AXIOMATIC ANALYSIS OF SMOOTHING METHODS IN LANGUAGE MODELS FOR PSEUDO-RELEVANCE FEEDBACK

BY

HUSSEIN HAZIMEH

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Professor ChengXiang Zhai

Abstract

Pseudo-Relevance Feedback (PRF) is an important general technique for improving retrieval effectiveness without requiring any user effort. Several state-of-the-art PRF models are based on the language modeling approach where a query language model is learned based on feedback documents. In all these models, feedback documents are represented with unigram language models smoothed with a collection language model. While collection language model-based smoothing has proven both effective and necessary in using language models for retrieval, we use axiomatic analysis to show that this smoothing scheme inherently causes the feedback model to favor frequent terms and thus violates the IDF constraint needed to ensure selection of discriminative feedback terms. To address this problem, we propose replacing collection language model-based smoothing in the feedback stage with additive smoothing, which is analytically shown to select more discriminative terms. Empirical evaluation further confirms that additive smoothing indeed significantly outperforms collection-based smoothing methods in multiple language model-based PRF models.

To my parents, for their love and support.

Acknowledgments

I owe my gratitude to my adviser ChengXiang Zhai whose dedication and advice made this work possible.

Table of Contents

Chapter 1

Introduction

Feedback is an essential component in every modern retrieval system as it allows incorporating user preferences. The most reliable type of feedback is relevance feedback where users would label the top documents returned by a retrieval system as relevant or non-relevant. As relevance feedback is a tedious task that requires labeling large numbers of documents in the collection, pseudo relevance feedback (PRF) is commonly used as an alternative. In PRF, the top documents returned by the retrieval system are assumed to be relevant and are used to expand the query. Although PRF is not as reliable as relevance feedback, empirical studies have shown that it is an effective general technique for improving retrieval accuracy [2, 3, 4, 5, 6]. Since it requires no user effort, the technique can be applied in any retrieval system.

Several types of PRF models exist in the literature (see Chapter 1 for a detailed review), among which language model (LM) based PRF methods are both theoretically well-grounded and empirically effective [4, 7]. Many PRF models have been proposed, including the divergence minimization model [4], the mixture model [4], the relevance model [3], and the geometric relevance model [8], in addition to many other improved variants [5, 6].

While these models differ in how they are derived, they generally take a set of top-ranked documents from an initial retrieval result as input and attempt to estimate a unigram language model, referred to as the feedback language model, based on these documents to capture their topic. The feedback language model θ_F can then be linearly interpolated with the original query language model θ_Q to form an "expanded" query language model:

$$\theta_Q^{'} = (1 - \alpha)\ \theta_Q + \alpha\ \theta_F$$

This chapter and the subsequent ones are a joint work with ChengXiang Zhai which has been published in ACM ICTIR15 (see [1] for bibliographic information)

where $\alpha \in [0, 1]$ is the interpolation coefficient that determines the weight assigned to the feedback language model. This is similar to how the Rocchio feedback algorithm works in the vector space model [9]. The effectiveness of a PRF method is thus directly determined by the quality of the estimated feedback language model θ_F.

In all the current LM-based PRF models, the estimated feedback LM is based on the aggregation of all the language models of the feedback documents. The aggregation is achieved using an averaging function (such as the arithmetic or geometric mean), and it may also involve weighting each feedback document based on how well it matches the query (i.e., retrieval score of each feedback document in the initial retrieval result), though uniform weighting is also often used. The intuition captured in such an aggregation-based estimate is to favor words that have high probabilities according to all the individual LMs of feedback documents (i.e., occur frequently in all the feedback documents).

Such an aggregation function alone, however, would not consider the occurrences of these terms in the global collection, and indeed it would tend to give high probabilities to many non-informative popular words in the collection that are intuitively not useful for expanding a query. Thus, some of these models further rely on the use of a collection language model to penalize the terms that are too common in the collection. As a result, the terms that have high probabilities in the final estimated feedback LM would be those that occur frequently in all feedback documents, but not very frequently in the entire collection; this is the same effect as what the Rocchio algorithm achieves in the vector space retrieval model when TF-IDF weighting is used.

Formally, let the feedback set be $F = \{D_1, D_2, ..., D_n\}$, where D_i is the ith feedback document. In a language model-based approach to PRF, we consider a unigram language model, θ_i, estimated based on each feedback document D_i, in addition to the collection LM, θ_C, which is estimated based on the entire collection of documents. The goal of a PRF method is to estimate a feedback LM, denoted by θ_F, based on all the document LMs, $\theta_1, ..., \theta_n$, and the collection LM θ_C.

Let w be any feedback word that appears in at least one of the documents in F, then a state-of-the-art PRF method would, in general, assign w a

probability $P(w|\theta_F)$ defined as follows:

$$P(w|\theta_F) \propto f\Bigg(A\Big(P(w|\theta_1), ..., P(w|\theta_n)\Big), P(w|\theta_C)\Bigg) \tag{1.1}$$

where $A : \mathbb{R}^N \rightarrow \mathbb{R}$ is an aggregation function that combines the probabilities of words in each of the feedback documents. For example, A can be the arithmetic or geometric mean. $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ is an arbitrary function that is increasing in the first argument (so that words that occur frequently in feedback documents would tend to have a higher value of f) and decreasing in the second (so that words that occur frequently in the collection would tend to have a smaller value of f).

This generalized view of the LM-based PRF models is intuitive as we want the final probability of the word to be high for relevant discriminative words, i.e. those that appear in most of the feedback documents and have a low to moderate probability of occurrence in the collection LM.

As in all cases of language model estimation, the feedback documents' LMs (θ_is) have to be smoothed to account for the fact that a document is generally a very small sample for estimating a word distribution over the entire vocabulary. Smoothing is also necessary to avoid assigning zero probabilities to (many) unseen words in documents (an inevitable consequence of the commonly used Maximum Likelihood estimator). The general motivation behind smoothing the feedback documents' language models is that a word may not occur in some of the feedback documents, and not smoothing its probability might overpenalize it. For example, if the PRF model has the function A as the geometric mean and the feedback word occurs in all the feedback documents except one, then the probability assigned to the word will be zero if smoothing is not used.

So far, in virtually all work on LM-based PRF models, smoothing of $P(w|\theta_i)$ has been based on some form of interpolation of the maximum likelihood estimate and the probability of the word according to the collection language model, $P(w|\theta_C)$, which is proportional to the number of occurrences of the word in the whole collection. Dirichlet prior and Jelinek-Mercer are especially popular choices of collection-based smoothing because of their good performance when used in a retrieval function such as Query Likelihood or Kullback-Leibler (KL) divergence [7]. Indeed, smoothing with a collection

language model has not only proven to be effective for retrieval, but also essential for achieving the effect of IDF weighting in a retrieval function [7].

