Approximating True Relevance Model in Relevance Feedback

ZHANG, PENG

A thesis submitted in partial fulfilment
of the requirements of
Robert Gordon University
for the degree of Doctor of Philosophy

January 2013

Abstract

Relevance is an essential concept in information retrieval (IR) and relevance estimation is a fundamental IR task. It involves not only document relevance estimation, but also estimation of user's information need. Relevance-based language model aims to estimate a relevance model (i.e., a relevant query term distribution) from relevance feedback documents. The true relevance model should be generated from truly relevant documents. The ideal estimation of the true relevance model is expected to be not only effective in terms of mean retrieval performance (e.g., Mean Average Precision) over all the queries, but also stable in the sense that the performance is stable across different individual queries. In practice, however, in approximating/estimating the true relevance model, the improvement of retrieval effectiveness often sacrifices the retrieval stability, and vice versa.

In this thesis, we propose to explore and analyze such effectiveness-stability tradeoff from a new perspective, i.e., the bias-variance tradeoff that is a fundamental theory in statistical estimation. We first formulate the bias, variance and the trade-off between them for retrieval performance as well as for query model estimation. We then analytically and empirically study a number of factors (e.g., query model complexity, query model combination, document weight smoothness and irrelevant documents removal) that can affect the bias and variance. Our study shows that the proposed bias-variance trade-off analysis can serve as an analytical framework for query model estimation. We then investigate in depth on two particular key factors: document weight smoothness and removal of irrelevant documents, in query model estimation, by proposing novel methods for document weight smoothing and irrelevance distribution separation, respectively. Systematic experimental evaluation on TREC collections shows that the proposed methods can improve both retrieval effectiveness and retrieval stability of query model estimation. In addition to the above main contributions, we also carry out initial exploration on two further directions: the formulation of bias-variance in personalization and looking at the query model estimation via a novel theoretical angle (i.e., Quantum theory) that has partially inspired our research.

Keywords: Relevance Feedback, True Relevance Model, Bias-Variance Analysis, Document Weight Smoothing, Distribution Separation Method, Personalization, Quantum

Acknowledgements

I am greatly indebted to my supervisors, Prof. Dawei Song and Prof. John McCall, for their valuable guidance, fruitful discussions and tremendous support throughout my PhD. I am also grateful to Prof. Yuexian Hou at Tianjin University, China for his thoughtful ideas and discussions. Special thanks to Dr. Jun Wang, Mr. Xiaozhao Zhao, Mr. Tingxu Yan, Mr. Leszek Kaliciak, Mr. Lei Wang and Mr. Ulises Cervino Beresi for their collaborations or discussions with my research.

Many thanks to Robert Gordon University for granting me full studentship. Gratitude also goes my colleagues, Ibrahim, Jean-Claude, Sandy, Thiery, David, Malcolm, Guofu, Peter, Sadiq, Amina, Richard, Noura, Micheal, Claire, Ben, Nuka, Olivier, and Yanghui, for making CTC a convenient research environment.

I am especially indebted to my parents Mr Jiayu Zhang and Ms Guizhen Shi for their love and unconditional support. I am also thankful to my sister Mrs Lan Zhang. Finally, thanks to my girlfriend for her love, support, encouragement, patience and understanding.

Declarations

I declare that all of the work in this thesis was conducted by the author except where otherwise indicated, and this thesis has not been submitted at any other university.

Parts of the work outlined in this thesis have appeared in the following publications.

Chapter 4

- Peng Zhang, Dawei Song, Jun Wang, Xiaozhao Zhao, Yuexian Hou, On Modeling Rank-Independent Risk in Estimating Probability of Relevance. Accepted by *the 7th Asia Information Retrieval Societies Conference (***AIRS 2011***)* (acceptance rate=24%), 18-20 December 2010, Dubai, United Arab Emirates.

- Peng Zhang, Dawei Song, Xiaozhao Zhao, Yuexian Hou, A Study of Document Weight Smoothness in Pseudo Relevance Feedback. In *the 6th Asia Information Retrieval Societies Conference (***AIRS 2010***)* (acceptance rate=22%), LNCS 6458, pp. 521-538. 1-3 December 2010, Taipei

Chapter 5

- Peng Zhang, Yuexian Hou and Dawei Song. Approximating True Relevance Distribution from a Mixture Model Based on Irrelevance Data. In: *Proceedings of The 32nd Annual ACM SIGIR Conference (***SIGIR 2009***) (acceptance rate=16.5%) .* , pp. 107-114, 19-23 July 2009, Boston, USA

Chapter 6

- Peng Zhang, Dawei Song, Xiaozhao Zhao and Yuexian Hou (2011). Investigating Query-Drift Problem from a Novel Perspective of Photon Polarization. *The 3rd International Conference on the Theory of Information Retrieval (***ICTIR 2011***)*, LNCS, pp. 332-336. 12-14 September 2011, Bertinoro, Italy.

- Xiaozhao Zhao, Peng Zhang, Dawei Song, Yuexian Hou (2011). A Novel Re-Ranking Approach Inspired by Quantum Measurement. **Best Poster Award** *in The 33rd European Conference on Information Retrieval (***ECIR'2011***)*, LNCS 6661, pp. 721-724. 19-21 April 2011, Dublin.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Information retrieval (IR) is a scientific field that studies how to retrieve information objects (e.g., documents) that are relevant to the user's information need. Relevance is an essential IR concept and relevance estimation is a fundamental IR task. This task involves estimating not only the relevance of documents, but also the relevance of query terms. In this thesis, our main focus is on the query model estimation and our goal is to approximate the truly relevant query model that represents the underlying information need.

This goal is challenging due to the fact that in practice the original query terms input by users are insufficient to represent their underlying information needs. For example, if a user only types "Michael Jordan" in the search box, the underlying information need could be about the famous basketball player Michael Jordan, or about the UC Berkeley's Professor Michael Jordan. It is hard to tell what the user's information need is solely based on the original query "Michael Jordan".

Relevance feedback, which can be explicit, implicit or pseudo, is a post-query process to estimate the information need and enhance IR performance by creating a revised query model (van Rijsbergen 1979). Each kind of user relevance feedback has its own advantages and disadvantages. *Explicit relevance feedback* has been shown useful to improve the IR performance (Buckley & Salton 1995). For example, when the user explicitly indicates that the documents about the basketball player Michael Jordan are relevant, it is much easier to identify his/her information need. However, users in general may be constrained and reluctant to provide explicit relevance feedback on a relatively large number of top ranked documents (Dumais, Joachims, Bharat & Weigend 2003, Jansen, Spink & Saracevic 2000, Henzinger, Motwani & Silverstein 2002). *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 and viewing time, etc (White, Ruthven & Jose 2005). It avoids the need of user's explicit judgments, but in the expense of the accuracy of the implicit relevance

judgements inferred from user interactions. Indeed, it is still largely under exploration on what interactions should be taken into account and how good they are as relevance factors. In addition, academic researchers often have relatively limited access to these user interaction data for implicit feedback. *Pseudo relevance feedback* simply assumes a number of top ranked documents 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.

1.2 Challenges in Approximating True Relevance Model

Relevance-based language model (Relevance Model or RM) (Lavrenko & Croft 2001) can estimate a relevant query model (i.e., a relevant term distribution) from relevance feedback documents. Assuming the information need can be represented by the truly relevant documents, the true relevance model should be generated from the truly relevant documents (see also Section 2.2.3 in Chapter 2 for details of the true relevance model).

However, the relevance feedback documents from which the relevance model is generated is often a mixture of relevant and irrelevant documents, especially in pseudo relevance feedback and implicit relevance feedback. In addition, different queries may have different numbers of relevant/irrelevant documents in the whole feedback document set. Even though we may have some explicit feedback data for some queries, it is usually impossible to obtain all explicit feedback for all queries in practice. Thus, there is a high level of uncertainty in the sense that the relevant information in relevance feedback varies for different queries. Such uncertainty poses a series of challenges to approximate the true relevance model.

- *Effectiveness-Stability Tradeoff.* To approximate the true relevance model is essentially a query model estimation problem. The ideal estimation is expected to be not only *effective* in terms of mean retrieval performance (e.g., mean average precision) over all queries, but also *stable* in the sense that the performance is stable across individual queries. However, in practice, there is often a tradeoff between the effectiveness and stability of the query model estimation. For example, in pseudo relevance feedback (PRF), the expanded query model (which is a revised query model) is generally more effective yet often less stable than the original query model (Amati, Carpineto, Romano & Bordoni 2004, Collins-Thompson 2009b). This stability problem is often rooted on the inclusion of irrelevant documents in the feedback documents used for the query model estimation. It is important to solve or alleviate such tradeoff, in order to improve both retrieval effectiveness and robustness. Please also see Section 1.4.1 for examples of the effectiveness and robustness tradeoff.

- *Theory on Query Model Estimation.* To understand the origin of the problems

(e.g., effectiveness-stability tradeoff) in the query model estimation, it is important and necessary to resort to the statistical estimation theory, which provides powerful principles and formulations to understand the estimation quality. It is, however, challenging to find the right pieces of the estimation theory and integrate them into IR theory. The resulting new IR theory should be able to not only explain the effectiveness-stability tradeoff problem in the query model estimation, but also guide us to build formal methods to improve the retrieval effectiveness and stability.

- *Theory on Document Relevance Estimation.* The document relevance estimation plays an important role in the query model estimation. For example, the weights of feedback documents used to estimate the query model are often the normalized relevance scores of documents in the initial ranking. There are many existing theories that have been proposed to estimate the document relevance effectively. It is important to explore new theoretical aspects about the document relevance estimation, and the effects on the tradeoff between retrieval effectiveness and retrieval stability.

- *Generic Methods and Applications.* In addition to the above theoretical aspects, one essential task is to build models or formulations and then apply them successfully in the query model estimation. Systematic evaluations should be constructed by considering many aspects, e.g., retrieval effectiveness, retrieval stability and estimation quality, etc. Moreover, it is important that the proposed formulations or models can be potentially applied to other IR tasks or even other fields, rather than only in query model estimation task.

1.3 Research Aims and Objectives

Our research aim is to tackle the above challenges in approximating the true relevance model in relevance feedback. The research objectives are:

- To build a unified theoretical framework based on statistical estimation theory, and analyze factors that can affect the retrieval effectiveness and stability in query model estimation.

- To explore new theoretical aspects for the document relevance estimation and connect the new theory with the above unified framework.

- Based on the above framework or theory, to develop corresponding methods that can improve the estimation quality, and alleviate the retrieval effectiveness-stability tradeoff.

1.4 Contributions

1.4.1 A Novel Bias-Variance Analysis Framework

We propose to study the aforementioned retrieval effectiveness-stability *tradeoff* problem from a novel theoretical perspective, i.e., bias-variance *tradeoff*. The bias-variance tradeoff is fundamental in the estimation theory (Zucchini, Berzel & Nenadic 2005, Bishop 2006). In this thesis, we propose to formulate the *performance bias and variance* which are related to the retrieval effectiveness and stability, respectively. We can look into the problem of *improving* the retrieval effectiveness and stability from the perspective of *reducing* performance bias and variance, respectively.

We first provide some examples of variance to explain the concept of the retrieval stability, and also discuss the difference between the proposed variance measurement in this thesis and the existing measurement in the literature.

Examples of Variance

Suppose that there are two queries q_1 and q_2, and for each query we use the average precision (AP) to measure/observe the retrieval performance of a query model estimation method (or a IR system in general). Assume that we have two estimation methods A and B, where A and B can correspond to the original query representation and the expanded query representation, respectively.

In Table 1.1, for q_1 and q_2, the AP of method A can be 0.3 and 0.1, respectively, and the AP of B can be 0.6 and 0.008, respectively. It turns out that the mean average precision (MAP) over all queries for methods A and B are (0.3+0.1)/2 and (0.6+0.08)/2, which are 0.2 and 0.34, respectively. In this thesis, the retrieval effectiveness is measured by the MAP. Therefore, A is less effective than B.

We computes the variance of retrieval performance across all concerned queries (denoted as VAP). Specifically, for A, VAP is computed by:

$$\text{VAP}_A = \frac{1}{2}[(0.3 - 0.2)^2 + (0.1 - 0.2)^2] = 0.01$$

which calculates the derivation of AP (i.e., 0.3 and 0.1 for q_1 and q_2, respectively) from its mean: MAP (i.e., 0.2). Similarly, we can compute VAP for B as

$$\text{VAP}_B = \frac{1}{2}[(0.6 - 0.34)^2 + (0.08 - 0.34)^2] = 0.0676$$

It turns out the VAP_B is greater than VAP_A. In my thesis the retrieval stability is measured by VAP, and the smaller VAP generally means the better retrieval stability. In the above example, we can say that A is more stable than B. Recall that A and B corresponds to the original query model and expanded query model, respectively.

Table 1.1: Examples for Different Metrics

Method	A		B		T	
	q_1	q_2	q_1	q_2	q_1	q_2
AP	0.3	0.1	0.6	0.08	0.7	0.2
MAP	0.2		0.34		0.45	
VAP	0.01		0.0646		0.0625	
$Bias$	0.25		0.11		0	
Var	0.01		0.0646		0.0625	
$Bias^2 + Var$	0.0725		0.0797		0.0625	
Robust Index	0		0		1	
$< Init$	0		0.5		0	

In the literature, Collins-Thompson (2009a) addressed the retrieval stability across queries using the general concept of variance, in the sense that the query expansion can not always improve the retrieval performance of the original query model. However, they did not compute the variance (i.e., VAP) of retrieval performance across queries. In (Collins-Thompson 2009a), the variance is considered as a risk, and the risk measurement is $R - Loss$ which is quite different from VAP. Specifically, $R - Loss$ in (Collins-Thompson 2009a, Collins-Thompson 2009b) calculates the *average* net loss of relevant documents (due to failure), which is the number of relevant documents lost in the top 1000 retrieved documents by the query expansion.

Examples of Bias-Variance

Now, we show that how to integrate MAP and VAP into a single framework, i.e., the bias-variance framework. Let us first assume that we have a query model estimation method T which can have an upper-bound performance for every query. We then assume that the target AP (i.e., the upper-bound AP) is 0.7 and 0.2 for queries q_1 and q_2, respectively [1]. Recall that for method A, AP is 0.3 and 0.1 for queries q_1 and q_2, respectively.

