). This means that given the input distributions M and I_S, we can only obtain $l(R, I_{\overline{S}})$. Therefore, one needs to refine/revise the distributions M and I_S, in order to reduce the lower bound and further compute a distribution that can bridge the gap between $l(R, I_{\overline{S}})$ and R. To this end, in [Zhang et al. 2009], we proposed a strategy as follows: If any term i meets the following condition:

$$\frac{M(i)}{I_S(i)} < 1 - \lambda_L \times \eta \tag{7}$$

then, the term i will be deleted from both M and I_S, where $\eta < 1$ is a parameter in our model to control the refinement step. This refinement step can be explained in an intuitive way as follows. Specifically, if a term i satisfies Eq. 7, $M(i)$ is very small while $I_S(i)$ is relatively large, then we can consider this term as irrelevant. According to Eq. 5 and 7, after the refinement, the new lower bound is less than the current λ_L (see [Zhang et al. 2009] for detailed descriptions) .

The reduction of the lower bound value is because of the change of the two input distributions. It does not mean that the previous theoretical analysis of the lower bound is invalid. After the refinement step, one can normalize the refined input distributions, which then meet the probability distribution requirement of the previous analysis. The refinement step can be regarded as a pre-processing step.

4. A DSM REGULARIZATION FRAMEWORK

As aforementioned, the distribution refinement step can be regarded as a preprocessing step of the input distributions, but has not further advanced the objective function of DSM's optimization method in Eq. 6. On the other hand, when no explicit irrelevance feedback data is available, one needs to automatically identify some irrelevant documents/distribution, which inevitably involve relevant documents/terms, resulting in an impure irrelevance distribution. In this case, the refinement step, which is dependent on the quality of the seed irrelevance distribution, may not work as well as in the explicit feedback scenario. In other words, separating such an "impure" irrelevance distribution may make DSM's output term distribution drift further away from the true relevance distribution.

In order to solve the relevance drifting problem, we propose to advance the DSM's objective function by incorporating a regularization term into it. Such a regularization term will act as a penalty term to constrain DSM's output distribution, to make this distribution approach the true relevance distribution as closely as possible. In this article, we will introduce three different kinds of regularization term, each corresponding to a specific regularization algorithm.

The different regularization algorithms have different motivations to build the regularization terms (see Section 4.1-4.3 for details). The first algorithm is based on a standard regularization strategy, i.e., the sparse representation for the relevance distribution, which pushes more term probabilities to be zero and make the probabilities of relevant terms be more dominant. The second algorithm assumes that the estimated relevance distribution by DSM should be close to a reference distribution which is a reflection of some relevance evidence. The third algorithm is to make use of the mixture distribution. This is to avoid the unexpected situation that the output distribution of DSM has two few relevance information, even less than the original mixture distribution. Such a case can happen when the input seed irrelevant distribution is also a mixture that contains much relevant information, so that separating the seed irrelevance information can yield a bad estimation for the relevance distribution.

The above three regularization algorithms can form a regularization framework. Now, we first introduce the general formalism of such a regularization framework. As aforementioned, such a framework is based on the previous objective function in Eq. 6, and has a regularization term on DSM's estimated relevance distribution. The target of the regularization is to estimate or approximate the relevance distribution R, rather than $l(R, I_{\overline{S}})$ (see Eq. 1). Therefore, we denote an estimated distribution of DSM as $\widehat{R}$, with its optimal estimation R^* be the output distribution of the regularized DSM.

This framework can then be generally formulated as:

$$\begin{aligned}(R^*, \lambda^*) = \arg\min_{\widehat{R},\hat{\lambda}} & \left\{[\rho(\widehat{R},\ I_S)]^2 + \kappa \times Regularization(\hat{R})\right\} \\ s.t. \quad & \lambda_L \le \hat{\lambda} \le 1\end{aligned} \tag{8}$$

where we replace $\hat{l}(R, I_{\overline{S}})$ with $\widehat{R}$ (a possible estimation for the relevance distribution) in the objective function of Eq. 6, and add a regularization term $Regularization(\widehat{R})$ (as the penalty term) to advance DSM's objective function. The output distribution R^* of the regularized DSM, is the optimal $\widehat{R}$. In the following, we are going to describe each specific regularization term $Regularization(\widehat{R})$ one by one in each algorithm.

4.1. Achieving Sparseness via Formal Sparse-based Regularization

The distribution refinement step can make the estimated relevance distribution of DSM become more sparse, since this step removes some terms and makes some term probabilities approaching 0 (see Lemma 3 in [Zhang et al. 2009]). Here, we can have an alternative approach to such a sparse representation by introducing a regularization term directly in DSM's objective function. Specifically, by considering a sparse regularization term $\|\widehat{R}\|_2^2$ on $\widehat{R}$, we can formulate the following optimization problem:

$$(R^*, \lambda^*) = \arg\min_{\widehat{R},\hat{\lambda}} \left\{ [\rho(\widehat{R},\ I_S)]^2 + \kappa\|\widehat{R}\|_2^2 \right\} \\ s.t. \quad \lambda_L \leq \hat{\lambda} \leq 1 \tag{9}$$

where $\widehat{R}$ can be obtained via Eq. 2, i.e.,

$$\widehat{R} = \frac{1}{\hat{\lambda}} \times M + (1 - \frac{1}{\hat{\lambda}}) \times I_S. \tag{10}$$

To simplify the following derivation, we let $\xi = 1/\hat{\lambda}$. Then, we get $\widehat{R} = \xi \times M + (1-\xi) \times I_S$. The regularization term $\|\widehat{R}\|_2^2$ is a typical regularization method to make the output distribution $\widehat{R}$ become sparse, and the coefficient κ is a typical parameter to control the degree of sparseness.

To solve this optimization problem, we let $a = \sum_i^m (I_S(i) - \frac{1}{m})(M(i) - I_S(i)) = \sum_i^m (I_S(i))(M(i) - I_S(i))$, $b = \sum_i^m (I_S(i) - \frac{1}{m})^2$, and $c = \sum_i^m (M(i) - I_S(i))^2$, where m is the number of involved terms. Let $f(\xi)$ be the objective function $[\rho(\widehat{R},\ I_S)]^2 + \kappa\|\widehat{R}\|_2^2$. After a series of mathematical derivations [1], the solution of $f'(\xi) = 0$ leads to the real value root of the following 5-order polynomial function:

$$g(\xi) = p_5\xi^5 + p_4\xi^4 + p_3\xi^3 + p_2\xi^2 + p_1\xi^1 + p_0 \tag{11}$$

where $p_5 = bc^3\kappa$, $p_4 = 5abc^2\kappa$, $p_3 = 8a^2bc\kappa + 2b^2c^2\kappa$, $p_2 = a^3 - abc + 4a^3b\kappa + 6ab^2c\kappa$, $p_1 = a^2b - b^2c + 4a^2b^2\kappa + b^3c\kappa$, and $p_0 = ab^3\kappa$.We can compare the objective values of $f(\xi)$ between the extreme arguments (i.e., the root of $g(\xi)$ in Eq. 11) with two boundary arguments (i.e., 1 and $1/\lambda_L$) , to find the optimal ξ and $\hat{\lambda}(\hat{\lambda} = 1/\xi)$, based on which one can compute the optimal $\widehat{R}$ using Eq. 10 and obtain the output distribution R^*.

4.2. DSM Regularization with a Reference Distribution A

We can also regularize DSM by introducing a regularization term $\|\widehat{R} - A\|_2^2$. We assume an available distribution A, which reflects some kind of evidence of relevance. One possible relevance evidence, namely the contextual information of query terms in documents, will be described in Section 4.4. Here, we focus on the regularization algorithm using an available A. This idea leads to the following optimization problem:

$$(R^*, \lambda^*) = \arg\min_{\widehat{R},\hat{\lambda}} \left\{ [\rho(\widehat{R},\ I_S)]^2 + \kappa\|\widehat{R} - A\|_2^2 \right\} \\ s.t. \quad \lambda_L \leq \hat{\lambda} \leq 1 \tag{12}$$

The regularization term $\|\widehat{R} - A\|_2^2$ can measure the distance between one estimated distribution $\widehat{R}$ and the available reference distribution A. The larger $\|\widehat{R} - A\|_2^2$ will enforce more penalty on the objective function, so that this regularization term constrains the DSM's output distribution to be not far away from the reference distribution A. Note that Section 4.4 will describe how to generate of the distribution A from the the contextual information of query terms in documents.