However, in this paper, we show that using collection LM-based smoothing, while effectively avoids the assignment of zero probabilities, is not suitable for LM-based PRF models and can lead to the selection of very frequent and non-informative feedback terms, thus making the PRF model unable to select discriminative feedback terms. Specifically, the problem lies in that when such smoothing is used, the function A will favor words that occur frequently in the collection, and consequently f will assign higher weights to these frequently occurring words. However, PRF models are expected to penalize frequent words (when two words have equal frequencies in the feedback documents, the word with lower frequency in the collection should be favored). While it is still possible to enforce the preference for selecting discriminative words through having $P(w|\theta_C)$ as the second argument in f, our axiomatic analysis of the current PRF models reveals that smoothing based on the collection LM would cause violation of the IDF constraint [6] and introduce an **inherent interference** between smoothing and favoring discriminative words. We show analytically the following dilemma: if we want to avoid favoring common words caused by collection-based smoothing, we must substantially increase the influence of the collection LM, which, however, would result in an extremely skewed word distribution with probability mass mostly concentrated on the matched original query words, restricting the benefit of feedback.

To address this problem, we propose replacing the collection LM-based smoothing strategy with additive smoothing which does not favor frequent words, yet can also dynamically set probabilities for unseen words in adaptation to the document length. We show that using additive smoothing in the current PRF models would ensure that they will not favor common terms anymore, and that some PRF methods will consistently favor discriminative terms thus satisfying the IDF constraint. Empirical evaluation further confirms that additive smoothing indeed significantly outperforms collection-based smoothing methods in multiple LM-based PRF models.

In the rest of the paper, we first review related work in Chapter 1 and then briefly introduce several representative LM-based PRF methods in Chapter 3. In Chapter 4, we present a detailed axiomatic analysis of the collection LM-based smoothing methods with the IDF constraint, and prove theorems to

show analytically the inherent interference of smoothing with a collection LM and satisfying the IDF constraint. In Chapter 5, we then introduce the new strategy of using additive smoothing for PRF and show that it solves the problem of favoring common words in the collection. In Chapter 6, we propose a new measure to empirically assess how discriminative the feedback terms output by a PRF method are. We present our experiment design and results in Chapter 7, and summarize our work in Chapter 8.

Chapter 2

Related Work

Relevance feedback has been studied first in the vector space model, and the Rocchio feedback algorithm [9], while proposed several decades ago, remains a state-of-the-art feedback method today. Relevance feedback has also been studied in classic probabilistic models and is the basis of the RSJ model [10] and its extensions e.g., [11]. (More discussion on relevance feedback can be found in [12]).

When relevance judgments are available, relevance feedback is generally very effective [10, 13, 14, 15]. While effective, however, relevance feedback requires users to make effort to judge the relevance of the top-ranked documents, which is impractical in many situations. In contrast, pseudo relevance feedback (PRF), also called blind feedback or automatic feedback [16, 2], simply assumes a certain number of top-ranked results from an initial retrieval round to be relevant and performs feedback under this assumption, thus it can be applied to any retrieval system without requiring any effort from the user. PRF leverages the terms in top-ranked documents that have high correlation with the query terms in order to expand the query, and it has proven to be a general effective technique for improving retrieval accuracy, especially for satisfying high-recall information needs.

Most relevance feedback methods can be adapted for pseudo-relevance feedback, but the most effective PRF methods seem to be those based on statistical language models, including, e.g., the relevance model [3], mixture model and divergence minimization [4], and many extensions of these models (e.g., [5, 6, 17]).

Although many LM-based PRF models have been proposed and studied, there seems to be no single winner. [5] compared different LM-based PRF methods and empirically studied the reasons behind the better performance in some of the models. According to this study, the Relevance Model, which uses the arithmetic mean of feedback language models, appears to be slightly

advantageous over other models, but a recent work [8] shows that changing the arithmetic mean in the Relevance Model to geometric mean can be beneficial, which implicitly suggests that the divergence minimization model [4] may be also advantageous.

To gain better understanding of relative strengths and weaknesses of different PRF models, [6] performed an axiomatic analysis on several well-known PRF models and studied whether these models satisfy a set of desirable properties. One important property studied by [6] is the IDF effect which states that the PRF model should assign higher probabilities to feedback terms with lower occurrence in the collection. This is an important property to satisfy since when performing feedback we are interested in discriminative words that are usually not so common in the collection. Our work extends this previous work to analyze the relation between smoothing and the IDF effect more accurately and reveals new insights about the interference of smoothing with a collection LM and achieving the IDF effect.

[17] reported that using additive smoothing in the feedback stage in addition to maximizing the entropy in the divergence minimization model led to performance improvements over several datasets. In our analysis, we axiomatically explain the reason behind the improvement in [17]'s new model. We also derive a more general new result: additive smoothing is generally preferred to collection-based smoothing for LM-based PRF methods and this is confirmed in our empirical experiments. Thus, our work directly results in multiple new LM-based PRF models that are potentially more effective than the existing ones and can be used immediately in all retrieval systems.

Chapter 3

Representative PRF Models

As background, in this chapter, we give an overview of several representative PRF models that are both efficient and effective.

3.1 Divergence Minimization Model

The Divergence Minimization Model (DMM) estimates $P(w|\theta_F)$ in a way similar to the Rocchio algorithm: θ_F is chosen such that it is close to the centroid of the relevant documents and far away from the collection language model [4]. Formally, the DMM solves the following optimization problem:

$$\theta_F = \arg\min_{\theta} \left(\frac{1}{|F|} \sum_{i=1}^{|F|} D(\theta \;||\; \theta_i) - \lambda D(\theta \;||\; \theta_C) \right)$$

where $D(.||.)$ is the KL divergence and $0 < \lambda < 1$ is a free parameter that determines the extent to which common words in the collection are penalized. Since the objective is a strongly convex function being minimized over a closed convex set (probability simplex), a unique minimizer θ_F exists. Fortunately, the problem has a closed form solution and can be solved using Lagrange multipliers while forcing the constraint $\sum_{w \in V} P(w|\theta_F) = 1$. The solution has the following form:

$$P(w|\theta_F) \propto \frac{\left(\sqrt[|F|]{\prod_{i=1}^{|F|} P(w|\theta_i)} \right)^{\frac{1}{1-\lambda}}}{P(w|\theta_C)^{\frac{\lambda}{1-\lambda}}}$$

where $\propto$ indicates that the probability normalization factor has been omitted. The solution agrees with the intuition of the DMM: the probability of a feedback term is proportional to its occurrence in the feedback set and inversely

proportional to that in the collection. Note that the DMM is an instantiation of Equation (1) where A is the geometric mean and $f(x_1, x_2) = \left(\frac{x_1}{x_2^{\lambda}}\right)^{\frac{1}{1-\lambda}}$. Although the DMM has a good theoretical justification, several studies have indicated that it suffers in performance. In the next chapter, we show that the reason behind the poor performance is mainly due to the collection-based smoothing.