Then, for A, the bias is

$$Bias_A = \frac{1}{2}[(0.7 - 0.3) + (0.2 - 0.1)] = 0.25$$

which calculates the average difference between the target AP and the actual AP of A over all queries. The above bias also equals

$$Bias_A = \frac{1}{2}(0.7 + 0.2) - \frac{1}{2}(0.3 + 0.1) = 0.45 - 0.2 = 0.25$$

[1]The method T is not necessarily the ideal estimation of the true relevance model which has the maximum AP (i.e., 1) for every query. Please also see the detailed formulation and discussion of the estimation method T in Section 2.2.3 of Chapter 2.

where 0.45 is the target MAP and 0.2 is the MAP of A. Therefore, the bias of A can be the target MAP (i.e., 0.45) minus the MAP (i.e., 0.2) of A. It turns out that the smaller bias indicates the better retrieval effectiveness. For the method B, we can compute its corresponding bias as 0.11 (see Table 1.1). Then, it turns out that A is less effective than B.

We use VAP as the variance (also denoted as Var) in bias-variance formulation. Recall that the smaller the VAP is, the better retrieval stability will be reflected. Then, it shows that A is more stable than B. Recall that A is less effective than B. It turns out an effectiveness-robustness tradeoff, corresponding to bias-variance tradeoff.

In Chapter 3, we will give a more general definition of the performance bias-variance in Section 3.1.2. We will also define the additional performance bias-variance directly based on the difference between the actual performance and the performance target in Section 3.1.3, as well as the examples of additional bias-variance in Section 3.1.4. we investigate different bias-variance decomposition in Section 3.1.5 and Section 3.1.6. Furthermore, we summarize the difference between different performance bias-variance decompositions in Section 3.1.7.

A New Methodology to Analyze Query Model Estimation

The bias-variance framework not only provides novel evaluation metrics (e.g., bias or variance), but also offers a principled methodology to analyze the influence of the query model complexity, query model combination and available relevance/irrelevance judgements on the retrieval performance. For instance, as shown in Table 1.1, the original query model (i.e., A) has bigger bias but smaller variance than the expanded query model (i.e., B), yielding a bias-variance tradeoff. This can be explained by one principle of bias-variance tradeoff, which states that the simple method can have bigger bias but smaller variance (Zucchini et al. 2005, Bishop 2006). The expanded query model is more complex than the original query model, due to the fact that it has more parameters (e.g., the number of expanded query terms) and has more assumptions (e.g., top-ranked documents can be relevant). On the other hand, generally speaking, the model combination and more training data can be helpful to reduce both bias and variance simultaneously. Therefore, in the query modeling problem, the query model combination and the more available relevance judgements can reduce the performance bias and variance simultaneously, which means that both retrieval effectiveness and stability can be improved.

In addition to the performance bias-variance, we also formulate the *estimation bias and variance* of an estimated query model. The estimation bias and variance *directly* compute how closely an estimated query model can approach the true one. Specifically, the estimation bias represents the expected estimation error over all queries, while the estimation variance is the variance of estimation error across different individual queries. The sum of bias and variance can yield the total estimation error which can directly

indicate the total estimation quality. To our knowledge, this is the first time to investigate the estimation quality using estimation bias-variance. This study based on the estimation bias-variance can give finer-grained insights on the estimated query model itself.

In Chapter 3, based on the bias-variance tradeoff, we systematically analyze several estimated query models through investigating a number of key *factors* (i.e., query model complexity, query model combination, document weight smoothness and irrelevant documents removal) that can affect query model estimation. We then propose a set of hypotheses with respect to those factors on bias-variance tradeoff and on reducing both of them simultaneously. A series of experiments based on TREC datasets have been conducted to test the hypotheses. Experimental results on both performance bias-variance and estimation bias-variance generally verify the hypotheses. It demonstrates that the proposed bias-variance formulation can provide valuable theoretical insights on the tradeoff between retrieval effectiveness and stability and explain whether the two retrieval criteria can be improved simultaneously.

Guided by the insights obtained from the above experiments and analysis, we further proposed two lines of methods to improve the retrieval effectiveness and/or stability. Each direction corresponds one factor that can influence the query model estimation, as described in Sections 1.4.2 and 1.4.3.

Comparison with the Mean-Variance Analysis

The proposed bias-variance analysis is different from the existing mean-variance analysis in document ranking (Wang 2009, Wang & Zhu 2009, Zhu, Wang, Cox & Taylor 2009). In mean-variance analysis, it is argued that document ranking should not only provide the point estimation (i.e., the mean) of the document relevance estimate (i.e., the relevance score), but also should consider the uncertainty (i.e., the variance) associated with the relevance score. It turns out that the mean and variance in (Wang 2009, Zhu et al. 2009)) are associated to the document relevance score, while the bias-variance is associated with the retrieval performance. In addition, the mean-variance analysis was conducted for the document ranking, while our bias-variance analysis is to analyze different factors in query model estimation. Moreover, the mean-variance analysis does not involve the analysis about the influence of model complexity, model combination and available relevance judgements on the query modeling or the formal model design.

In addition, another important difference between mean-variance and bias-variance is that the latter has the formulation of bias-variance decomposition of the expected squared error (see Sections 3.1.5 and 3.1.6). In the mean-variance study (Wang & Collins-Thompson 2011, Collins-Thompson 2009a, Collins-Thompson 2009b), only variance is representing the risk/error. On the other hand, in our bias-variance analysis, both bias and variance are decomposed from an error, and solely bias or variance is just one part of the error. The total error summed by the performance bias and variance can actually form a

new robustness metric, as described next.

A New Robustness Metric

Now let us introduce a new robustness metric, which is $Bias^2 + Var$. In our opinion, retrieval robustness is a criterion that combines retrieval effectiveness and stability. Both effectiveness and stability are important in evaluating the robustness of an estimation method. Considering only one criterion (effectiveness or stability) is not sufficient. Thus, the summed quantity of bias and variance, which takes into account both retrieval effectiveness and stability, can be considered as a metric for the retrieval robustness. The smaller the $Bias^2 + Var$ is, the better the robustness will be reflected. The bias-variance decomposition can naturally formulate the effectiveness-stability decomposition of retrieval robustness.

In the literature review (see Section 2.3.3), we provide a comparison between the proposed robustness metric (i.e., $Bias^2+Var$) and the existing robustness metrics. It turns out that the existing robustness metrics are different from ours in terms of formulations and observations. For example, in Table 1.1, the Robustness Index (RI) (Zighelnic & Kurland 2008, Collins-Thompson 2009b) can not distinguish the robustness between these two methods A and B. On the other hand, $Bias^2 + Var$ is able to distinguish the retrieval robustness between them, suggesting that A is more robust than B. In addition, another robustness metric (saying $<_{Init}$ in (Zighelnic & Kurland 2008)) always regard the original query model as one of the most robust models. On the other hand, our robustness metric usually does not regard the original query as the most robust one. For example, based on $Bias^2 + Var$, the method T (which has the upper-bound performance per query) is more robust than the method A (corresponding to the original query model).

In addition, more importantly, regarding novelty, our robustness metric based on bias plus variance can have different theoretical properties. First, our metric provides a decomposition of retrieval robustness into retrieval effectiveness and retrieval stability. Second, it can be studied via the principles of the bias-variance analysis. For instance, the model combination and available relevance judgements can be helpful to reduce both bias and variance simultaneously, yielding the smaller $Bias^2 + Var$ which represents the better robustness.

1.4.2 A Novel Document Weight Smoothing Method

In relevance feedback methods, the document weight is used to indicate the importance of the corresponding document in estimating the query model. We propose a novel method to smooth the document weights. Our bias-variance analysis and experimental results in Chapter 3 show that the smoothness of document weights is an important factor can influence the retrieval performance and the estimation quality of the query models.

The proposed smoothing method can improve the smoothness of truly relevant feedback documents. The truly relevant documents (as judged in the TREC ground truth) often have the same relevance judgements, but they may have quite different document weights. Therefore, the smoothness of the weights among truly relevant documents is quite important. This importance is also supported by our experiments. In addition, smoothing document weights can alleviate the negative impact of irrelevant documents being mis-ranked highly on the relevance feedback. Such smoothing can also broaden the topic coverage of query expansion and prevent the query drifting towards some specific topic represented by the mis-ranked top documents. Moreover, the proposed method can preserve the original rank. This property is related to a new theoretical aspect (i.e., rank-independent risk) of document relevance estimation described next.

We propose to study the rank-independent risk in the document relevance estimation. Recall that the document weight is closely related to the document relevance estimation since the document weight is computed by the normalized relevance score of document. Our observation is that although a precise estimation for the probability of relevance of document can guarantee an optimal ranking (Robertson 1977, Robertson & Zaragoza 2009), an optimal (or even ideal) ranking does not always guarantee that the estimated probabilities are precise. For instance, suppose that based on the actual relevance judgements of a group of users, the probabilities of relevance for two documents d_1 and d_2 are $p_1 = 0.74$ and $p_2 = 0.26$, respectively. Therefore, the correct rank is d_1 at first and then d_2. Assume that we have two sets of estimated probabilities by two models. One model gives $p_1 = 0.71$ and $p_2 = 0.29$, while the other gives $p_1 = 0.92$ and $p_2 = 0.08$. Both models give a correct rank. However, the second model overestimates d_1 and underestimates d_2. Theoretically, this example indicates that part of the estimation risk [2] could be independent of the rank. It also imposes practical risks in the applications, such as pseudo relevance feedback, where different estimated probabilities of relevance in the first-round retrieval will make a difference even when two ranks are identical.

It is important to clarify that the *rank-dependent risk* refers to the relevance estimation risk that can influence the rank, while the *rank-independent risk* does not [3]. In practice the ideal rank is usually unavailable, both types of risks may exist in the estimated relevance probabilities. Therefore, we can first aim to single out the effect of the rank-independent risk associated to different estimated relevance probability distributions when the resultant ranks are identical. The proposed smoothing method, which can preserve the ranking, is

[2] It should be noted that the estimation risk referred here is for each query, rather than across queries. We will analyze the influence of smoothing algorithm on the variance across queries in Section 3.2.2 in Chapter 3.

[3] The score normalization (Agarwal, Gabrilovich, Hall, Josifovski & Khanna 2009, Arampatzis, Robertson & Kamps 2009) can be regarded as one kind of rank-independent risk management. However, the proposed smoothing method is different from the existing score normalization method in that the former is a powers-based smoothing method, which is motivated by the difference between two rank-equivalent language modeling approaches (see Section 4.1.4).

suitable for the management of rank-independent risk. We also propose a weight allocation method which can re-rank the feedback documents on top of the proposed smoothing method.

For a given retrieval model, the rank-independent risk management method (i.e., the document weight smoothing) can be regarded as the micro-level adjustment, as opposed to the re-ranking approaches (tackling the rank-dependent risk) which can be regarded as macro-level adjustment for document relevance. Our proposed methods are applied and evaluated in both pseudo-relevance feedback and explicit relevance feedback contexts. Experimental results on several large-scale TREC collections have shown the effectiveness and stability of our methods.

1.4.3 A General Method for Distribution Separation

Irrelevant documents in feedback document set have long been a bottleneck in improving the performance of relevance feedback techniques. Based on our bias-variance analysis, removing irrelevant documents in relevance feedback can improve both the effectiveness and stability in the query model estimation.

In Chapter 5, we go beyond the document level by proposing a distribution separation method (DSM), which removes the irrelevant term distribution and separates the relevant term distribution (used for estimating the query model) from the mixture term distribution - corresponding to the whole relevance feedback document set. Our proposed method is based on distributions rather than documents, and is thus more general. In many cases, for example, we may have discarded old relevant or irrelevant documents after a number of search iterations. Nevertheless, we may still be able to keep updating relevant or irrelevant distribution incrementally, as well as keep tracking other items or features, such as query modifications, from which a term probability distribution can be formed.

Regarding the application of DSM, we consider two scenarios. One is with available seed irrelevant documents and the other is without any seed irrelevant data. For the former one, we can use the a small amount of explicit relevance feedback/judgements, which can indicate a document is relevant or not, to get the seed irrelevant documents. For the second scenario, automatic methods are needed to obtain the irrelevant documents or distribution. To this end, we adopt outlier detection methods by treating the irrelevant documents/terms are outliers. We systematically evaluate the proposed DSM on several large-scale TREC data sets. Evaluation results from extensive experiments demonstrate its effectiveness and stability.

1.4.4 Some Further Explorations

In addition to the above main contributions, we have also started to some further explorations on the application of proposed framework to personalization and on novel theoret-

ical angles (e.g., Quantum theory perspective) for the estimation of relevance (including document relevance and query relevance). Note that these are preliminary investigations that aim to set the scene for the future work.

Specifically, we first try to apply the bias-variance analysis in the personalization task. We can consider different users as different queries. To evaluate a personalization technique, there are two criteria, i.e., the mean user satisfaction over all users (corresponding to mean retrieval performance over all queries) and the variance of user satisfaction across individual users (corresponding to the variance of retrieval performance across individual queries). Then, we can use bias-variance analysis to study how to improve the user satisfaction for each user.

The bias-variance tradeoff idea was partially inspired by the Heisenberg's Uncertainty Principle, which states a tradeoff between the momentum and the position of an electron or photon and one can not precisely measure them simultaneously. We were initially seeking an analogy of such a tradeoff in IR. Bias-variance tradeoff can be considered as a kind of the uncertainty principle too (Grenander 1952, Geman, Bienenstock & Doursat 1992). We believe that to investigate Heisenberg's Uncertainty Principle in IR in depth, we need to build analogies of the quantum phenomenon in IR problems (e.g., document relevance estimation and query model estimation).

We propose an analogy of photon polarization (a key Quantum experiments) in IR, particularly for relevance feedback. We can view documents as photons, and the document retrieval process as measuring the probability of each document that can pass through the query's retrieval filter (as polarization filter). Then, the measured probability can be regarded as the estimated probability of relevance of each document. Quantum polarization experiment usually inserts an additional filter between the original filter and the photon receiver (e.g. a screen). Similarly, in query expansion, the expanded query is constructed for the second-round retrieval. We have derived formulations and constructed initial experiments in (Zhang, Song, Zhao & Hou 2011).