Like we did in the previous regularization algorithm, let $\xi = 1/\hat{\lambda}$, and $f(\xi) = [\rho(\widehat{R},\ I_S)]^2 + \kappa\|\widehat{R} - A\|_2^2$. In addition to the the variables a, b, and c defined previously, we let $d = \sum_i^m (M(i) - I_S(i))A(i)$, where m is the number of involved terms.

[1] Please refer to the detailed derivations in the supplementary appendix along with this paper.

After a series of mathematical derivations, the solution of $f'(\xi) = 0$ leads to the real value root of the following 5-order polynomial function:

$$g(\xi) = p_5\xi^5 + p_4\xi^4 + p_3\xi^3 + p_2\xi^2 + p_1\xi^1 + p_0 \tag{13}$$

where $p_5 = bc^3\kappa$, $p_4 = 5abc^2\kappa - bc^2d\kappa$, $p_3 = 8a^2bc\kappa + 2b^2c^2\kappa - 4abcd\kappa$, $p_2 = a^3 - abc + 4a^3b\kappa + 6ab^2c\kappa - 4a^2bd\kappa - 2b^2cd\kappa$, $p_1 = a^2b - b^2c + 4a^2b^2\kappa + b^3c\kappa - 4ab^2d\kappa$, and $p_0 = ab^3\kappa - b^3d\kappa$. We can then compare the objective values of $f(\xi)$ between the extreme arguments (i.e., the root of $g(\xi)$ in Eq. 13) with two boundary arguments (1 and $1/\lambda_L$) , to find the optimal $\hat{\lambda}(\hat{\lambda} = 1/\xi)$ for estimating the optimal $\widehat{R}$ using Eq. 10. In this way, we get the solution for the optimization problem in Equation 12, i.e., λ^* and R^*.

4.3. DSM Regularization with Both A and M

As we explained previously, in the automatic irrelevance feedback scenario, separating an impure irrelevance distribution (i.e., the automatically detected seed irrelevance distribution is of low quality) from the mixture one will result in a less accurate estimation of the relevance distribution. An extreme case is when all the detected seed *irrelevant* documents are in fact *relevant* (rather than irrelevant as they are supposed to be). In this case, the corresponding seed irrelevance distribution will certainly have a very low quality. After separating such a low quality irrelevance distribution, the output distribution of DSM will drift far away from the relevance distribution, and may even work worse than the original mixture model. Considering that the original mixture model M usually contains some relevant documents/terms, we think that M can act as another constraint for the separating process, in addition to the external relevance distribution A proposed in the previous subsection, to prevent the estimated relevance distribution $\widehat{R}$ of DSM from drifting too far away from both M and A.

Now, we can add two regularization terms based on $\|\widehat{R} - A\|_1$ and $\|\widehat{R} - M\|_1$, respectively. [2] We can then formulate an optimization problem using l_1 norm as:

$$\begin{aligned} (R^*, \lambda^*) = &\arg\min_{\widehat{R},\hat{\lambda}} \left\{ [\rho(\widehat{R},\ I_S)]^2 + \kappa \left(\|\widehat{R} - A\|_1 + \|\widehat{R} - M\|_1 \right) \right\} \\ &s.t. \quad \lambda_L \le \hat{\lambda} \le 1 \end{aligned} \tag{14}$$

Similar to the regularization term described in Section 4.2, this regularization term $\|\widehat{R} - A\|_1 + \|\widehat{R} - M\|_1$ can also push the resultant distribution R^* to get close to both the reference distribution A and the mixture distribution M. The reason why we make use of the mixture distribution, is to avoid the situation that the output distribution has less relevance information than the original mixture distribution, in case that the seed irrelevant distribution to be separated from the mixture one contains too much relevant information.

The formal solution to the optimization problem presented in Eq. 14 is computationally intractable. To make this regularization strategy feasible, we propose an approximate solution with two steps. The first step is to solve the first term of Eq. 14, i.e., $\min_{\hat{\lambda}} [\rho(\widehat{R},\ I_S)]^2$. This is the original DSM, and we denote its output as $\widehat{R}_0$ and $\hat{\lambda}_0$ accordingly. Then, in the second step, we solve the Taxicab metric based optimization

[2] Here, we adopt the l_1 norm rather than the previously used l_2 norm, since the l_2 norm based regularization term for $\widehat{R} - A$ and $\widehat{R} - M$ will result in an approximated solution which linearly combines the mixture distribution M *back* for DSM's output distribution.

problem as follows:

$$(R^*, \lambda^*) = \arg\min_{\widehat{R},\hat{\lambda}} \left\{ \|\widehat{R} - \widehat{R}_0\|_1 + \|\widehat{R} - A\|_1 + \|\widehat{R} - M\|_1 \right\} \\ s.t. \quad \lambda_L \leq \hat{\lambda} \leq 1 \tag{15}$$

The solution to this optimization problem is:

$$R^*(i) = \begin{cases} min(M(i), A(i)) & if\ \widehat{R}_0(i) < min(M(i), A(i)) \\ max(M(i), A(i)) & if\ \widehat{R}_0(i) > max(M(i), A(i)) \\ \widehat{R}_0(i) & Otherwise \end{cases} \tag{16}$$

This regularization assures that $R^*(i)$ falls in the interval limited by $M(i)$ and $A(i)$, in order to prevent the estimated value of $R^*(i)$ from being too far away from $M(i)$ or $A(i)$. Specifically, if the output distribution $R_0(i)$ of the basic DSM being too small (see the first case in Eq. 16) or too large (see the second case in Eq. 16), we let the estimated value for $R^*(i)$ be the intermediate value between $M(i)$ and $A(i)$. Otherwise, the original $\widehat{R}_0(i)$ is acceptable as a value of $R^*(i)$.

Accordingly, the corresponding linear coefficient λ^* of R^* can be adaptively computed for each term i as:

$$\lambda^*(i) = \begin{cases} \dfrac{M(i) - I_S(i)}{A(i) - I_S(i)} & if\ \ R^*(i) = A(i) \\ 1 & if\ \ R^*(i) = M(i) \\ \hat{\lambda}_0 & if\ \ R^*(i) = \widehat{R}_0(i) \end{cases} \tag{17}$$

4.4. Incorporating Contextual Information as Evidence of Relevance

Here, we describe a method to construct the reference distribution A (used in the second and third regularization algorithms), by extracting a positional dependency-based context distribution. Specifically, we explore the positional context of query terms in the feedback documents. Motivated by existing information retrieval models utilizing positional information [Song et al. 2008; Lv and Zhai 2010; Zhao et al. 2011; He et al. 2011], our intuition is that the words appearing nearer to query terms in the documents have higher dependencies to query, thus should have higher weights in the true relevance distribution.

To obtain the positional context distribution, we take a window based approach, i.e., only the words co-occurring with a query word within a fixed length window in the feedback documents are used. Specifically, for each document d, we only choose the terms that appear at least in one window centered by a query term. With the chosen words, we reconstruct a new document $Recons(d)$:

$$Recons(d, \sigma) = \{w_j \in d | \exists k, w_j \in Window(q_k, \sigma)\} \tag{18}$$

where $Window(q_k, \sigma)$ is a term sequence within a window centered by a query term q_k, σ denotes half the window size and w_j denotes a term token for the term w.

Formally, this method can be formulated as:

$$p(w|A) = \frac{1}{Z} \sum_{d \in D} p(w|Recons(d, \sigma)) \times Weight(d) \tag{19}$$

where $Weight(d)$ is the weight of a document. In this article, the weight is the normalized query likelihood of a document (see the next section for details). We also imple-

Table IV. Distributions Obtained by RM

Conditions	**Distributions by RM**
D_F is the whole feedback document set D	Mixture term distribution M
D_F is the relevant feedback document set D_R ($D_R \subseteq D$)	Relevance term distribution R
D_F is the irrelevant feedback document set D_I ($D_I = D - D_R$)	Irrelevance term distribution I
D_F is the *seed* irrelevant feedback document set D_{I_S}	Seed Irrelevance distribution I_S

mented a kernel-based context distribution, and the prior experiments show that the window-based distribution outperforms the kernel-based one.

5. DEPLOYMENT OF DSM IN IRRELEVANCE FEEDBACK SCENARIOS

As discussed earlier, the *seed* irrelevant distribution is one input for DSM. In this section we present a series of approaches to the seed irrelevance distribution in explicit and automatic irrelevance feedback scenarios.