3.2 Relevance Model

The Relevance Model (RM) is a well-known LM-based PRF method that has an intuitive probabilistic interpretation and has proven to be effective in several empirical studies. It assumes that each information need (i.e. a topic the user is interested in) has an underlying relevance model R, which is a multinomial distribution over words. Furthermore, it assumes that the query words and the feedback documents' words are randomly sampled from the distribution R, and then tries to estimate R based on simplifying assumptions. Below we present one instantiation of the relevance model:

$$P(w|\theta_F) = \sum_{i=1}^{|F|} P(\theta_i|q)P(w|\theta_i)$$

where q is the query and $P(\theta_i|q)$ can be estimated using the query likelihood (i.e. $P(q|\theta_i)$) assuming a uniform prior on the feedback documents:

$$P(\theta_i|q) = \frac{P(q|\theta_i)P(\theta_i)}{\sum_{i=1}^{|F|} P(q|\theta_i)P(\theta_i)} = \frac{P(q|\theta_i)}{\sum_{i=1}^{|F|} P(q|\theta_i)}$$

When linearly interpolated with the original query, the model we presented above is usually referred to as RM3 [18]. RM3 is an instantiation of Equation (1) where $f = A$ with A being a weighted arithmetic mean.

3.3 Geometric Relevance Model

[8] introduced the Geometric Relevance Model (GRM) which is a refined version of the Relevance Model that uses the normalized geometric mean instead

of the weighed arithmetic mean in the aggregation function. The normalized geometric mean has been shown to be a better approximation for the center of mass in the statistical manifold which was the main motivation behind introducing this model [8]. The GRM assigns the following probability to each word in the feedback LM:

$$P(w|\theta_F) \propto \prod_{i=1}^{|F|} P(w|\theta_i)^{P(\theta_i|q)}$$

where $P(\theta_i|q)$ is estimated in the same way as in the RM. The GRM can be viewed as an instantiation of Equation (1) with $f = A$, A being a weighted geometric mean.

Chapter 4

Axiomatic Analysis of PRF Models

In this chapter, we analyze the effect of collection LM-based smoothing in the feedback stage on the quality of the terms returned by the three models surveyed in Chapter 3. We do the analysis by inspecting the IDF effect in each model. The IDF effect is a desirable property to have in any PRF model, and it states the following: given two words with the same number of occurrences in the feedback documents, the feedback model should assign a higher probability to the word with higher IDF [6]. For instance, if we observe that the words "the" and "machine" have the *same* number of occurrences in the feedback documents, then we want "machine" to have a higher probability than "the" because the former is more discriminative. This effect is desirable as long as it is not overpenalizing words with low IDF.

Let $c(w, D_i)$ be the count of the word w in the ith feedback document. Formally, we define the IDF effect as in [6]:

IDF Effect. Given any two words w_1 and w_2 from the feedback collection F such that $c(w_1, D_i) = c(w_2, D_i)\ \forall i$ and $P(w_1|\theta_C) < P(w_2|\theta_C)$, a PRF model that outputs a θ_F with $P(w_1|\theta_F) > P(w_2|\theta_F)$ is said to support the IDF effect.

In what follows we assume that document LMs are smoothed using Dirichlet prior smoothing in which the probability of the word w occurring in the ith feedback document is smoothed according to:

$$P(w|\theta_i) = \frac{c(w, D_i) + \mu P(w|\theta_C)}{|D_i| + \mu}$$

where $\mu > 0$ is the mean of a Dirichlet distribution and $|D_i|$ is the size of the document D_i. Typically, a high value of μ such as 1000 is used to smooth document language models in both the retrieval and feedback stages. While the analysis is performed only for Dirichlet prior smoothing, the results still

hold for several other collection-based smoothing methods such as Jelinek-Mercer.

4.1 Divergence Minimization Model

Assume w_1 and w_2 are two feedback words as in the definition of the IDF effect . To check if the DMM supports the IDF effect [1], we analyze the sign of $\log P(w_1|\theta_F) - \log P(w_2|\theta_F)$ that should be positive in order to support the effect.

$$\log P(w_1|\theta_F) - \log P(w_2|\theta_F) \propto$$
$$\sum_{i=1}^{|F|} \Big(\log \frac{c(w, D_i) + \mu P(w_1|\theta_C)}{c(w, D_i) + \mu P(w_2|\theta_C)} - \lambda \log \frac{P(w_1|\theta_C)}{P(w_2|\theta_C)} \Big) \tag{4.1}$$

where $c(w, D_i) = c(w_1, D_i) = c(w_2, D_i)$. Analyzing the sign of (2) directly is a tedious task. In order to facilitate the analysis, we will get an attainable lower bound on (2) and then analyze the sign of the lower bound.

Since (2) is an increasing function in the variable $c(w, D_i)$, the lowest possible value for (2) occurs when w_1 and w_2 appear only once and only in one document in the feedback collection, i.e. when $c(w_1, D_j) = c(w_2, D_j) = 1$ for some j and $c(w_1, D_i) = c(w_2, D_i) = 0 \ \ \forall \ D_i \in F \ \ \text{s.t.} \ \ D_i \neq D_j$. This implies that if (2) has a positive sign in the case when w_1 and w_2 appear only once and only in document D_j, then the model will support the IDF effect as this choice of words minimizes (2). Choosing the latter pair of words, we can simplify (2) to:

$$\log \frac{1 + \mu P(w_1|\theta_C)}{1 + \mu P(w_2|\theta_C)} - (|F|\lambda - |F| + 1) \log \frac{P(w_1|\theta_C)}{P(w_2|\theta_C)} \tag{4.2}$$

Solving for μ that would make (3) positive and assuming $0 < \lambda < 1$, we get

[1] The IDF effect for DMM has been already studied in [6], however, we discovered a mistake in their derivation: the inequality under Equation (4) should be $\log(\frac{x}{y}) < \delta \log(\frac{x}{y})$ instead of the other way around, which made them arrive to the wrong conclusion that the DMM always (mildly) supports the IDF effect

an upper bound on μ:

$$\mu \leq \frac{k-1}{P(w_1|\theta_C) - kP(w_2|\theta_C)} \tag{4.3}$$

where

$$k = \left(\frac{P(w_1|\theta_C)}{P(w_2|\theta_C)} \right)^{|F|\lambda - |F| + 1}$$

Now that we got an upper bound on μ, we are ready to characterize the IDF behavior of the DMM.

Theorem 1. For $0 < \lambda < \frac{|F|-1}{|F|}$, the DMM cannot support the IDF effect, and for $\frac{|F|-1}{|F|} < \lambda < 1$, $\exists\ \mu > 0$ such that the DMM supports the IDF effect.
Proof. Any value of μ that satisfies (4) will make the DMM enforce the IDF effect since satisfying (4) will make the lowest possible value of (2) positive. We have two cases to consider:

Case 1: If $0 < \lambda < \frac{|F|-1}{|F|}$ then the right hand side of (4) will be negative since $k > 1$, but μ is the mean of a Dirichlet distribution which cannot be negative. Thus, in this case, the DMM does not support the IDF effect as we identified a choice of w_1 and w_2 for which w_2 will be favored no matter what μ is. It is also important to note that while the DMM does not support the IDF effect in this case, there may be other choices of w_1 and w_2 for which w_1 (the word with higher IDF) will be favored. However, to support the IDF effect, a model should favor w_1 for any choice of w_1 and w_2, and this is not the case here.