1.5 Thesis Outline

The remainder of the thesis is organized as follows. In Chapter 2, we will review various models for relevant estimation, and address the limitations of the state-of-the-art methods. This Chapter begins with the background of relevance and the probabilistic estimation methods for the document relevance, followed by two typical relevance feedback techniques and a discussion for the true relevance model. After that, in Section 2.3, we will review recent works on improving retrieval effectiveness and retrieval stability. We then move on to review the recent related work in addressing IR risks based on mean-variance analysis in Section 2.4. Finally, we will review the quantum-inspired IR, which is an emerging area and inspires some main ideas in this thesis.

The bias-variance analysis framework is proposed in Chapter 3. Specifically, in Section 3.1, we formulate the performance bias-variance, as well as the estimation bias-variance. Section 3.2 presents and analyzes various factors corresponding to various estimated query models in relation to the bias-variance tradeoff, along with a set of hypotheses. These factors include the query model complexity, query model combination, document weight smoothing, and irrelevant documents removal. In Section 3.3, we move on to the evaluation of concerned query language models, followed by discussions on the potential impact of the bias-variance analysis. Finally, we summarize our contributions about the bias-variance analysis.

In Chapter 4, we investigate document weight smoothing which is demonstrated as an important factor affecting the bias-variance in Chapter 3. Specifically, we propose to study the rank-independent risk associated to document weight smoothing in Section 4.1. Section 4.2 then presents two weight allocation methods to tackle the rank-dependent risk. We will construct experiments to evaluate the document weight smoothing and allocation methods in Section 4.4.

Another key factor that affects bias-variance is removing irrelevant documents. Chapter 5 presents a Distribution Separation Method (DSM) which goes beyond the document level of removing irrelevant documents directly. DSM is a distribution-level method which separates the true relevance distribution from the mixture one, by removing a seed irrelevance distribution. We will give the formulation of DSM in Section 5.1 which includes theoretical justifications and analysis. In Section 5.2.2, we propose to adopt outlier detection techniques to automatically obtain a seed irrelevant distribution, which can make DSM more practical and feasible. Section 5.3 gives a systematic evaluation about DSM based on not only retrieval performance, but also performance bias-variance.

We keep exploring the bias-variance formulation in the personalization task in Chapter 6. We also try to think about the estimation of relevance (including document relevance and query relevance) via a novel the Quantum theory perspective. Specifically, we propose an analogy of photon polarization (a key Quantum experiments) in IR (see Section 6.2).

At last, in Chapter 7, we conclude this thesis by summarizing our contributions in Section 7.1. We also address the limitations of our work and propose the future works in Section 7.2.

Chapter 2

Literature Review

Our literature review begins with the background of relevance and the probabilistic estimation methods for the document relevance. Then, two typical relevance feedback techniques for estimating expanded query model are introduced. We will also discuss the estimation of the true relevance model based on truly relevant feedback documents. After that, we will review recent proposed methods to improve the retrieval effectiveness and retrieval stability for document relevance estimation and relevance feedback. We then move to the most recent related work addressing IR risks based on mean-variance analysis. At last, we will briefly review the quantum-inspired IR, which is an emerging area and also somehow inspires some main ideas in this thesis.

2.1 Probabilistic Models of Relevance

Relevance is a fundamental abstract concept in IR. There are a number of different (and not converging) definitions of relevance (e.g., Mizzaro 1997, Cosijn and Ingwersen 2000, Borlund 2003, Xu and Chen 2006). It has been recognized that relevance should not be addressed by a single definition as the algorithmic relevance measured as statistical similarity of document and query representations (Lavrenko 2004). Relevance can encompass topicality, novelty, intelligibility, reliability, authoritativeness, scope, appropriateness, etc. Therefore, relevance is a multi-dimensional complex concept whose different dimensions do not act in isolation, but rather interact on different levels (Cosijn and Ingwersen 2000). In this thesis, however, we are mainly concerned with the topical relevance and probabilistic models to estimate such relevance.

Over decades, many probabilistic models have been developed (Maron & Kuhns 1960, Lafferty & Zhai 2003, Zhai 2007, Robertson & Zaragoza 2009) to estimate document relevance with respect to an information need (often represented as a query). The probability of relevance of each document corresponds to one basic retrieval question (Spärck Jones, Walker & Robertson 2000, Lafferty & Zhai 2003): what is the probability of a document

d is relevant to a query q? Accordingly, the probability of relevance can be formulated as $p(r|d,q)$ (Robertson & Zaragoza 2009).

In the classical probabilistic retrieval models (Robertson & Zaragoza 2009), it is usually the $p(r|d,q)$'s odds-ratio, i.e., $p(d|r,q)$ that is actually estimated. Hence, the event space of classic probabilistic models (e.g., the RSJ model (Robertson & Spärck Jones 1976)) is the space of documents, in relation to the query (Robertson 2005). The relevance component r is explicitly considered in the formulation (e.g., $p(r|d,q)$ or $p(d|r,q)$). There is a series of ranking-equivalent operations (e.g., odds-ratio) and approximations to compute final relevance scores. Since the final scores are often not probabilities, it might be necessary to transform the final scores to probabilities (Arampatzis et al. 2009, Agarwal et al. 2009, Gey 1994).

The language modeling (LM) approaches (Ponte & Croft 1998, Zhai & Lafferty 2001) are derived by asking how probable it is that this document generates the query (Sparck Jones, Robertson, Hiemstra & Zaragoza 2003). Early LM approaches compute the query likelihood $p(q|d)$, which can be formulated as:

$$p(q|\theta_d) = \prod_{j=1}^{m_q} p(q(j)|\theta_d) \tag{2.1}$$

where $p(q|\theta_d)$ is the query-likelihood, q is the original query, $q(j)$ is the j^{th} term of q, m_q is q's length, and θ_d is a smoothed language model for a document d.

The typical event space of LM approaches is the space of queries (Robertson 2005), which can be considered as a dual space of the event space in the classical probabilistic models. LM assumes that the query can be generated from the term distribution of document. Since original query terms may not be seen from a document, smoothing term distribution of the document plays a key role in the language modeling approaches (Zhai & Lafferty 2001).

The term probability smoothing methods basically smooth the maximum likelihood model of a document with the collection model. The maximum likelihood model can be formulated as

$$p(w|d) = \frac{tf(w,d)}{\sum_v tf(v,d)} \tag{2.2}$$

where $tf(w,d)$ is the occurrence frequency of a term w in a document d. After smoothing, we can rewrite the term probability to a language modeling θ notation as $p(w|\theta_d)$. One typical smoothing method is Jelinek-Mercer method, which constructs a linear combination between the maximum likelihood model and the collection model:

$$p_{\mu_1}(w|\theta_d) = (1-\mu_1)p(w|d) + \mu_1 p(w|\mathcal{C}) \tag{2.3}$$

where μ_1 is the linear combination coefficient (also called the interpolation parameter) and

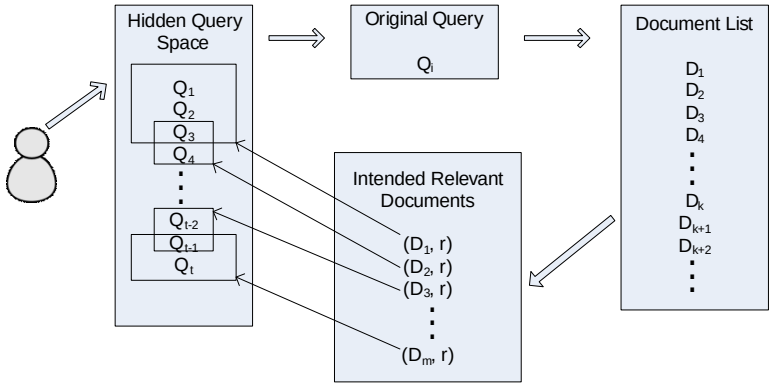


Figure 2.1: An example of query-generation idea in relevance feedback

$p(w|\mathcal{C})$ is the collection model. Another typical smoothing method is Dirichlet Method, which is given by

$$p_{\mu_2}(w|\theta_d) = \frac{tf(w,d) + \mu_2 p(w|\mathcal{C})}{\sum_w tf(w,d) + \mu_2} \tag{2.4}$$

where μ_2 is the Dirichlet smoothing parameter.

The query likelihood (i.e., $p(q|d)$ or $p(q|\theta_d)$) did not explicitly take into account the relevance component r in its formulation. Lafferty and Zhai (2001, 2003) then linked the LM approaches to the probability of relevance $p(r|d,q)$ and considered the classical probabilistic models and language modeling approaches into a unified generative model. It turns out that the classical probabilistic model (Robertson & Zaragoza 2009) is from the document-generation point of view, while the language modeling approaches are from the query-generation perspective [1].

2.2 Relevance Feedback

2.2.1 Query-Generation Idea in Relevance Feedback

The query-generation idea in language modeling approaches fits nicely in typical IR scenarios, particularly in relevance feedback. Let us see an example (in Figure 2.1), where a user has an information need (IN). Accordingly, we assume that there should be a hidden query space, which includes all the possible queries (or called query models) for representing the IN. This assumption is reasonable since before the search, even the user himself might not know how to represent his IN by a query. As always, the user would input an original query, which is from the query space and is often very simple. After the first-round retrieval, the system will return a document list D to the user. Through the relevance feedback, we could get a document set D_R that the user intends to regard as relevant to the IN. Then, we can derive a refined query model generated by D_R that is expected to

[1] We are aware that the document-generation idea is also used in some language modeling approaches (Manning, Raghavan & Schtze 2008), but we are focused on the query-generation idea in LM.

better reflect the user IN, subject to the amount of relevance feedback data we can get. This example shows a scenario of explicit relevance feedback.

Even though in the ideal case D_R contains all the truly relevant feedback documents that can reflect the IN, the derivation for the true relevance model is still challenging. First, one needs to answer whether a true or ideal query model should have multiple query representations corresponding to multiple topical aspects, or a single query representation that can balance multiple topical aspects. If we fix each query model as one single query representation (i.e., one query language model) as used in this thesis, a following challenge is that how to estimate such a true relevance model that can be consistent in different relevance feedback methods, as well as have the maximum retrieval performance for each query. To address this, we need to solve the effectiveness-stability tradeoff challenge mentioned in the introduction. Next, we first review two typical relevance feedback techniques and discuss the true relevance model.

2.2.2 Relevance Feedback Methods for Query Model Estimation

A classical method to construct the refined query is Rocchio's relevance feedback (Rocchio 1971), which aims to boost the terms from relevant documents and reduce the weights of terms from irrelevant documents. Note that we can also use Rocchio's model to perform negative feedback by ignoring the component *w.r.t.* relevant documents. Rocchio's relevance feedback is initially based on the Vector Space Model (Salton, Wong & Yang 1975), and here we rewrite it in the form of probabilistic models:

$$p(w|\widehat{\theta}_q^{(f1)}) = \alpha \times p(w|\widehat{\theta}_q^{(o)}) + \beta \times \frac{1}{|D_R|} \sum_{d \in D_R} p(w|\theta_d) - \gamma \times \frac{1}{|D_I|} \sum_{d \in D_I} p(w|\theta_d) \tag{2.5}$$

where the $\widehat{\theta}_q^{(o)}$ and $\widehat{\theta}_q^{(f1)}$ stand for the original and feedback-based refined queries, respectively; D_R and D_I indicate the relevant and irrelevant document sets, respectively; α, β and γ are three parameters. It turns out that this model needs explicit feedback for all relevant and irrelevant documents, and it involves three parameters to tune. The document weight in Rocchio's method can be formulated as:

$$S_q(d) = \begin{cases} \dfrac{\beta}{|D_R|} & d \in D_R \\ -\dfrac{\gamma}{|D_I|} & d \in D_I \end{cases} \tag{2.6}$$

Another well regarded pseudo-feedback method is the Relevance Model (Lavrenko & Croft 2001), which assumes that both the original query and top ranked documents are samples from a relevance model R. RM [2] can be used to estimate an expanded query

[2] In (Lavrenko & Croft 2001), RM has two versions, namely RM1 and RM2. In this thesis, we are mainly concerned about RM1.

language model.

$$p(w|\widehat{\theta}_q^{(f2)}) = \sum_{d \in D} p(w|\theta_d) \frac{p(q|\theta_d)p(\theta_d)}{\sum_{d' \in D} p(q|\theta_{d'})p(\theta_{d'})} \tag{2.7}$$

where $\widehat{\theta}_{q_i}^{(f2)}$ represents the feedback-based expanded query model, $p(\theta_d)$ represents the prior probability of document d, D denotes the documents that generate the expanded query model, $p(q_i|\theta_d)$ computes the query-likelihood (QL) score, and the normalized QL score serves as the document weight:

$$S_q(d) = \frac{p(q|\theta_d)p(\theta_d)}{\sum_{d' \in D} p(q|\theta_{d'})p(\theta_{d'})} \tag{2.8}$$

Terms with top probabilities in the distribution $p(w|\widehat{\theta}_q^{(f2)})$ can be used as the expanded query model (one kind of revised query model).

The documents in D are often pseudo-relevant documents (i.e., top-ranked documents after the first-round retrieval), rather than the truly relevant documents. Therefore, one problem of RM is that the document set D is often a mixture of relevant and irrelevant documents. The term distribution derived by RM is thus a mixture of relevant and irrelevant terms. Therefore, we do not consider the mixture term distribution $\theta_q^{(f2)}$ in Eq. 2.7 as the true relevance model. In practice, the inclusion of irrelevant documents/terms can also largely hurt the retrieval performance of RM.

2.2.3 On the True Relevance Model

The ideal estimation of the true relevance model should have the best retrieval performance for each query. If we use AP as the evaluation metric, the maximum AP is 1, meaning that all the relevant documents are retrieved and ranked before the irrelevant ones. However, this maximum AP is usually not achievable in practice.

Therefore, we are trying to develop a *true query model* to approximate the *true relevance model*. The true query model is expected to have the upper-bound performance for each query, among all the query models we will study in the relevance feedback scenario. The true query model corresponds to the target estimation method T in Chapter 1. The specific formulation of the true query model is needed for our further analysis.