5.1. Explicit Irrelevance Feedback

In this paper, we will deploy DSM based on the distributions obtained by Relevance Model (RM) [Lavrenko and Croft 2001]. The feedback term distribution F obtained by RM is formulated as:

$$p(w|F) = \sum_{d \in D_F} p(w|d) \frac{p(q|d)}{Z_F} \tag{20}$$

where $p(w|d)$ is probability of term w in a document d, $p(q|d)$ is the query likelihood (QL) score of the document d, and $Z_F = \sum_{d' \in D_F} p(q|d')$ is the summed QL scores over the documents in feedback document set D_F. Since the document prior $p(d)$ is often assumed as uniform in RM, we omit $p(d)$ in the above formulation. The feedback distribution can also be smoothed with a collection term distribution C via $p(w|F) = (1 - \mu_C)p(w|F) + \mu_C p(w|C)$.

In the explicit feedback scenario, the seed irrelevant documents can be selected from a small percentage (e.g., 10%-30%) of top-ranked irrelevant documents in D. In implementation, we simulate the explicit feedback based on the ground-truth associated to the test collections [Zhang et al. 2009]. Once we have the seed irrelevant documents D_{I_S}, we can build the seed irrelevance distribution I_S by RM based on Eq.20 by setting D_F as D_{I_S}. To obtain the mixture distribution M and other distributions, one can use different feedback document set (subject to its availability) illustrated in Table IV.

We adopt a top-down manner, i.e., the highly ranked irrelevant documents (as apposed to the lowly ranked ones) are selected as seeds. This is because, the higher the documents are in the original ranking, the larger document weights (and thus a greater impact) they have in construction of the mixture distribution. In RM (Eq. 20), the normalized score $p(q|d)/Z_M$ is used as document weight when generating the mixture model. As a consequence, separating the irrelevance model derived from the highly ranked irrelevant documents (with larger weights) would have more influence on the DSM process.

5.2. Quality Measurement of Seed Irrelevant Documents

As we explained, a large document weight in the mixture distribution is a characteristic of "good" irrelevant document. If we know which documents are irrelevant, it is better to select the top-ranked irrelevant documents as the seed ones. In automatic irrelevance feedback, however, such explicit information is not available. Therefore, we need to design a quality metric for the automatically detected irrelevant documents.

In order to integrate the document weight in the quality measurement and penalize the quality when the truly relevant documents are selected, we define a metric, namely

Penalized Weighted Precision of Irrelevance (PWPI):

$$PWPI(D_{I_S}, q) = \frac{\sum_{d \in D_{I_S}} f(d,q) \times g'(d,q)}{|D_{I_S}|} \tag{21}$$

where D_{I_S} is the detected irrelevance document set; $f(d,q)$ is the document weight of d in the mixture distribution given the query q; and $g'(d,q) = -1$ if d is "relevant" to q and 1 otherwise. In RM, the document weight $f(d,q)$ is $p(q|d)/Z_F$ (see Eq. 20). It turns out that the document weight $f(d,q)$ is the normalized query likelihood $p(q|d)$.

5.3. Automatic Irrelevance Detection

To automatically detect the seed irrelevant documents, we adopt a two-step approach. First, we re-rank the feedback documents by a re-ranking method. We will investigate three re-ranking methods, each corresponding to a different ranking strategy, which will be detailed later in Section 5.3.1. Second, in the re-ranked feedback document list, since no explicit judgement data is available, we select irrelevant documents in a bottom-up manner, i.e., considering a number of the low ranked documents as the seed irrelevant documents.

We adopt this two-step method due to the following reasons. These re-ranking methods (detailed next) are designed in order to rank the irrelevant documents in the low positions of the document list. In other words, a re-ranking method, to some extent, can more reliably place an irrelevant documents at a lower position in the ranking. After the re-ranking, the lowly ranked documents in the re-ranked list are more likely to be irrelevant, compared with those in the original rank list. Therefore, in the second step, after the re-ranking, we can select the lowly-ranked documents as irrelevant documents with a higher confidence.

After we get a set of seed irrelevant documents D_{I_S} automatically, we can derive the seed irrelevance distribution I_S, by setting D_F as D_{I_S} in Eq.20. It should be noted that D_{I_S} is not necessarily a subset of D_I (which contains the truly irrelevance documents) in the automatic irrelevance feedback scenario.

5.3.1. Three Re-ranking Methods. We investigate three re-ranking methods, each corresponding to one ranking strategy or technique. We think that such a variety of re-ranking approaches can help us better understand how to automatically select irrelevant documents. The first one is based on the clustering hypothesis [Rijsbergen 1979], and it re-ranks the feedback documents according to a k-nn similarity score which considers the dependencies among the k-nearest neighbouring documents. The second re-ranking method not only considers the document dependencies, but also utilizes the original relevance scores of feedback documents. For the third method, we adopt a learning-to-rank approach which can learn a new ranking of the feedback documents via a learner trained based on several relevance features. Next, we detail these ranking methods.

Outlier-based Re-ranking Method According to the clustering hypothesis [Rijsbergen 1979][Tombros and van 2004], the topically-relevant documents tend to cluster together, while the irrelevant ones would be scattered. By considering the scattered irrelevant documents as outlier documents, we then propose to use outlier detection methods to automatically detect the irrelevant documents. In this article, motivated by the k-nn distance scores [Ramaswamy et al. 2000; Angiulli and Pizzuti 2002], we adopt the k-nn similarity scores as the indicator for the outlier documents. The smaller the k-nn similarity score is, the more likely a document is an outlier. A number of most likely outlier documents are then selected as the seed irrelevance documents. Therefore, the re-ranking is based on the k-nn similarity score, which is calculated as

$$knn\_score(d) = \sum_{d_i;d_i \in knn(d)} Cos(d, d_i)$$

This method is simple and only considers the inter-document similarities. It does not take into account the original relevance scores of feedback documents. To some extent, such simplicity could be a benefit: If a re-ranking method is highly dependent on the original relevance scores (which are proportional to the document weights described in Section 5.2), these good irrelevant documents with large document weights (ranked high in the original ranking), are less likely to be re-ranked to the low positions, where the seed irrelevant documents are selected. Indeed, this observation may not be always valid and we need to implement other re-ranking approaches that integrates the original relevance scores, for a systematic investigation.

QPRP-based Re-ranking Method For a re-ranking method closely dependent on the original relevance scores, we will utilize a Quantum Probability Ranking Principle (QPRP) approach which was recently developed for document re-ranking purpose [Zuccon and Azzopardi 2010]. Regardless of QPRP's quantum motivation, this method performs well in the re-ranking scenario and its formulation is simple and neat. It is argued that QPRP can simplify the parameter estimation in the Portfolio Theory based re-ranking method that also relies on the original relevance scores and the inter-document dependencies [Wang and Zhu 2009; Zuccon et al. 2010]. Specifically, QPRP ranks the documents in a greedy manner using the following strategy:

$$d_r = \arg\max(f(d) + \sum_{d_x \in RA} I_{d,d_x}) \tag{22}$$

where $f(d)$ is the original relevance score of the current document d and the interference term $I_{d,d_x} (I_{d,d_x} = \sqrt{f(d)}\sqrt{f(d_x)}Cos(d, d_x))$ encodes the inter-document similarity between d and previously ranked documents d_x, which is initially set to be the top-most document in the original ranking. As we can see from Eq. 22, if a document is ranked higher originally and is closely related to a higher re-ranked document, its rank order will be promoted in its re-ranked list. This strategy implies that a document with a large document weight and ranked top in the original ranking, is less likely to be in the low positions of the re-ranked list.

L2R-based Re-ranking Method In addition to the above two methods, we also adopt a Learning to Rank (L2R) based re-ranking method, which integrates many relevance features and has a principled learning algorithm. Through training of the learning model, this method is expected to maximize the effectiveness (e.g., MAP) of the feedback documents. We re-rank feedback documents with the LambdaMART model (a widely used learning to rank model [Burges 2010]). We think that by adopting this L2R-based method, we can explore the potential of a re-ranking method as one step to detect the irrelevant documents. Note that the features we take into account include several original relevance scores (e.g., query likelihood scores and BM25 scores). Please see Section 6.2.2 for some set-up details of the L2R based methods.

5.3.2. Discussion on Different Re-ranking Methods. Now we compare the above three re-ranking methods from two perspectives: algorithm design and ability to detect irrelevant documents. From the algorithm design point of view, the Outlier re-ranking method is the simplest as it just considers the document dependency for the re-ranking purpose. It does not utilize the original relevance score as used in the other two methods. The QPRP-based re-ranking method not only considers the document dependency, but also incorporates the original relevance score. The L2R-based reranking method considers various kinds of relevance scores as features and involves a training procedure. This method would be more time-consuming than the other two.