Case 2: If $\frac{|F|-1}{|F|} < \lambda < 1$ then the right hand side of (4) will be strictly positive since $k < 1$. Therefore, the DMM supports the IDF effect in this case for a range of μ values (the range can be obtained by choosing w_1 and w_2 that minimize the right-hand side of (4)). □

One might think that case (2) will lead to performance improvements as the IDF effect is being enforced. However, the condition we got on λ requires it to be close to 1 since the number of feedback documents $|F|$ is usually chosen to be 10 or more.

Theorem 2. If $\lambda \approx 1$, the DMM will assign all the probability mass to exactly one of the feedback terms.

Proof. When λ approaches 1, the DMM will assign one feedback word most of the probability mass, and the other feedback terms will be assigned negligible probabilities. To see why this phenomenon happens, we rewrite $P(w|\theta_F)$ as follows:

$$P(w|\theta_F) \propto \left(\frac{A(w|F)}{P(w|\theta_C)^{\lambda}} \right)^{\frac{1}{1-\lambda}} \tag{4.4}$$

where $A(w|F)$ is the geometric mean of the probabilities of the word w in all the feedback documents. Let w_h be the feedback word with the highest $\frac{A(w|F)}{P(w|\theta_C)}$ ratio among all other feedback words. This word will probably appear a lot in the feedback documents and will have a high IDF. Now, to calculate the final probability assigned by the DMM to the word w_h, we have to normalize by the sum of the probabilities of all other feedback terms. Thus, the final probability of w_h can be written as:

$$P(w_h|\theta_F) = \frac{\left(\frac{A(w_h|F)}{P(w_h|\theta_C)^{\lambda}} \right)^{\frac{1}{1-\lambda}}}{\left(\frac{A(w_h|F)}{P(w_h|\theta_C)^{\lambda}} \right)^{\frac{1}{1-\lambda}} + \sum_{w \in F s.t. w \neq w_h} \left(\frac{A(w|F)}{P(w|\theta_C)^{\lambda}} \right)^{\frac{1}{1-\lambda}}} \tag{4.5}$$

Taking the limit as λ tends to 1, we have:

$$\lim_{\lambda \to 1} P(w_h|\theta_F) = 1 \tag{4.6}$$

where the limit is equal to 1 since the terms corresponding to w_h in the numerator and denominator have the highest order. In practice, setting λ to any value near 1 will make the word w_h dominate the probability of the feedback language model. Note that in the analysis we assumed that the word with the highest $\frac{A(w|F)}{P(w|\theta_C)}$ ratio is unique. In the unlikely event where more than one word have the highest ratio, the probability mass will be split equally between such words. □

We conclude that in case (2), when $\frac{|F|-1}{|F|} < \lambda < 1$, one of the feedback terms will essentially get all the probability. Since the original query words usually have the highest occurrence in the feedback set in addition to relatively high IDF, probably one of the query words will be the w_h considered in the analysis above (although there are cases where a non-query word can have the highest ratio and dominate the feedback LM).

This analysis confirms the empirical studies performed on the divergence minimization model before. [4] noticed that when $|F| = 10$, the DMM's performance severely drops for $\lambda > 0.9$. The reason behind the drop is clear now: $\lambda > 0.9$ falls in case (2) of our analysis where the DMM will assign most of the probability to one term, and consequently the performance will be similar to that of the search engine without feedback. Also, several other studies indicated that the DMM suffers in performance even when its parameter λ is tuned [5, 6]. Such studies found that λ values in the range $(0.1, 0.4)$ give the highest performance which was still below that of other well-known feedback techniques. Small values of λ fall into case (1) in our analysis where the performance drop is due to the lack of support for the IDF effect, and this is in turn due to smoothing using the collection language model in the feedback stage. To conclude, we have clearly identified the problem with the DMM as previous attempts in the literature failed to clearly pinpoint the problem; using a collection LM to smooth the feedback documents will cause the DMM to suffer in performance no matter what the choices of the parameters λ and μ are.

4.2 Relevance Model

Assuming w_1 and w_2 are as in the definition of the IDF effect, we analyze the sign of $P(w_1|\theta_F) - P(w_2|\theta_F)$ when Dirichlet prior smoothing is used:

$$P(w_1|\theta_F) - P(w_2|\theta_F) = \sum_{i=1}^{|F|} \left(P(\theta_i|q) \frac{\mu \Big(P(w_1|\theta_C) - P(w_2|\theta_C) \Big)}{|D_i| + \mu} \right) < 0 \tag{4.7}$$

(8) is unconditionally negative for any valid value of μ implying that the currently used RM does not support the IDF effect, and on the contrary, it consistently favors more frequent words in the collection. (8) also shows that the longer the feedback documents are, the lower is the extent to which common words are rewarded. Clearly, this makes the RM's performance dependent on the length of the feedback documents.

4.3 Geometric Relevance Model

Similar to the case of the RM, we analyze the sign of $\log P(w_1|\theta_F) - \log P(w_2|\theta_F)$ where w_1 and w_2 are as in the definition of the IDF effect:

$$\log P(w_1|\theta_F) - \log P(w_2|\theta_F) = \sum_{i=1}^{|F|} \left(P(\theta_i|q) \log \frac{c(w, D_i) + \mu P(w_1|\theta_C)}{c(w, D_i) + \mu P(w_2|\theta_C)} \right) < 0 \quad (4.8)$$

where $c(w, D_i) = c(w_1, D_i) = c(w_2, D_i)$. Since (9) is negative, the GRM does not support the IDF effect and also suffers from the problem of consistently favoring frequent words in the collection [2]. (9) is an increasing function in $c(w, D_i)$ so a higher $c(w, D_i)$ makes (9) less negative. Therefore, the higher the occurrence of w_1 and w_2 in the feedback documents is, the lower is the extent to which w_2 is favored. This is different from the case of RM where the documents lengths controlled how common words are rewarded.

To recap, we have shown that using collection LM-based smoothing at the feedback stage will cause several of the well-known PRF models to penalize discriminative words.

[2] [6] has already pointed this out when using Jelinek-Mercer smoothing

Chapter 5

Additive Smoothing for PRF Models

To solve the problem of favoring highly frequent words in LM-based PRF methods, we propose using additive smoothing in the feedback stage, while keeping the collection LM-based smoothing in the retrieval stage in order to preserve the IDF and document length normalization heuristics. In additive smoothing, the probability of each feedback document is adjusted as follows:

$$P(w|\theta_i) = \frac{c(w, D_i) + \gamma}{|D_i| + \gamma|V_F|}$$

where $\gamma > 0$ is the smoothing parameter that can be considered as a pseudo count, and $|V_F|$ is the size of the vocabulary of the feedback documents. Below we study the behavior of the different PRF models under additive smoothing.