We now address the derivation for the true relevance model based on two conditions. First, as discussed previously, the true query model should be generated from the truly relevant documents. Second, we also expect that the estimation of the true query model can be consistent in the aforementioned two relevance feedback methods.

Correspondingly, two factors, i.e., the *irrelevant documents/terms removal* and *document weight smoothness*, are important. They are all related to the rationality of the document weight. For instance, if the document weight for an irrelevant feedback document is set to 0, it means that this irrelevant document has been removed in the query

model estimation.

After removing all irrelevant feedback documents, another problem is the smoothness of the document weight. We think that the ideal document weight should be proportional to the relevance judgements/values of documents. Correspondingly, we assume that the ideal document weight

$$S_q^*(d) \propto r_q(d) \tag{2.9}$$

where $r_q(d)$ is the true relevance judgement/value (which can be binary or graded) of document d given the query q. Since now every relevant documents with the same relevance degree has the same document weight, the document weights in Eq 2.9 are more smooth than the normalized query likelihood (see Eq. 2.8).

This choice of document weight can also potentially make the two kinds of estimation methods (i.e. Rocchio and RM) become more consistent. The two methods yield different query models (see Eq. 2.5 and Eq. 2.7). However, if we use the document weight in Eq. 2.9 for RM, the two methods could estimate the same query model. To illustrate this, we first ignore the component *w.r.t.* the original query model $\widehat{\theta}_q^{(o)}$ in Rocchio's model. Actually, RM can be also combined with the original query model (Abdul-Jaleel, Allan, Croft, Diaz, Larkey, Li, Metzler, Smucker, Strohman, Turtle & Wade 2004, Lv & Zhai 2009).

Let us see the case when $r_q(d)$ is binary, where $r = 1$ means relevant and $r = 0$ means irrelevant. These relevance values are typical settings in the TREC test collections. Suppose we know the relevance judgement of every document, and we can use the document weight in Eq. 2.9 for both RM and Rocchio's method. Then, we can have

$$p(w|\widehat{\theta}_q^{(f1)}) \propto \sum_{d \in D_R} p(w|\theta_d) \quad \text{and} \quad p(w|\widehat{\theta}_q^{(f2)}) \propto \sum_{d \in D_R} p(w|\theta_d) \tag{2.10}$$

It turns out the two methods can give the same query model. Recall that we ignored the component *w.r.t.* the original query model $\widehat{\theta}_q^{(o)}$ in Rocchio's model.

When $r_q(d)$ has another two values, say 1 and -1, where $r = 1$ means relevant and $r = -1$ means irrelevant. Using the document weight in Eq. 2.9, the two methods can also give the same query model.

$$p(w|\widehat{\theta}_q^{(f1)}) \propto \sum_{d \in D_R} p(w|\theta_d) - \sum_{d \in D_I} p(w|\theta_d) \quad \text{and} \quad p(w|\widehat{\theta}_q^{(f2)}) \propto \sum_{d \in D_R} p(w|\theta_d) - \sum_{d \in D_I} p(w|\theta_d) \tag{2.11}$$

This consistency after using the document weight in Eq. 2.9 is very important in the derivation of the true query model. It is necessary for the true query model can be compatible among different estimation methods. In this thesis, we are mainly focused on the binary relevance. Therefore, the true query model can be estimated by:

$$p(w|\theta_q) \propto \sum_{d \in D_R} p(w|\theta_d) \tag{2.12}$$

This estimation of the true query model has some good properties. First, it fully complies with the query-generation idea and the query is generated by truly relevant documents. Second, it does not involve any extra assumptions, e.g., this relevant document may have bigger document weight just because that the query likelihood is higher. Third, it does not have adjustable parameters. At last, we will show that this true query model has the optimal retrieval performance in the experiments. We will build our theoretical and empirical analysis based on this representation.

We do not argue that the true query model in Eq. 2.12 is the idea estimation of true relevance model. The derivation of true relevance model could still be an open problem. However, we argue that the true relevance model should be estimated using a feedback-based method (e.g., RM) with the true relevance information (e.g., relevance judgements for documents).

2.2.4 Open Research Problems in Relevance Feedback

There are a number of open research problems related to relevance feedback approaches, that are to be addressed in this thesis.

First, how can we build a theoretical framework to analyze the estimation quality of any estimated query model *w.r.t.* the true query model? The estimated query model could be generated from a feedback document set that includes irrelevant documents. It is important that the framework considers various conditions corresponding to various ratios of irrelevant documents in the whole feedback document set.

Second, how can we make the document weights match the true document relevance values as well as possible? In practice, we often do not have the explicit feedback data and hence the true document relevance data is not available. It is important to smooth the document weight properly to improve the retrieval performance.

Third, how can we deal with some scenarios when we may know the irrelevant terms but we do not know the irrelevant documents. It is important to make use of the irrelevant term distribution to improve the estimation quality of the query model. This can also make our model become more general.

2.3 Improving Retrieval Effectiveness and Stability in LM

Now, we are going to review models and techniques that aim to improve the retrieval effectiveness and/or stability of the language modeling approaches. The retrieval effectiveness refers to the mean retrieval performance (e.g., mean average precision (MAP)) over all queries, while the retrieval stability is in the sense that the performance is stable across all queries.

2.3.1 Effectiveness-Oriented Methods

Smoothing in Language Modeling

The *smoothing* method is often related to smoothing the term probability of document language model and it is a very important factor in affecting the retrieval effectiveness of language modeling approaches (Zhai & Lafferty 2001, Ponte & Croft 1998). It generally includes *global* smoothing methods (Miller, Leek & Schwartz 1999, Ponte & Croft 1998, Zhai & Lafferty 2001) and *local* smoothing strategies (Liu & Croft 2004, Kurland & Lee 2004, Wei & Croft 2006).

Global methods smooth every document language model with the same background model, e.g., the collection model in Eq. 2.3 and Eq. 2.4. Local methods basically smooth the current document language model with its similar documents, e.g., utilizing corpus graph structures (Liu & Croft 2004, Kurland & Lee 2004, Wei & Croft 2006). An optimization framework for smoothing language models (Mei, Zhang & Zhai 2008) have been proposed to take into account two goals: fidelity and smoothness. Fidelity means that the smoothed language model should be close to the original language model, while smoothness means that the similar/close documents should have similar language models. It is argued that the optimization method in (Mei et al. 2008) can optimize the tradeoff between fidelity and smoothness, leading to a unified explanation and improved retrieval performance (e.g., MAP).

Score Regulation for Re-ranking

Various methods have been proposed to regularize the document relevance score or revise the document prior, thereby adjusting the document weights in relevance feedback. Based on the clustering hypothesis (Tombros & van Rijsbergen 2004), the score regulation method (Diaz 2005, Diaz 2007, Diaz 2008) forces the topically related documents to have similar relevance scores. In a similar manner, the graph-based smoothing framework proposed in (Mei et al. 2008) can also smooth the document relevance scores. To the best of our knowledge, neither of the above methods has been used to smooth the relevance scores for relevance feedback [3]. Moreover, they do not explicitly consider the document weight smoothness along the document rank list. As for revising the document prior, the rank-related prior was proposed in (Li 2008) by utilizing the document rank and document length. This method, however, does not consider the inter-document similarity. The rank-related priors (Li 2008) can be formulated as:

$$p(\theta_d) = \frac{1}{Z} \times \frac{\alpha + |d|}{\beta + Rank(d)} \tag{2.13}$$

[3]For the relevance feedback task, they (Mei et al. 2008) just used the DMWG method (i.e., smoothing document language model with word graph), rather than the DSDG one (i.e., smoothing document relevance score with document graph).

where $|d|$ is the d's document length and $Rank(d)$ is d's rank. This prior $p(\theta_d)$ and the QL scores are integrated as the document weights in RM (see Eq. 2.7).

Considering Term Dependency

Relevance Model (RM) does not explicitly consider the term dependency among query terms and those terms occurring in relevance feedback documents (Song, Huang, Bruza & Lau 2012). Recently, some methods have been proposed to model the term dependency in the ranking function and relevance feedback techniques.

In (Bruza & Song 2003), the term dependency captured by a probabilistic variant of the Hyperspace Analogue to Language (HAL) model (Burgess, Livesay & Lund 1998) has been integrated in Relevance Model. Song and Bruza (2003) presents an information flow model to capture the high-order term dependency by selecting terms with high-degree association to the subsets of the query. The information flow association is then integrated into language modeling approaches and achieves good query expansion results (Bai, Song, Bruza, Nie & Cao 2005). In (Pickens & MacFarlane 2006), a term context model based on maximum entropy is proposed to estimate the dependency between terms in documents and the query. Recently, an aspect query model with association rule mining (Song, Huang, Rüger & Bruza 2008, Song et al. 2012) has been proposed and improves the information flow based query expansion and the standard relevance model. More recently, in (Hou, He, Zhao & Song 2011) a pure high-order term dependency mining method has been proposed by using the information geometry. The term " pure " means that the high-order dependence that cannot be reduced to the random coincidence of lower-order dependencies. Experimental results (Hou et al. 2011) shows that this pure high-order dependence is useful in the scenario of relevance feedback.

A Markov random field (MRF) model (Metzler & Croft 2005) has been proposed to go beyond the independent term assumption in unigram and bigram retrieval models. The MRF model is then integrated in the query expansion models for relevance feedback, and a latent concept expansion (LCE) model which shows promising results has been proposed (Metzler & Croft 2007). The expanded concepts (single or multiple words) in LCE are chosen from the top ranked documents in the initial retrieval results, and weighted independently of the original query terms. This assumption may lead to the risk of topic drift, especially with long documents. In order to tackle the problem, Lang and Metzler (Lang, Metzler, Wang & Li 2010) proposed a Hierarchical MRF model based on LCE to model the dependencies between expansion terms and original query at the passage level.

The aforementioned methods mostly rely on statistical models but do not take into account the syntactic/linguistic information in the text. A dependency language model (Gao, Nie, Wu & Cao 2004) has been proposed to take into account not only the statistical dependency but also linguistic information (e.g., POS tags) in the language model. Recently,

the event information has been integrated into query expansion methods (HAL and RM) and shows performance improvements. The events are extracted from predicate-argument structures and a dependency parsing tree (Yan, Maxwell, Song, Hou & Zhang 2010).

Supervised Learning for Good Documents or Terms

Recently, He and Ounis (2009) proposed to detect relevant feedback documents by using a variety of features and classifying the feedback documents into relevant and irrelevant sets. Note that detecting relevant documents and removing irrelevant documents are dual problems. Cao et al. (2008) proposed to select good feedback terms in the pseudo relevance feedback documents, also by classification methods using a number of defined features. Although the above methods show promising results, they rely on supervised learning methods, e.g., classification tools. Our work does not use learning methods.

2.3.2 Robustness-Oriented Methods

Here, we mainly review those methods to improve the robustness of relevance feedback, where the query expansion is conducted and used for a second-round retrieval. The robustness of relevance feedback is related to the stability of retrieval performance across all queries (Amati et al. 2004, Collins-Thompson 2009a).

The robustness issue of query expansion is rooted on the fact that for some queries, the retrieval performance has been improved, while for other queries, the retrieval performance drops. One main cause of such dropped performance can be the query-drift phenomenon, i.e., the change of the query topic (after query expansion) away from the underlying intent of the original query input by the user (Collins-Thompson 2008).

Query-Drift Problem

Collins-Thompson (2008) provided a comprehensive review about the query-drift problem. According to the analysis, there are three reasons for the query-drift problem, including poor initial retrieval, poor coverage of query aspects, and noise terms in feedback model.

The poor initial retrieval is related to the poor estimation of document relevance in the initial ranking. As a result, there are very few relevant feedback documents or the relevant ones do not have high document weights in the retrieved documents. Therefore, the inclusion of irrelevant feedback documents (with high impact on the query model estimation) may lead to the revised/expanded query drifting from the underlying intent of the original query. One straightforward solution to this problem is re-ranking feedback documents (Mitra, Singhal & Buckley 1998, Crouch, Crouch, Chen & Holtz 2002). In this thesis, we will pay more attention on smoothing feedback document weights without and with changing the ranking.

It is argued that the good coverage of query aspects is an important factor in preventing the query drift (Harman & Buckley 2004, Collins-Thompson & Callan 2005). For example, given a original query "President Hu Jintao visits US", if the revised query mainly covers "Hu Jintao" but neglects "US", it certainly drifts from the original query. A poor initial retrieval can result in a poor coverage of query aspects. The good initial retrieval could also lead to poor coverage of query aspects (Buckley 2004).

The noise terms are particularly those terms that have high probabilities/scores but are irrelevant to the information need. The cause for inclusion of noise terms is because that the widely-used tf/idf scoring scheme for terms can make some highly frequency noise terms (e.g., stop words) due to the high tf value (Collins-Thompson 2008). It is argued in (Collins-Thompson 2008) that the feedback model can have less noises by using multiple predictors for the feedback-based query model.

Combination with Original Query

To prevent the query drift from the original query, one approach is to combine the feedback-based query model with the original query model. For example, in the above example, if the revised query mainly covers "Hu Jintao" and certainly drifts from the original query, we can combine the revised query with the original query to adjust the query model estimation. This combination process can be formulated as:

$$\widehat{\theta}_q^{(c)} = \lambda \widehat{\theta}_q^{(o)} + (1-\lambda)\widehat{\theta}_q^{(f)} \tag{2.14}$$

where $\widehat{\theta}_q^{(c)}$ is the combined query models, λ is the combination coefficient of the original query $\widehat{\theta}_q^{(o)}$, and $1-\lambda$ is the coefficient of the feedback-based expanded query model $\widehat{\theta}_q^{(f)}$. The combined query model in Eq. 3.45 is often referred as RM3 (Abdul-Jaleel et al. 2004).

Tao and Zhai (2006) proposed a method to integrate the original query with feedback documents in a probabilistic mixture model and then regularize the parameter estimation. Li (2008) considered the original query as a short document, used rank-related priors and investigated term selection in RM. Lv and Zhai (2009) proposed to adaptively combine the original query and feedback information.

Fusing Retrieval Results

In (Zighelnic & Kurland 2008), it is proposed to fuse the document relevance scores corresponding to the original query and the expanded query, in order to prevent query drifting. In (Kozorovitzky & Kurland 2011, Meister, Kurland & Kalmanovich 2011), the inter-document similarities are adopted into the fusion method to further improve the robustness of the query expansion.