In terms of the ability to detect irrelevant documents, as we described in Section 5.2, a large document weight, which corresponds to a large original relevance score, in the mixture distribution is a characteristic of “good” irrelevant document. If a re-ranking method is dependent on the original relevance score (as in QPRP and L2R-based reranking methods), these “good” irrelevant documents with large document weights and ranked highly in the original ranking, are less likely to move down to the lower positions in the re-ranked list. Therefore, a simple outlier method, which is solely based on outlier scores, may select the seed irrelevant documents more effectively. We will continue to explore this analysis in Section 6.4.1.

6. EMPIRICAL EVALUATION

6.1. Test Collections

The evaluation involves four standard TREC collections, including WSJ (87-92, 173,252 documents), AP (88-89, 164,597 documents) in TREC Disk 1 & 2, ROBUST 2004 (528,155 documents) in TREC Disk 4 & 5, and WT10G (1,692,096 documents). These data sets involve a variety of texts, ranging from newswire articles to Web/blog data. Both WSJ and AP data sets are tested on queries 151-200, while the ROBUST 2004 and WT10G collections are tested on queries 601-700 and 501-550, respectively. The *title* field of the queries is used to reflect the typical keyword based search scenarios. The Lemur 4.7 toolkit [Ogilvie and Callan 2002] is used for indexing and retrieval. All collections are stemmed using the Porter stemmer and stop words are removed in the indexing process. [3]

6.2. Evaluation Set-up

Initially, the top-ranked n ($n = 50$) [4] feedback documents D are retrieved by the query-likelihood (QL) based language model (LM), with Dirichlet prior [Zhai and Lafferty 2001b] set to a fixed value 700. RM is then selected as the baseline for the second-round retrieval due to the facts that RM is a successful relevance feedback model and consistently outperforms the standard LM model by 10%-15% [Lavrenko and Croft 2001]. In query expansion models, top 100 terms in the corresponding distributions are selected as expanded terms. We smooth the RM based distributions with the collection term distribution with the smoothing parameter μ_C being 0.5 [Zhang et al. 2009].

As for the evaluation metric for the retrieval performance, we use the Mean Average Precision (MAP), which is the mean value of average precision over all queries and reflects the overall retrieval effectiveness. Top 1000 documents retrieved by the negative KL-divergence method [Lafferty and Zhai 2001] are used for performance evaluation. In addition, we use the Wilcoxon significance test to examine the statistical significance of the improvements over the baseline (baseline model and significant test results are shown in the result tables).

6.2.1. Set-up for Explicit Irrelevance Feedback. Based on the TREC relevance judgements, a small percentage (denoted as r_n) of irrelevant documents in D are selected as the seed irrelevant documents D_{I_S}, where r_n can be 10%, 20% and 30%. We denote $|D_I|$ as the total number of irrelevant feedback documents and choose a number (i.e.,

[3] In the supplementary appendix, we report some preliminary evaluation results on another test collection, i.e., query logs from a commercial search engine. The supplementary appendix also includes more results on different smoothing configurations, parameter sensitivity study, and quality measurement of detected irrelevant documents.

[4] We tested various other values for n (e.g., 30, 50, 70) and obtained similar results.

Table V. Features for Learning to Rank.

ID	Features and Descriptions	Category
1	$\sum_{q_i \in q \cap d} c(q_i, d)$ in the document d.	Q-D
2	$\sum_{q_i} idf(q_i)$	Q
3	$\sum_{q_i \in q \cap d} c(q_i, d) \cdot idf(q_i)$ in document d	Q-D
4	$\|d\|$ document length	D
5	BM25 score of document d corresponding to q.	Q-D
6	LMIR.ABS score of document d corresponding to q.	Q-D
7	LMIR.DIR score of document d corresponding to q.	Q-D
8	LMIR.JM score of document d corresponding to q.	Q-D

$round(|D_I| \times r_n)$) of top-ranked irrelevant documents as the seed irrelevant documents. The refinement parameter η is chosen from the interval [0.4, 1] with increment 0.1 [Zhang et al. 2009].

6.2.2. Set-up for Automatic Irrelevance Feedback. We will also test DSM without explicit irrelevance information. As for the irrelevant documents detection methods, we use the Outlier-based method (Outlier), QPRP-based method (QPRP), and L2R-based method (L2R), described in Section 5.3.1. For each method, we select a number (i.e., $round(|D| \times r_n)$) of irrelevant documents as the seed irrelevant documents, where $|D|$ is the total number of feedback documents, and r_n ($r_n = 0.1, 0.2, 0.3$) is the ratio of selected documents in D, with the retrieval results for $r_n = 0.3$ are reported. The refinement parameter η of DSM is fixed as 1, since in automatic irrelevance scenario, the quality of the seed irrelevance distribution is not good enough and the refinement step does not work very well. This also motivates and helps us to focus on the investigation of a formal regularization algorithm.

In the Outlier-based re-ranking method, the parameter k for the k-nn score is set to 5, while we have similar results when k is 3 or 7. For the QPRP based re-ranking, there are no free parameters.

For the L2R-based method, we re-rank the pseudo feedback documents for each query with the LambdaMART model. We extract a series of widely used features from corresponding document collections and test queries. They are listed in Table V. In the "Category" column, Q stands for query feature, D stands for document feature, and Q-D means a query-document pair of features. 5-folds cross validations are conducted for each dataset. RankLib[5] is utilized to run the LambdaMART algorithm, in which "-norm" is set as "zscore", and all LambdaMART-specific parameters are set as default values. In BM25 model, k_1, k_3 and b are free parameters, and we set $k_1 = 2.5$, $k_3 = 0$ and $b = 0.8$. In language models, we set parameter $\delta = 0.7$ in LMIR.ABS, $\mu = 2000$ in LMIR.DIR and $\lambda = 0.1$ in LMIR.JM [Zhai and Lafferty 2001b].

6.2.3. Set-up for Regularization Methods of DSM. For both explicit and automatic irrelevance feedback scenarios, we evaluate the regularized DSM (denoted as **DSM+**) with three different regularization algorithms. We denote the three regularization methods described in Section 4 as DSM+SparseReg (for the sparse regularization method in Section 4.1), DSM+ContextReg (for the contextual distribution A based regularization in Section 4.2), and DSM+TaxicabReg (for the Taxicab approximation for the regularization using both A and M in Section 4.3), respectively. In the explicit scenario, we set κ in SparseReg and ContextReg to 10. For the scenario where irrelevant documents are detected automatically, κ is set to 450. The evidence distribution used in regularization experiment is the window-based context distribution as described in Section 4. We fix the window radius σ to 5.

[5] https://sourceforge.net/p/lemur/wiki/RankLib/

6.2.4. Notations of Various Relevance Feedback Methods for Query Expansion. We now summarize the notations used for the methods that are to be evaluated in the experiment. **RM+** denotes RM running on the documents in $D - D_{I_S}$, which is a feedback document set with seed irrelevant documents (which forms a set D_{I_S}) directly removed from D. **DSM–** denotes the distribution separation method (DSM) without the refinement step (i.e., $\eta = 1$ in Section 3.3). **DSM** denotes DSM with refinement step. Therefore, DSM equals to DSM– when η is set to 1 in DSM. **DSM+** denotes the regularized DSM with three regularization algorithms, i.e., sparse regularization (DSM+SparseReg), contextual regularization (DSM+ContextReg), and Taxicab approximation based regularization (DSM+TaxicabReg). **DSM+q** means that after DSM's regularization, the estimated relevance distribution is then interpolated with the original query q.

In both explicit and automatic irrelevance feedback, we will evaluate the above methods *step-by-step* for a systematic evaluation. For the automatic irrelevance feedback, we compare DSM+ with two strong baselines, namely RM3 and PRM. **RM3** is a linear combination between the expanded query by RM and the original query q [Abdul-Jaleel et al. 2004]. Positional Relevance Model (**PRM** [Lv and Zhai 2010]) is a state-of-the-art that exploits contextual information.