5.1 Divergence Minimization Model

Assuming w_1 and w_2 as in the definition of the IDF effect and using additive smoothing we get:

$$\begin{aligned}
&\log P(w_1|\theta_F) - \log P(w_2|\theta_F) \propto \\
&\sum_{i=1}^{|F|} \Big(\log \frac{c(w_1, D_i) + \gamma}{c(w_2, D_i) + \gamma} - \lambda \log \frac{P(w_1|\theta_C)}{P(w_2|\theta_C)} \Big) = \\
&- |F|\lambda \log \frac{P(w_1|\theta_C)}{P(w_2|\theta_C)} > 0
\end{aligned} \tag{5.1}$$

(10) is unconditionally positive for any choice of γ and λ which means that the DMM supports the IDF effect in this case and consequently favors more discriminative words. The extent to which high IDF words are favored can be controlled by the parameter λ. By using additive smoothing, the DMM is

now performing what it is intended to do: minimize the divergence from the centroid of the feedback documents' LMs and maximize the divergence from the collection language model. When collection-based smoothing was used, the DMM was not performing the intended objective, while on the contrary, it was favoring frequent words in many cases.

5.2 Relevance Model

Assuming w_1 and w_2 as before while using additive smoothing:

$$P(w_1|\theta_F) - P(w_2|\theta_F) = \sum_{i=1}^{|F|} \left(P(\theta_i|q) \frac{c(w_1, D_i) + \gamma - c(w_2, D_i) - \gamma}{|D_i| + \gamma|V_f|} \right) = 0 \tag{5.2}$$

(11) shows that when we use additive smoothing for the RM, the model will not favor common words and will treat all words equally, i.e. irrespective of how they occurred in the collection. Since $P(w_1|\theta_F) = P(w_2|\theta_F)$, strictly speaking, the IDF effect is still not supported as it requires $P(w_2|\theta_F) > P(w_1|\theta_F)$. Although the IDF effect is not supported, the performance is expected to improve since the model no longer favors common terms as was the case with collection-based smoothing.

5.3 Geometric Relevance Model

Assuming w_1 and w_2 as before while using additive smoothing:

$$\log P(w_1|\theta_F) - \log P(w_2|\theta_F) \propto \sum_{i=1}^{|F|} \left(P(\theta_i|q) \log \frac{c(w_1, D_i) + \gamma}{c(w_2, D_i) + \gamma} \right) = 0 \tag{5.3}$$

When additive smoothing is used, the GRM will also treat words irrespective of their frequency of occurrence in the collection, in contrast to the case of collection-based smoothing where common terms are always favored. We should note that this behavior is expected since the RM and GRM are not designed to penalize common words.

Chapter 6

Measuring PRF Method Discrimination

Although the IDF effect is an interesting property to analyze, it cannot provide direct insight on the extent to which the terms output by a certain PRF method are discriminative. Therefore, we need a good empirical measure that can indicate how discriminative the feedback terms are. Previous attempts in the literature have used the average of the IDF of the top terms output by the PRF method to quantify the method's discrimination. For example, [6] used Average_IDF $= \frac{1}{|Q|} \sum_{q \in Q} \sum_{w \in F} \frac{\log_{10} \frac{N}{N_w}}{t}$ where Q is the set of queries, N is the total number of documents in the collection, N_w is the frequency of the word in the collection C, and t is the number of feedback terms considered. However, such approaches do not take into account the probabilities assigned to the terms which might be problematic. For instance, consider a PRF method that consistently outputs many discriminative terms, in addition to a few common terms that get all the probability mass. In such a case, the average of the IDF values of all the top terms will be high, however the method is assigning most of the probability to the common terms so it is practically updating the query with common terms. Thus, using the average of the IDF might not be a good measure. A more suitable measure should assess the extent to which discriminative terms will *change* the query.

We propose a simple measure called the Discrimination Measure (DM) for quantifying the discriminative power of a PRF method. The measure takes into account both the IDF **and** the probabilities of the top terms and captures the idea that the higher the probability assigned to discriminative terms, the higher is the discrimination of the PRF method. Let $Q = \{q_1, q_2, ..., q_m\}$ be a set of queries associated with a probability distribution θ_Q, and $\theta_F(q)$ be the feedback language model output by the PRF method for the query q. Given

a PRF method, we define the Discrimination Measure as:

$$DM = \frac{\mathbb{E}_{\theta_C}[\theta_C]}{\mathbb{E}_{\theta_Q}\mathbb{E}_{\theta_F(q)}[\theta_C]} \approx \frac{\sum_{w \in C} P(w|\theta_C)^2}{\frac{1}{|Q|}\sum_{q \in Q}\sum_{w \in F(q)} P(w|\theta_{F(q)})P(w|\theta_C)}$$

where a uniform prior over the queries is assumed to get the right-hand side. The denominator can be viewed as the expected value of the DF (Document Frequency) of the feedback terms under the PRF method's distribution. The numerator is a normalization factor that makes the measure's values more meaningful. Consider a hypothetical PRF method that consistently outputs a feedback LM whose distribution is the same as that of the collection LM (i.e. $\theta_F = \theta_C$), the DM of this method is equal to 1. If a PRF method puts more probability mass on common terms compared to the hypothetical method, then $DM < 1$. On the other hand, if a PRF method assigns higher probabilities to discriminative terms compared to the hypothetical method, then $DM > 1$. In general, the more probability the PRF method assigns to discriminative terms, the lower is the expected DF of the feedback terms, and consequently the higher is the DM.

Chapter 7

Empirical Evaluation

To validate the results of the axiomatic analysis and gain more insight on the performance of LM-based PRF methods, we ran several experiments to empirically examine the validity of our analytical results and the main hypothesis that additive smoothing performs better than collection-based smoothing for PRF.

7.1 Datasets and Parameter Setting

We used four TREC collections in the experiments: AP (Associated Press 88-89, TREC Disks 1 & 2), WSJ (Wall Street Journal 87-92, TREC Disks 1 & 2), Robust (Robust 2004, TREC Disks 4 & 5 minus Congressional Record), and WT10g (TREC Web Corpus). A summary of the datasets' statistics is shown in Table 7.1. The queries were extracted from the title field of each topic. We carried out the experiments using MeTA toolkit[1] where we preprocessed all the collections using MeTA's default stopword list and performed stemming using the Porter2 English stemming algorithm. The KL divergence retrieval function was used along with Dirichlet prior smoothing with the smoothing coefficient μ set to 1000. The additive smoothing parameter γ was also fixed to 1 in all the experiments, except for the experiment in Section 7.2.3 which involves sweeping this parameter.