Here, we will show that the fusing method and the aforementioned combination method are essentially equivalent/similar to each other, when the relevance scoring function is

linear. Let us first rewrite the combination model for query q in a more general form:

$$\widehat{\theta}_q^{(c)} = \sum_i \lambda_i \widehat{\theta}_q^{(i)} \tag{2.15}$$

where $\widehat{\theta}_q^{(c)}$ is the combined query model, $\widehat{\theta}_q^{(i)}$ can be any query model (e.g., the original query model or any feedback-based query model) with the combination coefficient λ_i which satisfies that $\sum_i \lambda_i = 1$.

Given this form of combined query model, we can have the relevance score of a document d:

$$S(d, \widehat{\theta}_q^{(c)}) = S(d, \sum_i \lambda_i \widehat{\theta}_q^{(i)}) = \sum_i \lambda_i S(d, \widehat{\theta}_q^{(i)}) \tag{2.16}$$

It turns out that the relevance score (i.e. $S(d, \widehat{\theta}_q^{(c)})$) *w.r.t.* the combined query model $\widehat{\theta}_q^{(c)}$ is the fused/combined relevance scores (i.e. $\sum_i \lambda_i S(d, \widehat{\theta}_q^{(i)})$) of every relevance score $S(d, \widehat{\theta}_q^{(i)})$ of document d given the query model $\widehat{\theta}_q^{(i)}$.

In Eq. 2.16, we actually assume that the relevance score function $S(d, \widehat{\theta}_q)$ is linear. This assumption is generally valid in language modeling approaches. The negative cross entropy $H(\widehat{\theta}_q, \theta_d)$ between any estimated query model $\widehat{\theta}_q$ and document model θ_d can be used as the retrieval model for both original and expanded query models. We can have

$$-H(\sum_i \lambda_i \widehat{\theta}_q^{(i)}, \theta_d) = -\sum_i \lambda_i H(\widehat{\theta}_q^{(i)}, \theta_d) \tag{2.17}$$

This shows the linear assumption of the negative cross entropy is valid. The negative cross entropy model is rank-equivalent with the common-used KL-divergence retrieval model which will be described in the next section. After normalization, they have the same estimated relevance scores over all documents (Zhang, Song, Wang, Zhao & Hou 2011).

Uncertainty and Combination of Feedback Models

In Rocchio's model and Relevance Model, given one single query, only one single feedback-based query model (or called feedback model) is generated to perform the query expansion task. there have been studies in the uncertainty of feedback models by combining multiple query representations/models for each query (Collins-Thompson 2008).

Carpineto et al. (2001) proposed to select terms from different distributional methods based on feedback documents. Carpineto et al. (2002) further combined different term ranking methods in automatic query expansion. Amati et al. (2004) proposed to derive a more robust query model using selective query expansion. The selection criteria is based on a heuristic called InfoQ.

More recently, Collins-Thompson and Callan (2007) proposed to resample different feedback document models using Bootstrap sampling. It also provides a principled way for

the combination of different feedback models. Lv et al. (2011) proposed a FeedbackBoost method to improve robustness of the expanded query model. In (Collins-Thompson 2009b, Dillon & Collins-Thompson 2010), the risk and reward tradeoff and optimization for query expansion were discussed. We are going to investigate the risk issue in more detail in the next section.

2.3.3 The Existing Risk and Robustness Metrics

In the literature, Collins-Thompson (Collins-Thompson 2009a) addressed the retrieval stability across queries using the general concept of variance, in the sense that the query expansion can not always improve the retrieval performance of the original query model. However, they did not compute the variance (i.e., VAP) of retrieval performance across queries. In (Collins-Thompson 2009a), the variance is considered as a risk, and the risk measurement (called as $R-Loss$) in (Collins-Thompson 2009a) is quite different from VAP. $R-Loss$ in (Collins-Thompson 2009a, Collins-Thompson 2009b) calculates the *average* net loss of relevant documents (due to failure), which is the number of relevant documents lost in the top 1000 retrieved documents by the query expansion.

We now review existing robustness metrics in (Zighelnic & Kurland 2008, Collins-Thompson 2009b) used in query expansion. In (Zighelnic & Kurland 2008, Collins-Thompson 2009b), the robustness metric is called as robustness index (RI). The robustness index $RI(Q) = (n_+ - n_-)/|Q|$, where n_+ is the number of queries helped, n_- is the number of queries hurt, after query expansion. $|Q|$ is the total number of queries. In the above example (see Table 1.1 in Chapter 1), A corresponds to the original query, and B corresponds to the expanded query. It turns out that the RI of both methods are 0. This means that the RI is insufficient to distinguish the retrieval robustness between A and B. More generally, two methods (not necessarily being A or B) may have the same $n+$ and $n-$, but they may have quite different AP for each query [4]. In this case, the RI also can not distinguish the robustness between these two methods. On the other hand, our robustness metric (i.e., $Bias^2+Var$) is able to distinguish the retrieval robustness between two methods (see Table 1.1), suggesting that method A is more robust than method B. It is also able to distinguish the robustness among methods even when they have the same $n+$ and $n-$ but have quite different retrieval performance.

Note that in (Zighelnic & Kurland 2008), another robustness measure, denoted as $<Init$, is also adopted to test the percentage of queries for which the retrieval performance after query expansion is worse than that of the original query. The smaller $<Init$ indicates the better retrieval robustness of the query expansion. However, because original query has the minimal $<Init$, which is 0, this metric always regard the original query as the most robust query. It is not sensible since if this is true, it may not necessary to build other

[4] For example, for a query, A can improve the initial performance by 10%, but B can improve the initial performance by 20%. This has the same contribution to $n+$. The same example can be made to $n-$.

more robust query models. In addition, $<Init$ is insufficient to distinguish the retrieval robustness between two methods when they have the same $<Init$. On the other hand, our robustness metric does not regard the original query as the most robust one (see my thesis for detailed results). For example, based on $Bias^2 + Var$, the true query model is usually more robust than the original query model. In Table 1.1, the target method T corresponds to the true query model, and it is more robust than A (which was assumed as the original query model). In addition, $Bias^2 + Var$ is able to distinguish the robustness between methods A and T even though they have the same $<Init$.

It turns out that the existing robustness metrics are different from ours in terms of formulations and observations. In addition, more importantly, regarding novelty, our robustness metric based on bias plus variance can have different theoretical properties. First, our metric provides a decomposition of retrieval robustness into retrieval effectiveness and retrieval stability. Second, it can be studied via bias-variance analysis, which provides a principled analysis methodology to analyze model complexity, model combination, and available relevance judgements. For instance, the original query model is less complex than the expanded query model, resulting in that the original query model has bigger bias but less variance – bias-variance tradeoff. The model combination and available relevance judgements can be helpful to reduce both bias and variance simultaneously, yielding more robust query model. Therefore, the proposed $Bias^2 + Var$, as well as $Bias$ and Var, can be not only evaluation metrics, but also provide a principled analysis methodology to analyze IR models.

2.3.4 Limitations of the State of the Art

It has been argued that the relevance feedback can improve the retrieval effectiveness (e.g., MAP for all queries) but may hurt the performance for some individual queries, leading to worse stability of the retrieval performance, compared with the initial ranking based on the original query model (Amati et al. 2004, Collins-Thompson 2009a, Collins-Thompson 2008). It turns out that a tradeoff between retrieval effectiveness and stability in estimating query language model does exist. In this thesis, we aim to gain a deep understanding of such tradeoff. Specifically, we need to address the following important research problems.

- *The decomposition of retrieval robustness.* The robustness is certainly related to both retrieval effectiveness and retrieval stability across all queries. However, to our knowledge, few formulation has been derived to address the decomposition of robustness into effectiveness and stability. Due to the lack of this decomposition, the connection (e.g., the tradeoff) between them can not be fully exploited.

- *A theoretical explanation for retrieval effectiveness-stability tradeoff.* We need to understand whether this tradeoff is an intrinsic phenomenon in query language mod-

eling and why existing models can improve the retrieval effectiveness, or stability, or both. Even though the optimization framework in (Collins-Thompson 2008) has taken into account the tradeoff, we aim to find a more general theoretical explanation from the statistical estimation theory and identify key factors affecting the tradeoff.

- *Model complexity analysis for query language model.* There are many different approaches or variants to estimate the query language model. However, little attention has been paid to the model complexity. We expect that the above explanation is able to consider the model complexity. It is also important to note that such model complexity intuition/formulation can be applied to other IR tasks and problems.

- *Estimation quality w.r.t. true query model.* Existing models do not formulate and evaluate the estimation quality of any estimation query model *w.r.t.* the true query model. We believe that this is very important for us to understand whether an estimation method is effective and/or stable. This kind of evaluation is complementary to traditional evaluation which only focuses on the retrieval performance.

- *Non-relevance analysis.* It is very important to analyze the effect of different amount of irrelevant documents in the whole feedback document set. A theoretical explanation is needed and a systematic evaluation should be carried out, not only in terms of retrieval performance, but also with regard to the estimation quality *w.r.t.* the true query model.

In this thesis, our contribution is to use bias-variance decomposition as the formalism to decompose retrieval robustness into retrieval effectiveness and retrieval stability across all queries. We then investigate the retrieval effectiveness-stability tradeoff through the bias-variance tradeoff. Using the framework, the model complexity, the estimation quality, and the non-relevance analysis can be conducted based on the principles and intuitions of bias-variance analysis. Moreover, we also propose a distribution separation model to remove the non-relevance term distribution, which goes beyond the document level of non-relevance removal.

2.4 IR Risk and Mean-Variance Analysis

2.4.1 Retrieval Risks

The probabilistic ranking principle (PRP) (Robertson 1977) suggests that the document ranking in the order of decreasing *probability of relevance* of documents can give the optimal ranking effectiveness (e.g., in terms of the expected precision (Robertson 1977)) and minimize the overall risk (van Rijsbergen 1979). The risk in (Robertson 1977, van Rijsbergen 1979) refers to the retrieval risk, which is based on the loss function associated

with a decision on whether or not to retrieve a document. Therefore, the retrieval risk is closely related to the ranking effectiveness.

The risk minimization framework (Lafferty & Zhai 2001, Zhai & Lafferty 2006) suggests that the optimal ranking strategies can be obtained through considering suitable loss functions in different IR tasks. The documents are ranked in an ascending order of the expected risks. The ranking based on the relevance-based loss function turns out to be equivalent to the ranking based on the probability of relevance $p(r|d,q)$ in the classic probabilistic model. The ranking based on the proportional distance loss functions (Zhai & Lafferty 2006) leads to a general model-based ranking, called negative KL-divergence model. For any estimated query model (including original or expanded query model), the document retrieval can be based on the negative KL-Divergence (Lafferty & Zhai 2001) between the estimated query language model $\widehat{\theta}_q$ and document language model θ_d:

$$-D(\widehat{\theta}_q|\theta_d) = -H(\widehat{\theta}_q, \theta_d) + H(\widehat{\theta}_q) \tag{2.18}$$

where $H(\widehat{\theta}_q, \theta_d)$ is the cross entropy (see Eq. 2.17) between $\widehat{\theta}_q$ and θ_d, and $H(\widehat{\theta}_q)$ is the entropy of the $\widehat{\theta}_q$. According to the derivation in (Lafferty & Zhai 2001, Ogilvie & Callan 2002), if the original query with a maximum-likelihood estimator is used as the estimated query model, the negative KL-divergence is *rank-equivalent* to the query-likelihood approach.

It is argued that the risk minimization framework (Zhai & Lafferty 2006) is more general than the risks or losses mentioned in (Robertson 1977, van Rijsbergen 1979). First, the decision-theoretic formulation in (Zhai & Lafferty 2006) is not limited to binary decision as used in (Robertson 1977, van Rijsbergen 1979). Second, The retrieval risks/losses are formulated not only in terms of relevance, but also other factors such as novelty and redundancy (Zhai & Lafferty 2006). Third, such a framework explicitly suggests the interactive IR process and models the user variable in its formulation.

2.4.2 Mean-Variance Analysis

Researchers recently realized that most estimators for document relevance are best match in response to the query. The best match is a point estimation, which neglects the uncertainty of the document relevance (Wang 2009, Wang & Zhu 2009, Zhu et al. 2009). In (Zhu et al. 2009), using Dirichlet distribution – the conjugate prior of multinomial distribution of a document, the posterior probability gives a general form of language modeling approaches. Using the first moment (i.e., mean) and second moment (i.e., variance) of each variable (corresponding to each term) of a Dirichlet Distribution, a moment-based ranking function is derived (Zhu et al. 2009). The variance component is supposed to represent the uncertainty of the estimation.

In (Wang 2009, Wang & Zhu 2009), the mean-variance analysis was conducted by

considering the analogy of Portfolio Theory (PT) in IR. PT suggests ranking should be a task about how to select the right combination of documents, rather than match each document individually, in response to the query. Based on Portfolio theory, both the uncertainty of relevance estimation and the inter-document dependency can be considered.

$$\begin{aligned} E[R_n] &= \sum_{i=1}^{n} w_i E(r_i) \\ Var(R_n) &= \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j c_{i,j} \end{aligned} \tag{2.19}$$

where the mean $E[R_n]$ represents the expected overall relevance scores, and the variance $Var(R_n)$ computes the uncertainty (risk) associated to the relevance scores. $E(r_i)$ is computed by the relevance score of the document d_i, and the score is considered a mean value. $c_{i,j}$ can encode the correlation/covariance between d_i and d_j, as well as the uncertainty (variance) associated to the relevance scores of d_i and d_j. A practical solution is also given in (Wang 2009, Wang & Zhu 2009) to construct the document ranking.

2.4.3 Reward-Risk Analysis for Query Expansion

In (Collins-Thompson 2009b), a convex optimization framework was proposed to balance the reward and risk in query expansion. The work is also initially motivated by the Portfolio theory. The main difference between the work in (Wang & Zhu 2009) and that in (Collins-Thompson 2009b) is that the former is for document ranking and the later is about the query expansion in automatic relevance feedback.