6.3. Effectiveness of DSM and Regularized DSM in Explicit Irrelevance Feedback

Table VI. Evaluation on DSM and Regularized DSM in Explicit Irrelevance Feedback

MAP (chg% over RM)		$r_n = 0.1$	$r_n = 0.2$	$r_n = 0.3$
		WSJ8792		
RM (baseline)		0.3538	0.3538	0.3538
RM+		0.3710(+4.86%)$^{\alpha}$	0.3798(+7.35%)$^{\alpha}$	0.3872(+9.44%)$^{\alpha}$
DSM–		0.3693(+4.38%)$^{\alpha}$	0.3787(+7.04%)$^{\alpha}$	0.3776(+6.73%)$^{\alpha}$
DSM		0.3878(+9.61%)$^{\alpha\beta}$	0.4030(+13.91%)$^{\alpha\beta}$	0.4056(+14.64%)$^{\alpha\beta}$
DSM+	SparseReg	0.3902(+10.29%)$^{\alpha\beta}$	0.4045(+14.33%)$^{\alpha\beta}$	0.4108(+16.11%)$^{\alpha\beta}$
	ContextReg	0.3886(+9.84%)$^{\alpha\beta}$	0.4035(+14.05%)$^{\alpha\beta}$	0.4096(+15.77%)$^{\alpha\beta}$
	TaxicabReg	**0.4007**(+13.26%)$^{\alpha\beta}$	**0.4115**(+16.31%)$^{\alpha\beta}$	**0.4183**(+18.23%)$^{\alpha\beta}$
		AP8889		
RM (baseline)		0.3755	0.3755	0.3755
RM+		0.3939(+4.90%)$^{\alpha}$	0.4042(+7.64%)$^{\alpha}$	0.4109(+9.43%)$^{\alpha}$
DSM–		0.3898(+3.81%)$^{\alpha}$	0.4004(+6.63%)$^{\alpha}$	0.4050(+7.86%)$^{\alpha}$
DSM		0.4024(+7.16%)$^{\alpha\beta}$	0.4125(+9.85%)$^{\alpha\beta}$	0.4218(+12.33%)$^{\alpha\beta}$
DSM+	SparseReg	0.4033(+7.40%)$^{\alpha\beta}$	0.4174(+11.16%)$^{\alpha\beta}$	0.4230(+12.65%)$^{\alpha\beta}$
	ContextReg	0.4031(+7.35%)$^{\alpha\beta}$	0.4167(+10.97%)$^{\alpha\beta}$	0.4223(+12.46%)$^{\alpha\beta}$
	TaxicabReg	**0.4120**(+9.72%)$^{\alpha\beta}$	**0.4230**(+12.65%)$^{\alpha\beta}$	**0.4270**(+13.72%)$^{\alpha\beta}$
		ROBUST2004		
RM (baseline)		0.3262	0.3262	0.3262
RM+		0.3599(+10.33%)$^{\alpha}$	0.3772(+15.63%)$^{\alpha}$	0.3853(+18.12%)$^{\alpha}$
DSM–		0.3586(+9.93%)$^{\alpha}$	0.3745(+14.81%)$^{\alpha}$	0.3797(+16.40%)$^{\alpha}$
DSM		0.3813(+16.89%)$^{\alpha\beta}$	0.4033 (+23.64%)$^{\alpha\beta}$	0.4106(+25.87%)$^{\alpha\beta}$
DSM+	SparseReg	0.3820(+17.11%)$^{\alpha\beta}$	0.4027(+23.45%)$^{\alpha\beta}$	0.4106(+25.87%)$^{\alpha\beta}$
	ContextReg	0.3819(+17.08%)$^{\alpha\beta}$	0.4023(+23.33%)$^{\alpha\beta}$	0.4095(+25.54%)$^{\alpha\beta}$
	TaxicabReg	**0.3930**(+20.48%)$^{\alpha\beta}$	**0.4138**(+26.85%)$^{\alpha\beta}$	**0.4152**(+27.28%)$^{\alpha\beta}$
		WT10G		
RM (baseline)		0.2004	0.2004	0.2004
RM+		0.2439(+21.71%)$^{\alpha}$	0.2419(+20.71%)$^{\alpha}$	0.2631(+31.29%)$^{\alpha}$
DSM–		0.2447(+22.11%)$^{\alpha}$	0.2438(+21.66%)$^{\alpha}$	0.2609(+30.19%)$^{\alpha}$
DSM		0.2468(+23.15%)$^{\alpha}$	0.2594(+29.44%)$^{\alpha\beta}$	0.2724(+35.93%)$^{\alpha\beta}$
DSM+	SparseReg	0.2478(+23.65%)$^{\alpha}$	0.2572(+28.34%)$^{\alpha\beta}$	0.2697(+34.58%)$^{\alpha}$
	ContextReg	0.2471(+23.30%)$^{\alpha}$	0.2565(+27.99%)$^{\alpha\beta}$	0.2695(+34.48%)$^{\alpha}$
	TaxicabReg	**0.2581**(+28.79%)$^{\alpha\beta}$	**0.2734**(+36.43%)$^{\alpha\beta}$	**0.2949**(+47.16%)$^{\alpha\beta}$

α and β, respectively, indicate a statistically significant improvement over RM and RM+, at level 0.05

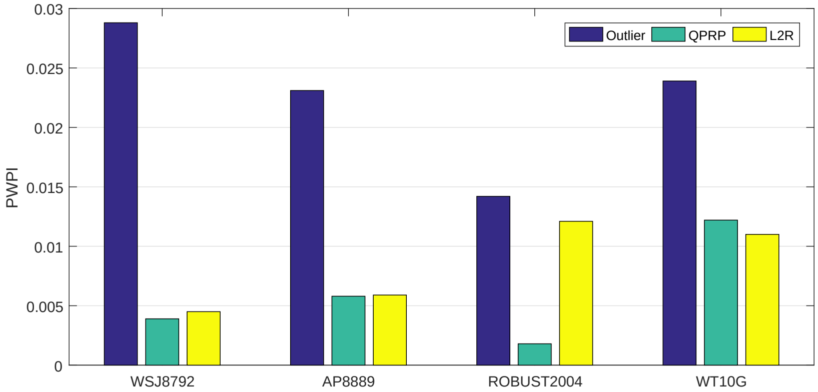

Fig. 2. Quality of Automatically Detected Irrelevant Documents

Now, we test the retrieval effectiveness of DSM when the seed irrelevant documents can be obtained from the explicit relevance feedback. The evaluation results are summarized in Table VI. Since we have reported the detailed results of (RM vs. RM+), (RM+ vs. DSM–) and (DSM– vs. DSM) in [Zhang et al. 2009], we are going to briefly summarize the results of the above pair-wise comparisons, and mainly describe the results of (DSM vs. DSM+) in explicit irrelevance feedback scenario.

The results show that RM+ (i.e., RM on $D - D_{I_S}$) can significantly improve RM on all collections. A trend we can reveal in Table VI is that the larger scale collections and the larger ratio of irrelevant documents are involved, the larger improvements can be achieved. RM+ has a similar performance with DSM–, which supports Remark 3.2 about the smoothing configuration. Next, we evaluate the DSM with the refinement step (described in Section 3.3). The results show that DSM not only significantly outperforms RM, but also outperforms RM+ and DSM–, which shows the effectiveness of the refinement step of DSM in explicit irrelevance feedback.

We now examine the effectiveness of the regularized DSM (denoted as DSM+). From Table VI we can find that the performance of DSM+SparseReg and DSM+ContextReg are close to that of DSM, while DSM+TaxicabReg achieves the best performance among the three regularization methods and can outperform DSM. Particularly, on WT10G, compared with DSM, DSM+ with Taxicab metric based regularization method (TaxicabReg) improves the performance by 4.58%, 5.40% and 8.26%, respectively when r_n is 0.1, 0.2 and 0.3. We can observe that for larger collection, the regularization step can bring more benefit. The performance difference between three regularization methods may be due to the following reasons. First, compared with the SparseReg and ContextReg, TaxicabReg utilizes more relevance evidence for the relevance model estimation. Second, in TaxicabReg, the relevance evidences (e.g., the contextual distribution A) directly constrain the relevance distribution's estimation $\hat{R}$, leading to an adaptive solution for $\hat{\lambda}$ (see Eq. 17).

6.4. Effectiveness of DSM and Regularized DSM in Automatic Irrelevance Feedback

6.4.1. Results of Quality Measurement of Detected Seed Irrelevant Documents. In automatic irrelevance feedback scenario, the quality measurement results of the detected irrelevant documents using three automatic detection methods are shown in Figure 2. The three methods are Outlier, QPRP and L2R, respectively (see Sections 6.2.3 and 5.3.1 for detailed descriptions).