In all the experiments, we split each dataset's topics into training and testing subsets as shown in Table 7.1. The training topics are used to learn the model parameters that optimize the MAP (Mean Average Precision), and the testing topics are used to report the MAP which is evaluated for the top 1000 retrieved documents, in addition to the P10 (Precision at 10 documents) and the DM (Discrimination Measure). In the training phase

[1] https://meta-toolkit.org/

we sweep the number of feedback documents $|F|$ between {10, 25, 50, 75, 100}, the number of feedback terms between {10, 25, 50, 75, 100}, and the interpolation coefficient α between {0.1, 0.2,..., 0.9}. Additionally, for the DMM with collection-based smoothing, we sweep the parameter λ in the range {0.1, 0.3, 0.5, 0.7, 0.9}, whereas for the additive smoothing variant of the DMM we sweep the parameter λ in the range {0.01, 0.03, 0.05, 0.07, 0.09} (the DMM requires larger values of λ to suppress the common terms and reach its optimal performance in case of collection-based smoothing).

7.2 Experimental Results

We now present both qualitative and quantitative results from our experiments to examine the analytical results and hypotheses.

7.2.1 Interference of Collection-based Smoothing with IDF Effect

We first examine the empirical behavior of the PRF methods when using collection-based smoothing vs. additive smoothing with specific consideration of the IDF effect. Specifically, we ran multiple queries and manually inspected the top feedback terms extracted by the three PRF methods using both Dirichlet prior smoothing and additive smoothing for different parameters.

In Table 2, we show the top 10 feedback terms for the query "Computer" extracted by the DMM and GRM using the AP collection (i.e. the words are shown before interpolation). Although we could not include the top feedback terms of the RM due to space constraints, it extracted words similar to the other two models. We should also note that the results obtained using this query are very representative of the other queries we examined in terms of the quality of the extracted terms.

When using DMM with Dirichlet prior smoothing and $\lambda = 0.1$ (Table 2 (a)), most of the top feedback terms are very common and not informative, and this is a direct consequence of the lack of support for the IDF effect as demonstrated in Theorem 1. Changing to $\lambda = 0.95$ (Table 2 (b)) while keeping the Dirichlet prior smoothing actually extracted high quality terms,

and this is explained by the support for the IDF effect since $\lambda > \frac{|F|-1}{|F|}$ ($|F|$ is set to 10 for the results in Table 2). However, almost all the probability mass is being assigned to the original query word, and this validates the result we got in Theorem 2 where we showed that the DMM will assign the majority of the probability to one word when λ becomes close to 1. When we switched to additive smoothing (Table 2 (c)), most of the terms became discriminative and the original query word got only 21% of the total probability mass.

Similarly, the GRM had very common terms when using Dirichlet prior smoothing (Table 2 (d)). When we switched to additive smoothing (Table 2 (e)), more discriminative terms got introduced. This confirms the results of the axiomatic analysis where we showed that the GRM and RM will no longer favor common terms when additive smoothing is used.

7.2.2 Additive vs. Collection-based Smoothing

Next, we want to examine the expected empirical benefit of additive smoothing over collection-based smoothing. Thus, we compared the MAP, precision at 10 documents, and the Discrimination Measure of the three methods for both Dirichlet prior smoothing and additive smoothing over the four collections. The results are reported in Table 7.3. Note that we optimized the parameters ($|F|$, number of feedback terms, and α) only for collection-based smoothing and used the learned parameters for additive smoothing, which might give collection-based smoothing an advantage in some of the reported values.

As shown in Table 7.3, additive smoothing improved the MAP and precision at 10 documents for all the methods and datasets. The improvement in MAP is statistically significant for almost all the values which confirms our hypothesis that additive smoothing can enhance the performance of PRF methods by preventing them from rewarding common terms. The average improvement in MAP for the DMM, RM, and GRM over all the datasets are 8.7%, 4.5%, and 8.6%, respectively. The average improvement in precision at 10 for the DMM, RM, and GRM over all the datasets are 6.4%, 5.1%, and 5.7%, respectively. The DMM and GRM seem to benefit more from additive smoothing compared to RM, and this can be attributed to the reliance of the RM on the arithmetic mean which is less sensitive to the smoothing method

used. Another observation is that the difference in performance between the three methods becomes significantly smaller when using additive smoothing.

The Discrimination Measure is low for all the collection-based smoothing variants and has values below 1 for three out of the four datasets, meaning that the traditional PRF methods were assigning more probability mass to the common words compared to the probability assignments of the collection LM θ_C. After switching to additive smoothing, all the methods got a several-fold increase in the DM implying that the expected value of the document frequency of the extracted terms got a several-fold decrease. The results show that the DMM generally has the highest DM when using additive smoothing, and this can be explained by the fact that the DMM supports the IDF effect, whereas the other two models do not discriminate based on the document frequency of the terms (the Robust collection is an exception where GRM had the highest DM and this due to the small λ (=0.01) that optimized the MAP in this case).

We also swept the number of feedback documents and the number of feedback terms over a range of values to study the robustness of the models under each smoothing scheme. The results on the AP data set are shown in Figure 1 (the other three data sets have shown similar patterns). The sensitivity of the different PRF methods under additive smoothing is very similar to that under collection-based smoothing, implying that additive smoothing does not affect the robustness of the different models. The performance of the models with additive smoothing is consistently higher than the collection-based smoothing counterparts for any value of $|F|$ and any number of feedback terms. We should also note that when the number of feedback documents increases, the RM and GRM appear to be more stable than the DMM, and this can be attributed to the fact that the DMM treats all documents equally whereas the RM and GRM assign each document a weight equal to its probability of relevance (i.e. lower ranked documents would have less effect on the extracted feedback terms).

7.2.3 Additive Smoothing Parameter

The results above show a clear advantage of additive smoothing over collection-based smoothing for PRF. We now turn to the question about how to opti-

mize additive smoothing for PRF, particularly how to optimize its parameter γ. To this end, we swept the additive smoothing parameter γ by decades between 10^{-5} and 1 to study its effect on performance. As shown in Figure 7.2, the RM maintains very stable performance as the additive smoothing parameter changes. In fact, we tried running the RM without any smoothing (i.e. $\gamma = 0$), and its performance remained the same (not using smoothing at the feedback stage will stop rewarding the common terms when compared to the collection-based smoothing, thus giving the same effect as additive smoothing). The MAP and precision at 10 of the DMM and GRM increase as γ increases (with one minor outlier), and the performance is maximized by $\gamma = 1$ which corresponds to the conventional add-one smoothing. Although the graphs shown were generated using AP collection, the same results discussed in this section are also valid for the three other collections.

7.3 Summary of Main Findings

Our experimental results have confirmed the predictions of our axiomatic analysis of the smoothing methods for LM-based PRF models, showing that the traditionally used collection-based smoothing indeed forces LM-based PRF models to reward common words, and additive smoothing solves the issue by focusing the probability mass on more discriminative terms thus increasing the retrieval performance significantly.