The objective function in the convex optimization framework mainly contains two parts:

$$\begin{aligned} R(x) &= p^T x \\ V(x) &= \frac{\kappa}{2} x^T \Sigma x \end{aligned} \tag{2.20}$$

where $R(x)$ is the reward and $V(x)$ is the risk associated to a vector $x = (x_1, \ldots, x_{|\mathcal{V}|})(x_i \in [0, 1])$. $|\mathcal{V}|$ is the number of all terms in the vocabulary. $p = (p_1, \ldots, p_{|\mathcal{V}|})$ and p_i can be the score/probability of term i given by the Relevance Model. The covariance matrix Σ is the covariance matrix for all the terms. The convex optimization framework is also able to take into account several constraints, e.g., domain knowledge, aspect balance, aspect coverage and query support, etc.

The reward and risk in Eq.2.20 can also be thought as the mean and variance respectively: the reward $R(x)$ represents the expected relevance scores of terms and the risk $V(x)$ represents the uncertainty/variance associated to the relevance scores. $V(x)$ also considered the covariance matrix among term variables. Recall that the covariance ma-

trix of document variables is also formulated in the uncertainty part in the mean-variance analysis for document ranking (Wang & Zhu 2009).

2.4.4 Limitations of the State of the Art

The mean-variance analysis actually inspire our bias-variance research. Now, we are going to first address some problems in the existing mean-variance based works.

- *Connection between mean and variance.* Although both mean and variance are considered in above formal models or evaluation metrics, the connection between mean and variance is still not clear. Without a understanding of such connection, it is difficult to explain what will happen to the variance if the mean changes, and vice versa? The goal is to improve the mean but reduce the variance simultaneously, but maybe the reality is that we can not do so in certain conditions (e.g., limited data/features), or we can just achieve this goal through extensively adjusting the parameters, or the improvement is actually minor. We should develop a theory about the connection, so that the tradeoff between the two can be explained and analyzed.

- *Rank-independent risk management.* The risks mentioned in the classic probabilistic models, the risk minimization framework and the mean-variance analysis are closely related to the ranking performance for each query. However, it neglects the rank-independent risk, which is independent of the current document ranking but can affect the next-round retrieval (e.g., relevance feedback). Therefore, it is important to consider such a rank-independent risk in a theoretical framework and develop formal models to manage such a kind of risk.

- *A unified bias-variance framework.* A unified bias-variance framework is needed to address both the above three problems as well as the those problems listed previously in Section 2.3, including the theoretical support for the observed tradeoff between the retrieval effectiveness (related to the mean) and stability (related to the variance), the model complexity of IR models, estimation quality *w.r.t.* the true query model, rank-independent risk and the non-relevance analysis.

Our contributions in this thesis are mainly to explore bias-variance analysis as a theoretical framework to address the above research challenges and problems. In addition, we propose to investigate rank-independent risk in the document weight smoothing method. The proposed analysis is expected to form an theoretical analysis framework and a new evaluation strategy for the query language modeling. The bias-variance analysis enriches the mean-variance analysis in the literature due to its ability to explore the IR model complexity, connection between mean and variance, as well as the non-relevance information, across all queries. It also has the potential to be applied into other IR tasks (e.g.,

personalization). Note that we do not argue that we have solved all the problems, but rather expect that our work can shed light on the IR theory on related problems.

2.5 Quantum Theory (QT) Inspired IR

The bias-variance analysis framework in this thesis was actually somehow inspired by the Heisenberg uncertainty principle, which states that it is not possible to measure the present position while determining the future momentum of an electron or photon. It means that there is a tradeoff of two uncertainties associated to position and momentum, respectively, of an electron or photon.

We were initially seeking the analogy of Heisenberg uncertainty principle in IR. Note that the uncertainty is a nature of IR problems. The key is to find an appropriate tradeoff in two uncertainties related to IR relevance estimation or evaluation metrics. We find that the bias-variance tradeoff is related to the Heisenberg uncertainty principle (Geman et al. 1992). We are not arguing that we have found some strong evidence of Heisenberg uncertainty principle, but we expect that this thesis can provide a start point for this problem. Now, let us first review the literature in the quantum-inspired IR area.

2.5.1 Quantum-inspired IR models

Recently, van Rijsbergen (2004) proposed to employ quantum theory (QT) as a theoretical formalism for modeling IR tasks. This work shows that major IR models (logical, probabilistic and vector) can be subsumed by the single mathematical formalism in Hilbert vector spaces (also can be complex space). Specifically, QT provides a geometrical vector representation for information objects (e.g., documents, queries, multimedia objects) in a complex Hilbert Space; measurement of observables as relevance status of information objects; probability calculation via the trace formula in Gleason's Theory (Gleason 1957); ability for logical reasoning through lattice structures, modelling the change of states via evolution operators.

Followed by van Rijsbergen (van Rijsbergen 2004), many approaches have been proposed and these approaches can be classified into three main themes (Song, Lalmas & van Rijsbergen et al. 2010): Spaces: geometrical representation and characterisation of context through semantic spaces; (2) Interferences: the interferences among documents, topics and user's cognitive status in contextual relevance measurement process; (3) Frameworks: general frameworks and operational methods for contextual and multimodal IR.

2.5.2 Limitations of the State of the Art

We now discuss some research problems related to the quantum-inspired IR models.

- *Uncertainty principle in IR.* To the best of our knowledge, little attention has been paid to the analogy of Heisenberg uncertainty principle in the IR problems and tasks. We believe this research direction is important because: 1) the uncertainty is an inherent nature in IR theory, models and applications. 2) Heisenberg uncertainty principle is a milestone to let people accept the Quantum Mechanics. Therefore, if we could develop an IR uncertainty theory like the Heisenberg uncertainty principle, it would make more and more researchers try to understand the Quantum-inspired IR research and make the Quantum-inspired IR models more applicable.

- *Application to the relevance feedback.* Current Quantum-inspired IR models are mainly related to the document ranking. Few related works were carried out for the relevance feedback, e.g., the query expansion tasks. Thus, it is important to develop the analogy of quantum phenomenon in relevance feedback task. As we addressed before, the uncertainty of the information need is an inherent problem of the query model estimation in relevance feedback. It is interesting to explore the uncertainty principle in relevance feedback like the uncertainty principle in Quantum Mechanics.

In this thesis, we are trying to address some problems. As we discussed before, the bias-variance analysis framework could be a simple analogy of the Heisenberg uncertainty principle in IR. Moreover, we also built the analogy of photon polarization (a key experiment in quantum world) in relevance feedback and developed algorithms to carry out the query expansion task. These works or thoughts are expected to provide some hints for the related research in the IR community.

Chapter 3

Bias-Variance Analysis Framework

In the previous chapter, we reviewed the literature about IR retrieval models and different methods for query model estimation in language modeling framework. We pointed out the limitations of state-of-the-art query models. One of them is the lack of a unified bias-variance framework which can not only provide novel evaluation metrics but also analyze the query model estimation in a principled way. In this chapter, we propose to study the *tradeoff* between retrieval effectiveness and stability from a novel theoretical perspective, i.e., bias-variance *tradeoff*.

The bias-variance tradeoff is fundamental in the estimation theory and has been extensively studied in density estimation (Zucchini et al. 2005), linear regression (Geman et al. 1992), classification (Valentini, Dietterich & Cristianini 2004), and other areas (Bishop 2006). In general, the bias represents the gap between the expectation (i.e. mean) of estimated values and the true target value, while the variance represents the variability over all estimated values.

This motivates us to formulate the *performance bias and variance* which are related to the retrieval effectiveness and stability, respectively. Specifically, assuming that we have a performance target (in practice, an upper bound performance), the performance bias represents the gap between the actual mean performance and the performance target. Thus, to improve the retrieval effectiveness as much as possible can be considered as an effort to make the gap between actual mean performance and performance target as small as possible. On the other hand, the performance variance corresponds to the variance of retrieval performance over different queries. Generally, the smaller performance variance reflects the better stability of the retrieval performance across all queries. In this manner, we can look into the problem of *improving* the retrieval effectiveness and stability from the perspective of *reducing* performance bias and variance, respectively. The proposed bias-variance formulation can provide more theoretical insights on the tradeoff between retrieval effectiveness and stability and explain whether the two retrieval criteria can be improved simultaneously.

In practice, the retrieval performance is assumed to reflect the quality of an estimated query model, given that a retrieval model is fixed to rank documents with respect to the query model. However, strictly speaking, it does not *directly* investigate the closeness of the *estimated* query model with respect to the *true* query model. Assume the true information need can be represented by a set of truly relevant documents, and the true query model can be generated from truly relevant documents. Such a true query model is expected to give the optimal retrieval performance [1]. We then formulate the *estimation bias and variance* of an estimated query model. The estimation bias and variance *directly* compute how closely an estimated query model can approach the true one. Specifically, the estimation bias represents the expected estimation error over all queries, while the estimation variance is the variance of estimation error (or quality) across different individual queries. The sum of bias and variance can yield the total estimation error which can directly indicate the total estimation quality. We think that the estimation bias-variance is also important, in addition to the performance bias-variance, in that it can give finer-grained insights on the estimated query model itself.

Based on general principles and intuitions of bias-variance tradeoff, we analyze several estimated query models through investigating factors that can affect query model estimation. We also analyze and explain different trends of bias and variance on different evaluation factors, e.g., different kinds of bias-variance, or different test collections. Based on these analysis, we then propose a set of hypotheses with respect to different factors on bias-variance tradeoff. A series of experiments based on TREC datasets have been conducted to test the hypotheses. Experimental results generally verify our hypotheses.

The proposed analysis is expected to form an theoretical analysis framework and a new evaluation strategy for the query language modeling. First, since the hypotheses are verified in our experiments, it turns out that the bias-variance tradeoff can be used to explain the tradeoff between the retrieval effectiveness and stability. Second, it can provide insights on how to improve effectiveness and stability separately, or simultaneously. For different applications with specific needs, e.g., stability-oriented tasks, one can adopt corresponding strategies in building query models. Third, it potentially leads to a new evaluation strategy, one can evaluate the estimation bias-variance of an estimated query model to get in-depth observations that are helpful to design better models to approximate the true query model. Last, we can use bias-variance figure plotted in Section 3.3, to observe the balance and trend of retrieval effectiveness and stability, in an integrated manner. The sum of bias-variance can be a robustness metric in terms of the combined effect of retrieval effectiveness and stability.

[1] The true query model is different from the typical expanded query model which are often generated from pseudo-relevant feedback (PRF) documents

3.1 Formulation of Bias and Variance

3.1.1 Introduction to Bias-Variance Analysis

The bias-variance analysis is a fundamental theory and has been extensively studied in parameter estimation (Lebanon 2010, Duda, Hart & Stork 2001), density estimation (Zucchini et al. 2005), linear regression (Geman et al. 1992), classification (Valentini et al. 2004, Lipka & Stein 2011), and other areas (Bishop 2006). We first briefly explain the classical bias-variance decomposition for the squared loss.

Let us consider an estimator $\widehat{y}$ for the unknown true target y, where $\widehat{y}$ is determined by the sample X. For different sample X, the value of $\widehat{y}$ varies. Thus, $\widehat{y}$ can be considered as a random variable. The expected squared error loss of the estimation can be decomposed to bias and variance:

$$\begin{aligned} E(\widehat{y}-y)^2 &= E(\widehat{y}-E(\widehat{y})+E(\widehat{y})-y)^2 \\ &= E(\widehat{y}-E(\widehat{y}))^2+(E(\widehat{y})-y)^2 \\ &= Var(\widehat{y})+Bias^2(\widehat{y}) \end{aligned} \tag{3.1}$$

where the expectation E is computed over all possible $\widehat{y}$, $Bias^2(\widehat{y})$ computes the squared error (i.e., $(E(\widehat{y})-y)^2$) of the expected value $E(\widehat{y})$ with respect to the true value y, and $Var(\widehat{y})$ computes the variance of $\widehat{y}$ across all samples.

The above formulation is a general description of the bias and variance. It can be applied to specific areas with specific explanations. For instance, in parameter estimation, the task is to estimate the parameter (e.g., mean or variance) of the underlying distribution of a given data sample [2]. On different samples (or sampling distributions), the estimated values can be different. In regression or classification, the task is to estimate the response value (in regression) or the class labels (in classification) for any test data point, given a training sample (or called training set). On different training samples, the estimated values could be different (Geman et al. 1992, Bishop 2006) and the estimated value can be considered as a random variable.

Generally speaking, given the limited size for each sample, there is a *tradeoff* between bias and variance (Geman et al. 1992). For example, a simple estimation method often involves less configurations (e.g., less parameters or assumptions) and has higher bias but lower variance, compared with the complex method (Geman et al. 1992). This means that the expected estimation error of the simple method is often larger than that of the complex one, but the estimated values of the simple method over different samples are more stable than those of the complex one. To reduce the bias and variance simultaneously, one often needs more data (e.g., larger sample size or more training data) (Brain & Webb.

[2] In this thesis, we consider the *sample* as a terminology of statistics and refer to each *sample* as a collection of data or information. In some other literature , a sample may be considered as a single data point.

1999, Bishop 2006, Perlich, Provost & Simonoff 2003), or well designed methods (e.g., combination method or so called ensemble method) (Valentini et al. 2004, Ghahramani, , Ghahramani & chul Kim 2003). In the context of query language modeling, we will analyze the above factors that can affect the bias and variance in Section 3.2.2.

3.1.2 Performance Bias-Variance

We now explain the analogy of the bias-variance analysis in IR. According to previous introduction, the bias considers the expected estimation value over all samples, while the variance represents the variability of the estimated values across different samples. In IR, for evaluating a retrieval model or a query model, we are concerned about its mean retrieval performance over all queries, and also the variability of retrieval performance across different queries. We can consider each query and its corresponding data (e.g., query terms, retrieved documents, or relevance judgements if available) as a sample to test the retrieval performance. Therefore, we can let the actual retrieval performance be a random variable and it can be different on different queries.

Recall that we consider the actual performance $\widehat{P}$ as a random variable. For a query q_i, we denote its actual retrieval performance as $\widehat{P}_i$, and denote the corresponding performance target as P_i. In query model estimation, given the query q_i, $\widehat{P}_i$ and P_i correspond to the estimated query model and the true query model, respectively.