Table VII. Evaluation on DSM and Regularized DSM using Automatic Detection Method

MAP (chg% over RM)		**WSJ8792**	**AP8889**	**ROBUST2004**	**WT10G**
RM (baseline)		0.3538	0.3755	0.3262	0.2004
		Irrelevant Documents Detection Method: OutlierD			
RM+		0.3635(+2.74%)$^{\alpha}$	0.3896(+3.75%)$^{\alpha}$	0.3362(3.03%)$^{\alpha}$	0.1996(-0.40%)
DSM–		0.3683(+4.10%)$^{\alpha}$	0.3904(+3.97%)$^{\alpha}$	0.3401(4.23%)$^{\alpha}$	0.2036(+1.60%)
DSM		0.3683(+4.10%)$^{\alpha}$	0.3904(+3.97%)$^{\alpha}$	0.3401(+4.26%)$^{\alpha}$	0.2036(+1.60%)
DSM+	SparseReg	0.3547(+0.25%)	0.3805(+1.33%)	0.3423(+4.94%)$^{\alpha}$	**0.2244**(+11.98%)$^{\alpha\beta}$
	ContextReg	0.3546(+0.23%)	0.3808(+1.41%)	0.3426(+5.03%)$^{\alpha}$	0.2241(+11.83%)$^{\alpha\beta}$
	TaxicabReg	**0.3819**(+7.94%)$^{\alpha\beta}$	**0.3966**(+5.62%)$^{\alpha\beta}$	**0.3560**(+9.14%)$^{\alpha\beta}$	0.2162(+7.88%)$^{\alpha\beta}$
		Irrelevant Documents Detection Method: QPRP			
RM+		0.3539(-0.02%)	0.3754(-0.02%)	0.3287(+0.74%)	0.2037(+1.65%)
DSM–		0.3540(+0.06%)	0.3815(1.60%)	0.3363(+3.10%$^{\alpha}$	0.2005(+0.05%)
DSM		0.3540(+0.06%)	0.3815(+1.60%)	0.3363(+3.10%$^{\alpha}$	0.2005(+0.05%)
DSM+	SparseReg	0.3580(+1.19%)	0.3708(-1.25%)	0.3421(+4.87%)$^{\alpha\beta}$	0.2320(+15.77%)$^{\alpha\beta}$
	ContextReg	0.3583(+1.27%)	0.3741(-0.37%)	0.3442(+5.52%)$^{\alpha\beta}$	**0.2324**(+15.97%)$^{\alpha\beta}$
	TaxicabReg	**0.3764**(+6.39%)$^{\alpha\beta}$	**0.3949**(+5.17%)$^{\alpha\beta}$	**0.3622**(+11.04%)$^{\alpha\beta}$	0.2313(+15.42%)$^{\alpha\beta}$
		Irrelevant Documents Detection Method: L2R			
RM+		0.3560(+0.62%)	0.3728(-0.72%)	0.3251(-0.37%)	0.2054(+2.50%)$^{\alpha}$
DSM–		0.3547(0.25%)	0.3743(-0.32%)	0.3305(+1.32%)	0.2031(+1.35%)
DSM		0.3547(0.25%)	0.3743(-0.32%)	0.3305(+1.32%)	0.2031(+1.35%)
DSM+	SparseReg	0.3561(+0.65%)	0.3674(-2.16%)	0.3401(+4.26%)$^{\alpha\beta}$	0.2331(+16.32%)$^{\alpha\beta}$
	ContextReg	0.3565(+0.76%)	0.3678(-2.05%)$^{\alpha}$	0.3410(+4.54%)$^{\alpha\beta}$	0.2333(+16.42%)$^{\alpha\beta}$
	TaxicabReg	**0.3762**(+6.33%)$^{\alpha\beta}$	**0.3886**(+3.49%)$^{\alpha\beta}$	**0.3508**(+7.54%)$^{\alpha\beta}$	**0.2373**(+18.41%)$^{\alpha\beta}$

α and β, respectively, indicate a statistically significant improvement over RM and RM+, at level 0.05

The quality results for Outlier are the best in terms of the evaluation based on averaged PWPI over queries. This observation is a little surprise but is generally consistent with our analysis in Section 5.3.1, where we discussed the advantages of the outlier method: it does not take into account the original relevance scores of feedback documents, as QPRP and L2R do. If a re-ranking method is highly dependent on the original relevant score, these good irrelevant documents with large document weights and ranked top in the original ranking, are less likely to be ranked in the low positions in the re-ranked list. Therefore, a simple outlier detection method whose re-ranked list is solely based on outlier scores, may select those document with large document weight (with top positions in the original rank) as seed irrelevant documents, which helps improve the values of the penalized weighted precision of irrelevance (PWPI). The PWPI-based irrelevance quality measurement results are also consistent with the retrieval performances of DSM in Table VII, which we will analyze next.

6.4.2. Results of DSM and Its Regularization using Irrelevant Documents Automatically Detected. The results of this set of experiments are summarized in Table VII. We can see that in the automatic detection scenario, DSM and its regularization methods still work well and can outperform RM+. In addition, we can observe that the Outlier method achieves better performance among these three automatic irrelevance detection methods, and this observation is also consistent with the quality measurement results we just discussed above.

We now report the pairwise comparison results. Firstly, we compare RM and RM+ for each irrelevant documents detection method. For Outlier method, RM+ outperforms RM on most of the datasets significantly except for WT10G. These results imply that, the irrelevant documents detected by the Outlier method have a relatively good quality, and the direct removal of them from the feedback documents benefits the estimation of a better relevance model. In comparison, QPRP or L2R may not provide useful irrelevant documents.

Table VIII. Evaluation on DSM and Regularized DSM Combined with Original Query in Automatic Irrelevance Feedback

MAP (chg% over RM3)		**WSJ8792**	**AP8889**	**ROBUST2004**	**WT10G**
RM3 (baseline)		0.3717	0.3899	0.3491	0.2332
		Irrelevant Documents Detection Method: OutlierD			
DSM+q	SparseReg	0.3828(+2.99%)$^\gamma$	**0.4023**(+3.18%)$^\gamma$	0.3550(+1.69%)	**0.2318**(-0.60%)
	ContextReg	0.3837(+3.23%)$^\gamma$	0.4006(+2.74%)	0.3520(+0.83%)	0.2308(-1.03%)
	TaxicabReg	**0.3839**(+3.28%)$^\gamma$	0.3995(+2.46%)$^\gamma$	**0.3591**(+2.86%)$^\gamma$	0.2207(-5.36%)
		Irrelevant Documents Detection Method: QPRP			
DSM+q	SparseReg	**0.3868**(+4.06%)$^\gamma$	0.4013(+2.92%)$^\gamma$	0.3560(+1.98%)$^\gamma$	0.2369(+1.59%)
	ContextReg	0.3851(+3.61%)$^\gamma$	0.4012(+2.90%)$^\gamma$	0.3559(+1.95%)$^\gamma$	0.2373(+1.76%)
	TaxicabReg	0.3858(+3.79%)$^\gamma$	**0.4048**(+3.82%)$^\gamma$	**0.3648**(+4.50%)$^\gamma$	**0.2423**(+3.90%)$^\gamma$
		Irrelevant Documents Detection Method: L2R			
DSM+q	SparseReg	**0.3883**(+4.47%)$^\gamma$	**0.4008**(+2.80%)$^\gamma$	0.3529(+1.09%)	0.2405(+3.13%)$^\gamma$
	ContextReg	0.3880(+4.39%)$^\gamma$	0.3973(+1.90%)$^\gamma$	0.3520(+0.83%)	0.2402(+3.00%)$^\gamma$
	TaxicabReg	0.3815(+2.64%)$^\gamma$	**0.4008**(+2.80%)$^\gamma$	**0.3568**(+2.21%)$^\gamma$	**0.2503**(+7.33%)$^\gamma$

γ indicates statistically significant improvement over RM3 at level 0.05.

Similar to the explicit feedback scenario, in the automatic scenario, the performances of DSM– and RM+ are also very close, which is guaranteed by the theoretical analysis in Section 3.1. When we compare DSM and DSM–, we find that in the automatic detection scenario, the refinement of DSM is not helpful when the seed irrelevant data is of low quality. We then fix η as 1 (i.e., DSM– is equivalent to DSM), to help us focus on the DSM's regularization methods. The results show that DSM with Outlier detection can significantly outperform the baseline on some collections. Specifically, DSM with Outlier detection outperforms RM by 4.10%, 3.97% and 4.26% on WSJ8792, AP8889 and ROBUST2004, respectively.

Now we compare the DSM and DSM+ with three regularization methods. We find that the DSM with sparse regularization (SparseReg) and contextual regularization (ContextReg) show stable improvements over DSM on ROBUST2004 and WT10G, while their performance on the other smaller datasets do not show much improvement. In contrast, DSM with a Taxicab approximated regularization (TaxicabReg) can stably improve DSM on most of the datasets. Specifically, for DSM+TaxicabReg with Outlier methods, it can outperform RM by 7.94%, 5.62%, 9.14% and 7.88% respectively on WSJ8792, AP8889, ROBUST2004 and WT10G. It shows that the regularization method (especially the Taxicab based one) can work well when the seed irrelevant documents are detected automatically.