Our results also show that the additive smoothing parameter $\gamma = 1$ maximized the performance of the DMM and GRM over all the collections. The RM's performance is not affected by the smoothing parameter, and running it without any smoothing (i.e. $\gamma = 0$) gives the same performance as additive smoothing. Overall, these results, along with our theoretical analysis, suggest that additive smoothing, instead of collection-based smoothing, should be used in all the LM-based PRF methods.

7.4 Tables and Figures

Collection	AP	WSJ	Robust	WT10g
#Docs	164k	173k	528k	1692k
#Queries (w. qrels)	99	100	249	100
Training Queries	51-100	51-100	301-450	451-500
Testing Queries	101-150	101-150	601-700	501-550

Table 7.1: Datasets' Statistics

w	$P(w\|\theta_F)$
comput	0.2213
time	0.0296
state	0.0293
new	0.0280
nation	0.0253
percent	0.0250
two	0.0229
say	0.0228
system	0.0228
report	0.0223

(a) DMM - Dirichlet Smoothing $\lambda = 0.1$

w	$P(w\|\theta_F)$
comput	0.9999
arpanet	3.8e-07
virus	1.6e-15
bug	6.2e-19
hacker	3.0e-20
network	8.3e-21
adv-research	2.2e-21
mellon	3.9e-22
thecomput	1.6e-22
data	9.8e-23

(b) DMM - Dirichlet Smoothing $\lambda = 0.95$

w	$P(w\|\theta_F)$
comput	0.2099
virus	0.0342
system	0.0294
univers	0.0265
network	0.0258
program	0.0239
research	0.0231
time	0.0227
arpanet	0.0187
data	0.0182

(c) DMM - **Additive Smoothing** $\lambda = 0.03$

w	$P(w\|\theta_F)$
comput	0.1561
state	0.0326
time	0.0310
new	0.0309
percent	0.0279
nation	0.0279
say	0.0254
two	0.0254
report	0.0247
year	0.0242

(d) GRM - Dirichlet Smoothing

w	$P(w\|\theta_F)$
comput	0.1962
virus	0.0334
system	0.0302
univers	0.0282
network	0.0254
program	0.0251
research	0.0238
time	0.0235
nation	0.0194
center	0.0182

(e) GRM - **Additive Smoothing**

Table 7.2: Top 10 Feedback Terms for the Query "Computer" for different smoothing strategies and parameters. Tables (a), (b), and (d) refer to the traditional Collection LM-based smoothing method used in the literature. Tables (c) and (e) show how using additive smoothing can lead to the selection of more discriminative terms.

Dataset	Measure	DMM		RM		GRM	
		Collection	Additive	Collection	Additive	Collection	Additive
AP	MAP	0.261	0.301*	0.291	0.303*	0.258	0.286*
	P10	0.4	0.44	0.412	0.434	0.416	0.426
	DM	0.524	1.94	0.648	1.69	0.527	1.75
WSJ	MAP	0.254	0.268*	0.253	0.267*	0.239	0.258*
	P10	0.426	0.458	0.46	0.494	0.42	0.46
	DM	0.75	2.67	0.84	2.36	0.686	2.45
Robust	MAP	0.287	0.299*	0.312	0.324*	0.298	0.322*
	P10	0.44	0.442	0.447	0.466	0.445	0.47
	DM	1.72	9.47	1.89	6.51	1.54	11.1
WT10g	MAP	0.196	0.215	0.205	0.214*	0.198	0.213*
	P10	0.324	0.349	0.339	0.351	0.337	0.355
	DM	0.812	10.6	1.16	6.56	0.808	8.57

Table 7.3: Comparison of retrieval performance between collection-based and additive smoothing for the DMM, RM, and GRM. All additive smoothing MAP values with an asterisk are statistically significant from their collection-based smoothing counterparts based on a paired two-tailed t-test with significance level = 0.05.

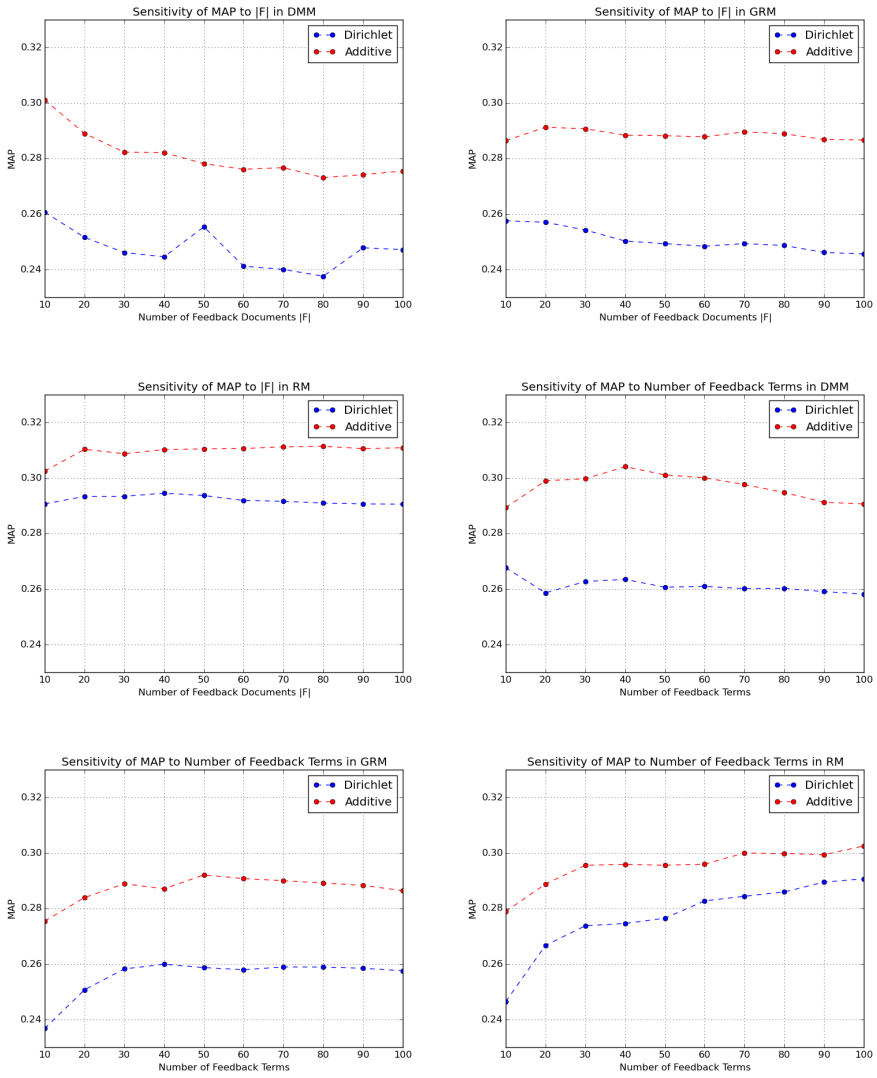

Figure 7.1: Sensitivity of the MAP to the number of feedback documents $|F|$ and the number of feedback terms for the DMM, GRM, and RM using Dirichlet prior Smoothing and Additive Smoothing. Additive smoothing significantly outperforms Dirichlet prior smoothing in all the models and for all values of the parameters.