Now, let $P_i - \widehat{P}_i$ be the difference between $\widehat{P}_i$ and P_i, and the average difference over all queries is:

$$\frac{1}{m}\sum_i (P_i - \widehat{P}_i) = \frac{1}{m}\sum_i P_i - \frac{1}{m}\sum_i \widehat{P}_i \tag{3.2}$$

where m is the number of all queries.

We first look at the actual performance part, i.e., $\frac{1}{m}\sum_i \widehat{P}_i$, in Eq. 3.2. We can consider it as an expected value over all queries:

$$E(\widehat{P}) = \sum_i \widehat{P}_i \times p(q_i) = \frac{1}{m}\sum_i \widehat{P}_i \tag{3.3}$$

where $p(q_i)$ is uniform, meaning that all queries are treated equally. A lot of efforts have been devoted to improve this expected performance. For instance, if the average precision (AP) is used as the performance metric, $\widehat{P}_i$ represents the AP for each individual query q_i and $E(\widehat{P})$ represents the mean average precision (MAP) over all queries. Note that other performance metrics can be used in Eq. 3.3.

Now let us look at the performance target part $\frac{1}{m}\sum_i P_i$ in Eq. 3.2. Let $P \equiv \frac{1}{m}\sum_i P_i$, which actually denotes the upper bound of $E(\widehat{P})$. Let the difference between the actual mean performance and target performance can be defined as the performance bias:

$$Bias(\widehat{P}) = P - E(\widehat{P}) \tag{3.4}$$

Table 3.1: Basic notations and descriptions related to query language modeling

Notation	Description
$\widehat{\theta}_{q_i}$	*estimated* query language model for q_i
θ_{q_i}	*true* query language model for query q_i
$\widehat{P}_i$	performance of an *estimated* query model $\widehat{\theta}_{q_i}$ for query q_i
P_i	performance target of the *true* query model θ_{q_i} for query q_i
$\widehat{\eta}_i$	KL-divergence between true model θ_{q_i} and *estimated* model $\widehat{\theta}_{q_i}$
η_i	KL-divergence between true model θ_{q_i} and *true* model θ_{q_i}
$\widehat{\xi}_i$	Cosine similarity between true model θ_{q_i} and *estimated* model $\widehat{\theta}_{q_i}$
ξ_i	Cosine similarity between true model θ_{q_i} and *true* model θ_{q_i}

The above $Bias(\widehat{P})$ equals to $\frac{1}{m}\sum_i(P_i - \widehat{P}_i)$ in Eq. 3.2, which considers the average difference between the actual performance $\widehat{P}_i$ and the performance target P_i over all queries. From Eq. 3.4, it turns out that the higher $E(\widehat{P})$ (i.e., the actual MAP) is, the smaller performance bias would be, for the same set of queries and the same upper bound performance P.

We now formulate the performance variance as

$$Var(\widehat{P}) = E(\widehat{P} - E(\widehat{P}))^2 \tag{3.5}$$

which represents the performance variability over different queries, and can indicate the stability of the retrieval performance. Again, in this thesis, $E(\widehat{P})$ denotes MAP and $Var(\widehat{P})$ represents the variance of average precision of all concerned queries. We can denote the variance of average precision as VAP. The smaller VAP indicates the better stability of the retrieval performance, generally meaning the better stability of the estimated query model. VAP computes the second central moment of AP, by considering the value of AP on different queries as a random variable. This is helpful to integrate VAP and MAP, the latter being the first moment of AP, into the bias-variance framework.

Now, we can add the bias and variance together, yielding

$$\begin{aligned} Bias^2(\widehat{P}) + Var(\widehat{P}) &= (E(\widehat{P}) - P)^2 + E(\widehat{P} - E(\widehat{P}))^2 \\ &= E(\widehat{P} - E(\widehat{P}) + E(\widehat{P}) - P)^2 \\ &= E(\widehat{P} - P)^2 \end{aligned} \tag{3.6}$$

This summed quantity $E(\widehat{P} - P)^2$ in Eq. 3.6 takes into account both performance bias and variance, which are related to retrieval effectiveness and stability, respectively, across all queries.

In our opinion, retrieval robustness is a combined criteria of retrieval effectiveness and stability. Both effectiveness and stability are important in evaluating the robustness of an IR system. Considering only one criteria (effectiveness or stability) is not sufficient. Thus,

the summed quantity in Eq. 3.6, which takes into account both retrieval effectiveness and stability, can be considered as a metric for the retrieval robustness. The bias-variance decomposition of $E(\widehat{P}-P)^2$ in Eq. 3.6 can naturally formulate the effectiveness-stability decomposition of retrieval robustness.

We do not argue that the overall quantity in Eq. 3.6 can cover every aspect of retrieval robustness in IR. However, it provides a decomposition perspective, which can help us understand and analyze the retrieval robustness. In addition, the bias-variance decomposition can help us analyze the tradeoff between the retrieval effectiveness and stability and then give us some clues on how to improve retrieval robustness.

3.1.3 Additional Performance Bias-Variance

In the above bias-variance formulation, the random variable is the actual performance $\widehat{P}$ which is different for different queries. Now, we are going to formulate an additional bias-variance based on the difference between the actual performance $\widehat{P}_i$ and the performance target P_i of each query q_i. First, let $\widehat{\rho}$ denote the random variable representing such a difference which can be different for different queries. Specifically, let

$$\widehat{\rho}_i = P_i - \widehat{P}_i \tag{3.7}$$

and accordingly its target $\rho_i = P_i - P_i$. Obviously, $\rho_i = 0$ for each query. Then, we can let $\rho = 0$ be the target difference for each $\widehat{\rho}_i$.

Next, we can define the bias of the random variable $\widehat{\rho}$ as:

$$Bias(\widehat{\rho}) = E(\widehat{\rho}) - \rho = E(\widehat{\rho}) \tag{3.8}$$

where $E(\widehat{\rho})$ is an expectation value over all queries:

$$E(\widehat{\rho}) = \frac{1}{m}\sum_i \widehat{\rho}_i = \frac{1}{m}\sum_i (P_i - \widehat{P}_i) \tag{3.9}$$

It turns out that $Bias(\widehat{\rho})$ equals to $\frac{1}{m}\sum_i(P_i - \widehat{P}_i)$. Recall that $Bias(\widehat{P})$ (in Eq. 3.4) also equals to $\frac{1}{m}\sum_i(P_i - \widehat{P}_i)$. Therefore, $Bias(\widehat{P})$ and $Bias(\widehat{\rho})$ are actually equivalent.

We now define the additional performance variance as

$$Var(\widehat{\rho}) = E(\widehat{\rho} - E(\widehat{\rho}))^2 \tag{3.10}$$

If P_i is a constant for every query q_i, then $Var(\widehat{\rho})$ equals to $Var(\widehat{P})$. To see this, we can

let $P_i = a$ for every query. Then,

$$\begin{aligned} Var(\widehat{\rho}) &= E(\widehat{\rho} - E(\widehat{\rho}))^2 \\ &= E(a - \widehat{P} - E(a - \widehat{P}))^2 \\ &= E(a - \widehat{P} - a + E(\widehat{P}))^2 \\ &= E(\widehat{P} - E(\widehat{P}))^2 \\ &= Var(\widehat{P}) \end{aligned} \tag{3.11}$$

Ideally, for each q_i, the performance target P_i can be the maximum performance value which is 1 (a constant) if using AP as the performance metric. However, in practice, it is unrealistic to define every P_i as a constant, because there is a system variance of performance targets in terms of hardness across different queries. In other words, for different queries q_i, P_i can be different. Given the existence of the system variance associated with P_i, $Var(\widehat{\rho})$ is different from $Var(\widehat{P})$. In $Var(\widehat{\rho})$, the random variable is $\widehat{\rho}$, rather than the actual performance $\widehat{P}$.

We will also investigate the additional performance bias-variance by proposing a regularized $\widehat{\rho}$, in order to reduce the impact of the aforementioned system variance on the bias and variance. We can first regularize the actual performance $\widehat{P_i}$ of each query q_i. Specifically, we can let

$$\widehat{P}_i' = \frac{\widehat{P}_i}{P_i} \tag{3.12}$$

where $\widehat{P}_i'$ is the regularized actual performance by considering the hardness of a query. Accordingly, the target of $\widehat{P}_i'$ is P_i', which is regularized as $\frac{P_i}{P_i} = 1$ (a constant for each query). In this manner, the system variance of the (regularized) performance target values can be eliminated. We can then define the regularized $\widehat{\rho}_i$ as:

$$\widehat{\rho}_i' = P_i' - \widehat{P}_i' = \frac{P_i - \widehat{P}_i}{P_i} \tag{3.13}$$

where $\widehat{\rho}_i'$ represents the regularized difference between the actual performance $\widehat{P}_i$ and the performance target P_i for each query q_i.

Based on the regularized performance difference $\widehat{\rho}_i'$ for every query, we can define another additional performance bias as $Bias(\widehat{\rho}')$ and variance as $Var(\widehat{\rho}')$, similarly to Eq. 3.8 and Eq. 3.10, respectively:

$$Bias(\widehat{\rho}') = E(\widehat{\rho}') - \rho' = E(\widehat{\rho}') \tag{3.14}$$

and

$$Var(\widehat{\rho}') = E(\widehat{\rho}' - E(\widehat{\rho}'))^2 \tag{3.15}$$

In the next subsections, we will present examples to show the aforementioned bias-

Table 3.2: Examples for Different Bias-Variance

System	A		B		T	
	q_1	q_2	q_1	q_2	q_1	q_2
AP	0.3	0.1	0.6	0.08	0.7	0.2
$Bias$(AP)	0.25		0.11		0	
Var(AP)	0.01		0.0646		0.0625	
$Bias^2$(AP) + Var(AP)	0.0725		0.0797		0.0625	
$\widehat{\rho}$	0.4	0.1	0.1	0.12	0	0
$Bias(\widehat{\rho})$	0.25		0.11		0	
$Var(\widehat{\rho})$	0.0225		0.0001		0	
$Bias^2(\widehat{\rho}) + Var(\widehat{\rho})$	0.0850		0.0122		0	
$\widehat{\rho}'$	0.5714	0.5	0.1429	0.6	0	0
$Bias(\widehat{\rho}')$	0.5357		0.3714		0	
$Var(\widehat{\rho}')$	0.0013		0.0522		0	
$Bias^2(\widehat{\rho}') + Var(\widehat{\rho}')$	0.2883		0.1901		0	

variance definitions and decompositions.

3.1.4 Examples of Additional Bias-Variance Definitions

Recall that in Chapter 1, we gave some examples of the performance bias-variance in Table 1.1. Here, we will describe the additional performance bias-variance using the examples in Table 3.2. Suppose that there are two queries q_1 and q_2, and for each query we use the average precision (AP) to measure/observe the retrieval performance. Assume that we have two estimation methods A and B, where A and B can correspond to the original query model and the expanded query model, respectively. Let us also assume that the target method T [3] (corresponding to the true query model) can give an upper-bound performance for every query.

For the method A, let us define

$$\widehat{\rho}_A = \text{AP}_T - \text{AP}_A \tag{3.16}$$

Therefore, $\widehat{\rho}_A$ is 0.4 (0.7-0.3) and 0.1 (0.2-0.1), for q_1 and q_2, respectively (see Table 3.2). We can see that $\widehat{\rho}_A$ represents the difference between the AP target and the AP value of method A. For method T, it turns out the ρ_T [4] is 0 for each query since $\text{AP}_T - \text{AP}_T = 0$. This also indicates that the variance of ρ_T across different queries is zero for the target method T. The definition of a constant ρ_T for each query is very important in the bias-

[3] The method T is not necessarily the ideal estimation which has the maximum AP (i.e., 1) for every query. In practice, we just assume that it has the best obtainable performance per query, among all the query model estimation methods we will analyze. The true query model, which is formulated in Section 3.2.2, can be regarded as our target method in the query expansion problem.

[4] We removed the hat of ρ_T since it is only concerned with the target AP.

variance decomposition shown in the next section. In this section, we first show the bias and variance based on the variable $\widehat{\rho}$.

The bias based on $\widehat{\rho}$ of method A can be denoted as $Bias_A(\widehat{\rho})$, which is

$$\begin{aligned} Bias_A(\widehat{\rho}) &= E(\widehat{\rho}_A) - \rho_T \\ &= \frac{1}{2}(0.4 + 0.1) - 0 = 0.25 \end{aligned}$$

where $E(\widehat{\rho}_A)$ is the averaged $\widehat{\rho}_A$ over all queries. Since ρ_T is 0, $Bias_A(\widehat{\rho})$ is actually $E(\widehat{\rho}_A)$, which represents the average difference between the method A's AP value and the AP target over all queries. Therefore, $Bias(\widehat{\rho})$ is 0.25 for method A, and is 0.11 for method B (see Table 3.2) [5].

Now, let us see the variance based on the random variable $\widehat{\rho}$. For method A,

$$\begin{aligned} Var_A(\widehat{\rho}) &= E(\widehat{\rho}_A - E(\widehat{\rho}_A))^2 \\ &= \frac{1}{2}[(0.4 - 0.25)^2 + (0.1 - 0.25)^2] = 0.0225 \end{aligned}$$

$Var_A(\widehat{\rho})$ represents the variance of the difference between the method A's AP value and the AP target across all queries.

In the similar manner as $Var_A(\widehat{\rho})$ (for method A), we can obtain $Var_B(\widehat{\rho}) = 0.0001$ (for method B). It turns out that $Var_A(\widehat{\rho})$ is bigger than $Var_B(\widehat{\rho})$. Recall that $Bias_A(\widehat{\rho})$ is also bigger than $Bias_B(\widehat{\rho})$. It turns out there is **no tradeoff** between the $Bias(\widehat{\rho})$ and $Var(\widehat{\rho})$. And, the tradeoff often does not exist in the TREC experiments for query modeling [6]. But, the bias-variance tradeoff is what we want to explore, especially in the scenario of query modeling. The original query model (method A) was expected to have a bigger bias and smaller variance, compared with the expanded query model (method B) which is more complex.