As for the parameter settings of the three regularization algorithms. Taxicab algorithm does not have an adjustable parameter in its formulation, while the other two algorithms have a parameter κ to control the degree of regularization. In the explicit scenario, κ was set to 10 since the DSM's performance is already very good and a small κ acts a fine tuning role. In the automatic scenario, however, the performance of DSM is not good enough due to the relatively low quality of the detected irrelevant documents. Therefore, we need a larger κ, e.g., 450 (used in the reported experiments), to make more influence on the original optimization problem. As we see in the results, a large κ for SparseReg and ContextReg may be not stable across different collections. On the other hand, TaxicabReg, which is free of adjustable parameter and directly integrates multiple evidences, is effective to improve the performance stability across different collections.

6.4.3. Comparisons with RM3 and PRM. Now, we adopt RM3 as a baseline. For a fair comparison, we combine the query expansion model given by the regularized DSM with the original query model, and denote the interpolated model as DSM+q. We first select the optimal interpolation parameter value in the interval [0,1] with increment 0.1 to obtain the best performance for RM3 on each collection. We then use the same

Table IX. Comparison of Regularized DSM (in Automatic Irrelevance Feedback) and Positional Relevance Model

MAP(chg% over PRM)		**WSJ8792**	**AP8889**	**ROBUST2004**	**WT10G**
PRM (baseline)		0.3572	0.3907	0.3344	0.2108
DSM+	SparseReg	0.3547(-0.70%)	0.3805(-2.61%)	0.3423(+2.36%)*	**0.2244**(+6.45%)*
($r_n = 0$,	ContextReg	0.3546(-0.73%)	0.3808(-2.53%)	0.3426(+2.45%)*	0.2241(+6.31%)*
OutlierD)	TaxicabReg	**0.3819**(+6.91%)**	**0.3966**(+1.51%)	**0.3560**(+6.46%)*	0.2162(+2.56%)*

* indicates statistically significant improvement over PRM at level 0.05

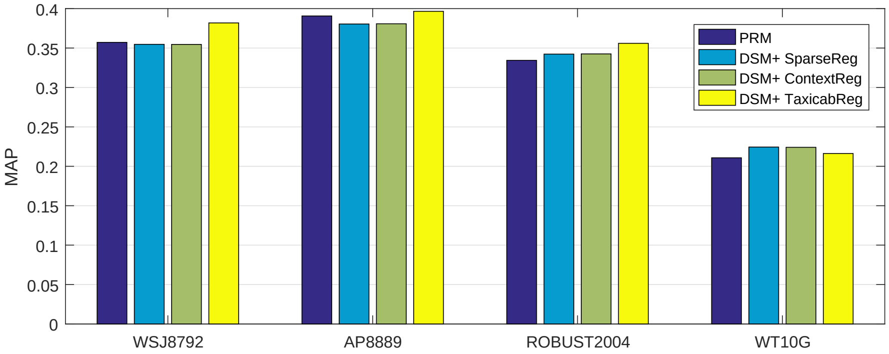

Fig. 3. Comparison of Regularized DSM (in Automatic Irrelevance Feedback) and Positional Relevance Model

parameter value for the regularized DSM. The results are reported in Table VIII. We can find that in most cases DSM+q can significantly outperform RM3.

Positional Relevance Model (PRM) [Lv and Zhai 2010] is a representative pseudo-relevance feedback method that exploits term positions and proximity. We implement PRM1 proposed in [Lv and Zhai 2010] with the window size parameter carefully tuned, and the best performances of PRM on each collection are reported. For automatic irrelevance feedback, as shown in Table IX and Figure 3, compared with PRM, DSM+ (with TaxicabReg) achieves significantly better performances on most collections.

6.5. Summary of Main Findings

In the explicit scenario, RM+ usually outperforms RM. This verifies our assumption that the removal of irrelevant feedback documents can benefit the query model estimation. As for the automatic scenario, RM+ does not achieve stable improvements over RM, which indicates that the automatically detected irrelevant documents are not good enough. RM+ with different automatic detection methods show different performance, and in general the outlier-based method achieves the best performance. The performance of DSM– is close to that of RM+. This is consistent with our theoretical analysis in Section 3.1 and demonstrates that the basic DSM (without the refinement) can separate the *seed* irrelevance distribution from the mixture distribution. DSM– performs well in explicit irrelevance feedback. However, with the same problem as RM+, it does not work stably for the automatic irrelevance feedback, leaving a room for developing advanced algorithms to improve the performance.

The difference between DSM and DSM– is that DSM has a refinement step with an adjustable parameter η. In the explicit scenario, DSM can significantly outperform DSM–, showing that the distribution refinement is helpful to improve retrieval performance. In contrast, for the automatic scenario, DSM's refinement step does not help much. This motivates us to investigate formal regularization methods for the further improvement. DSM+ with three regularization algorithms can help improve the original DSM. Compared with SparseReg and ContextReg, TaxicabReg can be more helpful to significantly and stably improve the retrieval performance in both explicit and automatic scenarios. When we further compare DSM with two strong baselines, namely RM3 and PRM, the results in automatic scenario show that DSM+q can significantly outperform RM3, and DSM+ can also significantly outperform PRM.

7. DISCUSSION

7.1. Discussion of Graded Relevance

In this paper, we assume that the relevance is binary, i.e., a document is either relevant or irrelevant. Therefore, the DSM's theoretical analysis in the previous section is concerned with three distributions: relevance, irrelevance and their mixture. From a more general point of view, relevance can be graded, and some test collections have already incorporated such graded relevance. Theoretically speaking, it is possible to incorporate the relevance grades into the DSM framework. For example, one possible solution is that we adjust the linear combination assumption in Eq. 1, so that the mixture distribution is combined by graded distributions, e.g., irrelevant distribution, moderately relevant distribution and highly relevant distribution. The procedure of DSM can be divided into two steps: separating irrelevant distribution from the mixture distribution first, and then separating moderately relevant distribution from the remained distribution.

For some applications, the irrelevant distribution may be mixed with a moderately relevant distribution, or with a highly relevant distribution, leading to different mixture distributions. DSM can then be applied directly, with different input mixture distributions. One problem is that, intuitively, separating a irrelevant distribution from a moderately relevant distribution can be more difficult than that from a highly relevant distribution. The seed irrelevance distribution may be more distinct from the highly relevant distribution, while the moderately relevant distribution can be less easily distinguished from the irrelevant distribution, especially when the quality of seed irrelevance distribution is not satisfactory. We will investigate this problem in-depth in our future work.

7.2. Discussion of Iteratively Updated DSM

As afore-mentioned, the basic theoretical framework of DSM can be generalized, resulting in variants of DSM that may be useful for other application scenarios. An interesting scenario would be that the relevance distribution estimated by DSM can be iteratively updated, in order to approximate the true relevance distribution more accurately. In such a scenario, a user or a simulated user picks irrelevant documents in each loop and accordingly the DSM iteratively separates the irrelevant term distribution from the mixture distribution. This approach could be applicable to online search engines, since a large number of online users can contribute to the iterative updating procedure. In this sense, DSM can make the search system evolutionary. This is similar to an active learning procedure, where a number of research problems could arise, e.g., a quality control of the iteratively updated irrelevant distribution, the new cost function in terms of the effort from the user's point of view, the estimation precision from the system's perspective, and the time cost spent in the iterative process.

8. CONCLUSIONS AND FUTURE WORK

This article aims to substantially enrich and advance a novel Distribution Separation Method (DSM). The basic idea of DSM is to separate a seed irrelevance term distribution from a mixture term distribution (derived from all feedback documents), and endeavor to yield a good approximation of the true relevance term distribution, given a seed irrelevance distribution. When the seed irrelevant documents are obtained by the explicit irrelevance feedback, DSM achieved good retrieval performance according to TREC evaluation results.

In this paper, we have developed a regularization framework with three algorithms, each with a regularization term added in the objective function of the original DSM. The first two algorithms can be feasibly computed. The third algorithm is computationally intractable, for which we developed an approximation solution that corresponds to a minimization for an optimization problem based on the Taxicab metric.

In addition, we have broadened the application scenarios of the original DSM, including both explicit irrelevance feedback to pseudo (automatic) irrelevance feedback. We have developed three re-ranking based methods for the automatic detection of seed irrelevant documents. To measure the quality of automatically detected irrelevant documents, we also proposed a quality metric and conducted a separate evaluation on quality of the irrelevant documents. This evaluation is helpful for investigating the irrelevance detection issue in-depth.