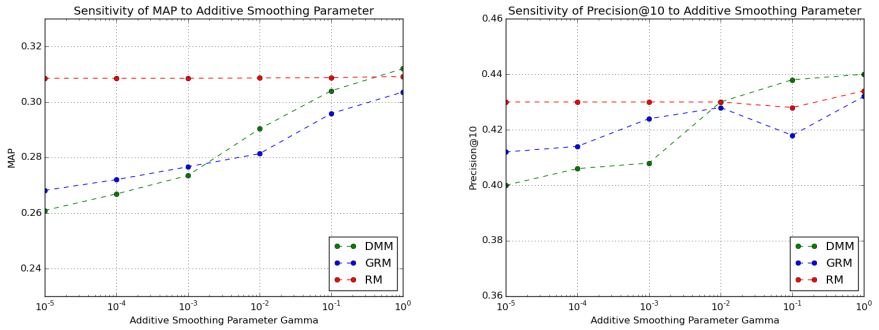

Figure 7.2: Sensitivity of the MAP and precision at 10 to the additive smoothing parameter γ.

Chapter 8

Conclusion and Future Work

Language model-based pseudo relevance feedback methods have proven effective in enhancing retrieval and can be applied to any retrieval system without requiring extra effort from users. Smoothing of language models for the feedback documents is necessary in these methods to address the issue of data sparsity and has been achieved using a collection language model-based smoothing strategy in virtually all the work so far. Although collection-based smoothing has been shown to be necessary and effective in the retrieval stage, we have axiomatically shown that collection-based smoothing in the feedback stage is non-optimal and inherently interferes with the desired preference for discriminative terms for feedback since it forces pseudo-relevance feedback methods to favor very frequent and non-informative words. We proposed replacing collection-based smoothing at the feedback stage with additive smoothing, which is analytically proven to ensure that the learned feedback model favors discriminative terms and empirically shown to achieve better retrieval accuracy when compared to collection-based smoothing.

The experiments show that the three models perform similarly when using additive smoothing, although the DMM supports the IDF effect whereas the RM and GRM treat feedback terms irrespective of their occurrence in the collection. This motivates an interesting question about the significance of the IDF effect. It has been established that by not rewarding common terms, the PRF method can attain good performance, however, does favoring discriminative terms through supporting the IDF lead to significant performance improvement? Our analysis also reveals a connection between a parameter in the divergence minimization model and the number of feedback documents; this is an interesting direction worth further exploration in the future. Yet another question motivated by our study is how to analytically bound the parameter of additive smoothing when used for PRF. The empirical results show that setting $\gamma = 1$ seems to work well in the experiments we have done,

but could it be possible to derive analytical bounds for this parameter?

References

[1] H. Hazimeh and C. Zhai, "Axiomatic analysis of smoothing methods in language models for pseudo-relevance feedback," in *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, ser. ICTIR '15. New York, NY, USA: ACM, 2015. [Online]. Available: http://doi.acm.org/10.1145/2808194.2809471 pp. 141–150.

[2] J. Xu and W. B. Croft, "Query expansion using local and global document analysis," in *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 1996, pp. 4–11.

[3] V. Lavrenko and W. B. Croft, "Relevance based language models," in *Proceedings of ACM SIGIR 2001*, ser. SIGIR '01. New York, NY, USA: ACM, 2001. [Online]. Available: http://doi.acm.org/10.1145/383952.383972 pp. 120–127.

[4] C. Zhai and J. Lafferty, "Model-based feedback in the language modeling approach to information retrieval," in *Proceedings of ACM CIKM 2001*, ser. CIKM '01. New York, NY, USA: ACM, 2001. [Online]. Available: http://doi.acm.org/10.1145/502585.502654 pp. 403–410.

[5] Y. Lv and C. Zhai, "A comparative study of methods for estimating query language models with pseudo feedback," in *Proceedings of ACM CIKM 2009*, ser. CIKM '09. New York, NY, USA: ACM, 2009. [Online]. Available: http://doi.acm.org/10.1145/1645953.1646259 pp. 1895–1898.

[6] S. Clinchant and E. Gaussier, "A theoretical analysis of pseudo-relevance feedback models," in *Proceedings of the 2013 Conference on the Theory of Information Retrieval*, ser. ICTIR '13. New York, NY, USA: ACM, 2013. [Online]. Available: http://doi.acm.org/10.1145/2499178.2499179 pp. 6:6–6:13.

[7] Cheng, "Statistical language models for information retrieval a critical review," *Found. Trends Inf. Retr.*, vol. 2, no. 3, pp. 137–213, Mar. 2008. [Online]. Available: http://dx.doi.org/10.1561/1500000008

[8] J. Seo and W. B. Croft, "Geometric representations for multiple documents," in *Proceedings of ACM SIGIR 2010*, ser. SIGIR '10. New York, NY, USA: ACM, 2010. [Online]. Available: http://doi.acm.org/10.1145/1835449.1835493 pp. 251–258.

[9] J. Rocchio, "Relevance feedback in information retrieval," *SMART Retrieval System Experiments in Automatic Document Processing*, 1971.

[10] S. E. Robertson and K. S. Jones, "Relevance weighting of search terms," *Journal of the American Society for Information science*, vol. 27, no. 3, pp. 129–146, 1976.

[11] C. J. V. Rijsbergen, *Information Retrieval*, 2nd ed. Newton, MA, USA: Butterworth-Heinemann, 1979.

[12] H. Fang and C. Zhai, "Web search relevance feedback," in *Encyclopedia of Database Systems*, 2009, pp. 3493–3497.

[13] G. Salton and C. Buckley, "Improving retrieval performance by relevance feedback," *Journal of the American Society for Information Science*, vol. 41, no. 4, pp. 288–297, 1990.

[14] J. Allan, "Relevance feedback with too much data," in *Proceedings of ACM SIGIR 1995*, 1995, pp. 337–343.

[15] I. Ruthven and M. Lalmas, "A survey on the use of relevance feedback for information access systems," *Knowl. Eng. Rev.*, vol. 18, no. 2, pp. 95–145, June 2003.

[16] C. Buckley, G. Salton, J. Allan, and A. Singhal, "Automatic query expansion using smart: Trec 3," *NIST special publication sp*, pp. 69–69, 1995.

[17] Y. Lv and C. Zhai, "Revisiting the divergence minimization feedback model," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, ser. CIKM '14. New York, NY, USA: ACM, 2014. [Online]. Available: http://doi.acm.org/10.1145/2661829.2661900 pp. 1863–1866.

[18] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade, "Umass at trec 2004: Novelty and hard," DTIC Document, Tech. Rep., 2004.