For the empirical evaluation, we have constructed a series of step-by-step performance comparisons between two different methods/configurations. In this way, we can observe the contribution of each method or a component of DSM. Based on these observations, we summarize our main findings in Section 6.5. DSM and its regularization methods can work well in explicit irrelevance feedback. With the help of proposed automatic irrelevant documents detection methods, the DSM, and particularly the regularized DSM, can achieve good performance in the automatic irrelevance feedback scenario. In the automatic scenario, DSM+ even significantly outperforms two strong baselines, namely RM3 and PRM.

It is worth mentioning that the core algorithms of DSM and its regularization are based on probability distributions rather than documents. Generally speaking, in our future work, DSM and its regularized versions are expected to be also applicable to other IR tasks or other fields, since there is no restriction in DSM that the distribution should only be about query terms or expanded query terms. For example, the relevance distribution can be obtained in the implicit feedback scenario, and it is interesting to construct more systematic study on DSM in the implicit scenario. In addition, for other dimensions of relevance, e.g., novelty and diversity, the relevance term distributions can be also better derived and approximated, by applying DSM on the corresponding mixture distributions and seed irrelevant ones.

The current version of DSM is limited to binary relevance. In the future, we plan to investigate the generalized DSM with graded relevance. The iteratively updated DSM using the active learning approach is also an interesting research topic. Furthermore, we will investigate the adaptation and application of DSM to other applications, such as online tweets filtering.

REFERENCES

Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah Larkey, Xiaoyan Li, Donald Metzler, Mark D. Smucker, Trevor Strohman, Howard Turtle, and Courtney Wade. 2004. Umass at trec 2004: Novelty and hard. In *TREC '04*.

Fabrizio Angiulli and Clara Pizzuti. 2002. Fast Outlier Detection in High Dimensional Spaces. In *PKDD '02*. Springer-Verlag, London, UK, UK, 15–26.

Paul N Bennett, Ryen W White, Wei Chu, Susan T Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the impact of short-and long-term behavior on search personalization. In *SIGIR '12*. ACM, 185–194.

Chris Buckley and Gerard Salton. 1995. Optimization of relevance feedback weights. In *SIGIR'95*. ACM, 351–357.

Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning* 11 (2010), 23–581.

Olivier Chapelle and Ya Zhang. 2009. A Dynamic Bayesian Network Click Model for Web Search Ranking. In *WWW '09*. ACM, 1–10.

Nitesh V. Chawla, Nathalie Japkowicz, and Aleksander Kotcz. 2004. Editorial: Special Issue on Learning from Imbalanced Data Sets. *SIGKDD Explor. Newsl.* 6, 1 (2004), 1–6.

Fernando Diaz. 2016. Pseudo-Query Reformulation. In *Advances in Information Retrieval - 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20-23, 2016. Proceedings*. 521–532.

Susan Dumais, Thorsten Joachims, Krishna Bharat, and Andreas Weigend. 2003. SIGIR 2003 workshop report: implicit measures of user interests and preferences. 37, 2 (2003), 50–54.

Mark D. Dunlop. 1997. The effect of accessing non-matching documents on relevance feedback. *TOIS* 15 (1997), 137–153.

Steve Fox, Kuldeep Karnawat, Mark Mydland, Susan Dumais, and Thomas White. 2005. Evaluating implicit measures to improve web search. *TOIS* 23, 2 (2005), 147–168.

Jianfeng Gao, Wei Yuan, Xiao Li, Kefeng Deng, and Jian-Yun Nie. 2009. Smoothing clickthrough data for web search ranking. In *SIGIR '09*. ACM, 355–362.

Ben He, Jimmy Xiangji Huang, and Xiaofeng Zhou. 2011. Modeling term proximity for probabilistic information retrieval models. *Information Sciences* 181, 14 (2011), 3017–3031.

Bernard J Jansen, Amanda Spink, and Tefko Saracevic. 2000. Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *IPM* 36, 2 (2000), 207–227.

Maryam Karimzadehgan and ChengXiang Zhai. 2011. Improving retrieval accuracy of difficult queries through generalizing negative document language models. In *CIKM '11*. 27–36.

John D. Lafferty and ChengXiang Zhai. 2001. Document Language Models, Query Models, and Risk Minimization for Information Retrieval. In *SIGIR '01*. 111–119.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance-Based Language Models. In *SIGIR '01*. 120–127.

Chao Liu, Fan Guo, and Christos Faloutsos. 2010. Bayesian Browsing Model: Exact Inference of Document Relevance from Petabyte-Scale Data. *TKDD* 4, 4 (Oct. 2010), 19:1–19:26.

Yuanhua Lv and ChengXiang Zhai. 2010. Positional relevance model for pseudo-relevance feedback. In *SIGIR '10*. 579–586.

Massimo Melucci. 2012. Contextual Search: A Computational Framework. *Foundations and Trends in Information Retrieval* 6, 4-5 (2012), 257–405.

Paul Ogilvie and Jamie Callan. 2002. Experiments using the Lemur toolkit. In *TREC '02: Proceedings of the ACM 11th Text Retrieval Conference*. 103–108.

Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. 2000. Efficient algorithms for mining outliers from large data sets. *SIGMOD Rec.* 29, 2 (2000), 427–438.

C. J. van Rijsbergen. 1979. *Information Retrieval*. Butterworth-Heinemann.

Joseph John Rocchio. 1971. Relevance Feedback in Information Retrieval. In *The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice Hall, 313–323.

Joseph L. Rodgers and Alan W. Nicewander. 1988. Thirteen Ways to Look at the Correlation Coefficient. *The American Statistician* 42 (1988), 59–66.

Amit Singhal, Mandar Mitra, and Chris Buckley. 1997. Learning Routing Queries in a Query Zone. In *SIGIR '97*. 25–32.

Dawei Song, Qiang Huang, Stefan Rüger, and Peter Bruza. 2008. Facilitating query decomposition in query language modeling by association rule mining using multiple sliding windows. In *ECIR '08*. 334–345.

Damiano Spina, Julio Gonzalo, and Enrique Amigó. 2013. Discovering filter keywords for company name disambiguation in twitter . *Expert Systems with Applications* 40, 12 (2013), 4986–5003.

Tao Tao and ChengXiang Zhai. 2006. Regularized Estimation of Mixture Models for Robust Pseudo-relevance Feedback. In *SIGIR '06*. ACM, 162–169.

Anastasios Tombros and Rijsbergen C. J. van. 2004. Query-sensitive similarity measures for information retrieval. *KAIS* 6, 5 (2004).

Jun Wang and Jianhan Zhu. 2009. Portfolio theory of information retrieval. In *SIGIR '09*. 115–122.

Xuanhui Wang, Hui Fang, and ChengXiang Zhai. 2008. A study of methods for negative relevance feedback. In *SIGIR '08*. 219–226.

Ryen W White, Ian Ruthven, and Joemon M Jose. 2005. A Study of Factors Affecting the Utility of Implicit Relevance Feedback. In *SIGIR '05*. 35–42.

ChengXiang Zhai and John Lafferty. 2001a. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01*. 403–410.

ChengXiang Zhai and John Lafferty. 2001b. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *SIRIR '01*. 334–342.

ChengXiang Zhai, Atulya Velivelli, and Bei Yu. 2004. A Cross-collection Mixture Model for Comparative Text Mining. In *KDD '04*. ACM, 743–748.

Peng Zhang, Yuexian Hou, and Dawei Song. 2009. Approximating true relevance distribution from a mixture model based on irrelevance data. In *SIGIR '09*. 107–114.

Peng Zhang, Qian Yu, Yuexian Hou, Dawei Song, Jingfei Li, and Bin Hu. 2016. Generalized Analysis of a Distribution Separation Method. *Entropy* 18, 4 (2016), 105.

Yi Zhang and Wei Xu. 2008. Fast exact maximum likelihood estimation for mixture of language model. *Inf. Process. Manage.* 44, 3 (2008), 1076–1085.

Jiashu Zhao, Jimmy Xiangji Huang, and Ben He. 2011. CRTER: using cross terms to enhance probabilistic information retrieval. In *SIGIR '11*. 155–164.

Guido Zuccon and Leif Azzopardi. 2010. Using the Quantum Probability Ranking Principle to Rank Interdependent Documents. In *ECIR'10*. 357–369.

Guido Zuccon, Leif Azzopardi, and C. J. van Rijsbergen. 2010. Has portfolio theory got any principles?. In *SIGIR '10*. 755–756.