For the example of the regularized $\widehat{\rho}$ (denoted as $\widehat{\rho}'$) of the method A, we have

$$\widehat{\rho}'_A = \frac{\text{AP}_T}{\text{AP}_T} - \frac{\text{AP}_A}{\text{AP}_T} = \frac{\text{AP}_T - \text{AP}_A}{\text{AP}_T} \tag{3.17}$$

Using this regularized variable $\widehat{\rho}'$, we can define bias and variance in the similar manner as $\widehat{\rho}$ (see the previous subsection). As shown in Table 3.2 and observed in our experiments, the tradeoff between bias and variance (based on $\widehat{\rho}'$) is more likely to exist, compared with the bias and variance based on $\widehat{\rho}$. However, bias-variance tradeoff (based on AP, see Sec. 1.4.1) is the most obvious one based on our systematic TREC experiments (see Section 3.3).

[5] In Table 1.1, the bias based on AP is also 0.25 and 0.11 for method A and B, respectively. It turns out that two kinds of biases (based on AP and $\widehat{\rho}$) are equivalent.

[6] See Section 3.3 for systematic experimental results.

3.1.5 Bias-Variance Decomposition of Expected Squared Error

According to the above definition of the variable $\widehat{\rho}_A$ (see Eq. 3.16), we have

$$E(\mathrm{AP}_A - \mathrm{AP}_T)^2 = E(\widehat{\rho}_A - \rho_T)^2 \tag{3.18}$$

which is the average squared difference between the AP of the method A and the AP of the target method T. Note that we cannot directly decompose $E(\mathrm{AP}_A - \mathrm{AP}_T)^2$, since the AP target (i.e., AP_T)) can be different for different queries. Then, we look at the bias-variance decomposition of the expected squared error based on the variable $\widehat{\rho}_A$ [7]. We have an example after the following formulation.

$$\begin{aligned}
&E(\widehat{\rho}_A - \rho_T)^2 \\
&= E(\widehat{\rho}_A - E(\widehat{\rho}_A) + E(\widehat{\rho}_A) - \rho_T)^2 \\
&= E[(\widehat{\rho}_A - E(\widehat{\rho}_A)) + (E(\widehat{\rho}_A) - \rho_T)]^2 \\
&= E[(\widehat{\rho}_A - E(\widehat{\rho}_A))^2 + (E(\widehat{\rho}_A) - \rho_T)^2 + 2(\widehat{\rho}_A - E(\widehat{\rho}_A))(E(\widehat{\rho}_A) - \rho_T)] \\
&= E(\widehat{\rho}_A - E(\widehat{\rho}_A))^2 + E((E(\widehat{\rho}_A) - \rho_T)^2) + E[2(\widehat{\rho}_A - E(\widehat{\rho}_A))(E(\widehat{\rho}_A) - \rho_T)] \\
&= E(\widehat{\rho}_A - E(\widehat{\rho}_A))^2 + (E(\widehat{\rho}_A) - \rho_T)^2 + 2E(\widehat{\rho}_A - E(\widehat{\rho}_A))E(\widehat{\rho}_A) - \rho_T)
\end{aligned}$$

Since $E(\widehat{\rho}_A - E(\widehat{\rho}_A)) = 0$, we can have,

$$\begin{aligned}
&E(\widehat{\rho}_A - \rho_T)^2 \\
&= E(\widehat{\rho}_A - E(\widehat{\rho}_A))^2 + (E(\widehat{\rho}_A) - \rho_T)^2 \\
&= Var_A(\widehat{\rho}) + Bias_A^2(\widehat{\rho})
\end{aligned} \tag{3.19}$$

For example [8],

$$E(\widehat{\rho}_A - \rho_T)^2 = \frac{1}{2}[(0.4-0)^2 + (0.1-0)^2]^2 = 0.0850,$$

which equals to

$$Bias_A^2(\widehat{\rho}) + Var_A(\widehat{\rho}) = 0.25^2 + 0.0225 = 0.0850$$

In fact,

$$E(\widehat{\rho}_A - \rho_T)^2 = \frac{1}{2}[(0.4-0)^2 + (0.1-0)^2]^2 = \frac{1}{2}[(0.7-0.3)^2 + (0.2-0.1)^2]$$

Since the above expected squared error is the average squared difference between the AP

[7] We will not show the decomposition based on the regularized variable $\widehat{\rho}'_A$, since it is similar to the decomposition based on $\widehat{\rho}_A$

[8] One may ask why we need a ρ_T (0) in the above formulation. The reason is that a constant target ρ_T for each query is very important to derive the bias-variance decomposition.

of the method A and the AP of the target method T, the expected squared error is 0 for the target method T.

On the other hand, regarding the bias-variance definition (directly) based on AP, the sum of the squared bias and variance of method A is

$$\begin{aligned} & Bias_A^2(\text{AP}) + Var_A(\text{AP}) \\ &= [E(\text{AP}_T) - E(\text{AP}_A)]^2 + E(\text{AP}_A - E(\text{AP}_A))^2 \\ &= (\text{MAP}_T - \text{MAP}_A)^2 + \text{VAP}_A \end{aligned} \tag{3.20}$$

We can also have:

$$\begin{aligned} & Bias_A^2(\text{AP}) + Var_A(\text{AP}) \\ &= [E(\text{AP}_T) - E(\text{AP}_A)]^2 + E(\text{AP}_A - E(\text{AP}_A))^2 \\ &= E(\text{AP}_A - \text{MAP}_T)^2 \end{aligned} \tag{3.21}$$

where MAP_T is the MAP of the target method. This decomposition is not the decomposition of the expected squared error (as in Eq. 3.19). However, in the next section we are going to show that this decomposition is actually a simple and practical version of the decomposition of another expected squared error.

3.1.6 Further Investigation of the Expected Squared Error

In the previous sections, we assume that the target method can give a upper-bound performance for every query. The target method can correspond to the true query model in the query expansion scenario. However, the true query model (see Eq. 2.12 and Eq. 3.44) is just an estimation of the true relevance model. Recall that the ideal estimation of the true relevance model should have the maximum performance (i.e., 1 for AP) for each query.

More generally, the target method can be a *virtual* system/model that has the optimum performance, based on the best TREC result per query in all previous years. However, it is likely that the current best performance for some queries will no longer be the best in the near future. For example, solely for a specific query or query category, it is very likely that an advanced algorithm would be designed and achieve better performance. Maybe Google has already achieved better performance using their (unreported) algorithm.

If we use the expected squared error in Eq. 3.25 to measure a target method, its error will be 0. Then, it means that the target method is perfect. However, as we discussed before, it is likely that the current best performance (i.e., the upper-bound performance) for some queries will no longer be the best in the near future.

To sum up, we need to theoretically set up an upper-bound performance which is certainly above the actual possible performance, and construct an expected squared error which can reflects the error (or the room of improvement) of the target method. Since AP

is the evaluation metric we are mainly concerned with, we set up 1 as its upper-bound for each query. Then, we have an expected squared error as:

$$E(\mathrm{AP} - 1)^2 \tag{3.22}$$

where AP is a random variable, which represents the AP value of an estimation method (including the target method). The expectation value (denoted as E) is over all concerned queries.

Next, we are going to see the two kinds of decomposition of the above expected squared error, each corresponding to one variable ($\widehat{\rho}$ or AP).

Decompositions associated with $\widehat{\rho}$

Recall that we define a variable $\widehat{\rho}_A$ in Eq. 3.16. Here, for a more general definition, we remove the subscript A and define $\widehat{\rho}$ as

$$\widehat{\rho} = \mathrm{AP}_T - \mathrm{AP} \tag{3.23}$$

where AP represent AP values of a concerned method, AP_T is an estimated/empical upper bound of a target method.

We can first rewrite Eq. 3.22 as follows:

$$\begin{aligned} & E(\mathrm{AP} - 1)^2 \\ = \; & E(\mathrm{AP} - \mathrm{AP}_T + \mathrm{AP}_T - 1)^2 \\ = \; & E((\mathrm{AP} - \mathrm{AP}_T) + (\mathrm{AP}_T - 1))^2 \\ = \; & E(\mathrm{AP} - \mathrm{AP}_T)^2 + E(\mathrm{AP}_T - 1)^2 + 2E(\mathrm{AP} - \mathrm{AP}_T)(\mathrm{AP}_T - 1) \end{aligned} \tag{3.24}$$

Intuitively, the difference between AP and 1 can include two parts, which are the difference between AP and AP_T, as well as the difference between AP_T and 1. The first part is corresponding to the expected squared error between AP and AP_T:

$$\begin{aligned} & E(\mathrm{AP} - \mathrm{AP}_T)^2 \\ = \; & E(\widehat{\rho} - \rho) \\ = \; & E(\widehat{\rho} - E(\widehat{\rho}))^2 + (E(\widehat{\rho}) - \rho_T)^2 \\ = \; & Var(\widehat{\rho}) + Bias^2(\widehat{\rho}) \end{aligned} \tag{3.25}$$

which is the same as Eq. 3.19, except for that the above equation remove the subscript A.

The second part corresponds to the expected squared error between AP_T and 1:

$$\begin{aligned} & E(\mathrm{AP}_T - 1)^2 \\ = \; & Var(\mathrm{AP}_T) + (1 - \mathrm{MAP}_T)^2 \end{aligned} \tag{3.26}$$

The above equation means that even for the target method, its AP (i.e., AP_T) still has error (i.e., $E(\mathrm{AP}_T - 1)^2$). This means that there is still room to be further developed. In other words, the mean performance can be further improved and the variance can be further reduced. However, if we only consider the error $E(\mathrm{AP} - \mathrm{AP}_T)^2$ (in Eq. 3.25), the error represented in Eq. 3.26 will be totally neglected.

When we fix the practical upper bound, we can regard $Var(\mathrm{AP}_T)$ as the variance of the query difficulty. The quality $E(\mathrm{AP}_T - 1)^2$ can represent the total error of the target method.

Note that $E(\mathrm{AP} - 1)^2$ in Eq. 3.24 does not always equal to the sum of $E(\mathrm{AP} - \mathrm{AP}_T)^2$ in Eq. 3.25 and $E(\mathrm{AP}_T - 1)^2$ in Eq. 3.26. Therefore, $E(\mathrm{AP} - \mathrm{AP}_T)^2$ and $E(\mathrm{AP}_T - 1)^2$ can not be regarded as a strict decomposition of $E(\mathrm{AP} - 1)^2$.

Decomposition directly associated with AP

First, let us recall the bias and variance in Eq. 3.20 and Eq. 3.21, and re-formulate a more general bias-variance decomposition by removing the subscript A in Eq. 3.20 and Eq. 3.21:

$$\begin{aligned} & E(\mathrm{AP} - \mathrm{MAP}_T)^2 \\ &= Bias^2(\mathrm{AP}) + Var(\mathrm{AP}) \\ &= (\mathrm{MAP}_T - \mathrm{MAP})^2 + Var(\mathrm{AP}) \end{aligned} \tag{3.27}$$

We are going to illustrate that the above decomposition of $E(\mathrm{AP}_A - \mathrm{MAP}_T)^2$ is actually a simple and practical version of the decomposition of the expected squared error $E(\mathrm{AP} - 1)^2$.

To see this, we decompose Eq. 3.22 as follows:

$$\begin{aligned} & E(\mathrm{AP} - 1)^2 \\ &= E[\mathrm{AP} - E(\mathrm{AP}) + E(\mathrm{AP}) - 1]^2 \\ &= E(\mathrm{AP} - E(\mathrm{AP}))^2 + [E(\mathrm{AP}) - 1]^2 \\ &= Var(\mathrm{AP}) + (\mathrm{MAP} - 1)^2 \\ &= (1 - \mathrm{MAP})^2 + Var(\mathrm{AP}) \end{aligned} \tag{3.28}$$

It turns out the variance in Eq. 3.27 and Eq. 3.28 are the same, which is $Var(\mathrm{AP})$. The term $(1 - \mathrm{MAP})^2$ in Eq. 3.28 has the same trend with $Bias^2(\mathrm{AP})$ (i.e., $(\mathrm{MAP}_T - \mathrm{MAP})^2$), provided that MAP_T is the upper bound of the MAP of all concerned methods [9]. The above observations indicate that decomposition of the error represented in Eq. 3.27 is a simple version of the decomposition in Eq. 3.28.

When we fix a target method with upper-bound performance per query, Eq. 3.28 shows

[9] $(1 - \mathrm{MAP})^2$ can be regarded as a squared bias of AP with regard to a maximum performance 1. We did not call it bias, in order to avoid the conflict of the previous bias with respect to MAP_T shown in Eq. 3.21 and Eq. 3.27

that the target method still has an error $(1 - \text{MAP}_T)^2 + Var(\text{AP}_T)$, which means that it still has room to be further developed.

If we calculate the error of the target method using Eq. 3.27, it shows that it has an error $(\text{MAP}_T - \text{MAP}_T)^2 + Var(\text{AP}_\text{T}) = Var(\text{AP}_\text{T})$, which means that we are satisfied with the mean performance (i.e., MAP) of the target method, but the performance is still varied across queries. In other words, for some hard queries, its performance is still low and an advanced algorithm is needed to be designed to improve the performance for such hard queries.

3.1.7 Comparison between different Bias-Variance Decomposition

Now we mainly discuss the difference between the bias-variance formulation based on $\widehat{\rho}$ (in Eq. 3.25) and the bias-variance formulation based on AP (in Eq. 3.27).

First, the bias-variance based on $\widehat{\rho}$ corresponds to the assumption that the target method is perfect and has no error. On the other hand, the bias-variance based on AP (in Eq. 3.27) indicates that the target method still has error [10], which is the variance of its AP across queries. This variance reflects the variance of query difficulty across queries.

Second, according to our systematic experiments, the bias and variance based on $\widehat{\rho}$ often does not have tradeoff, while the bias and variance based on AP can have more clear tradeoff. The bias-variance based on $\widehat{\rho}$ tends to neglect the variance. An extreme case is that the target method has zero variance even though its AP can also varies across queries. Note that the variance of AP of a target method can be larger than the variance of AP of a test method (i.e., A or B in Table 3.2). But this does not mean that the error of the target method is bigger than the error of the test method. It is because that the variance is just one part of the error, and the other part is the bias [11]. Our systematic experimental results show that the $Bias^2(\text{AP}) + Var(\text{AP})$ of the target method can have the smallest total error.

Third, we are going to show that the decomposition of $E(\text{AP} - \text{MAP}_T)^2$ in Eq. 3.27 can be rewrote to include the bias-variance decomposition based on $\widehat{\rho}$ in its formulation

[10] In query expansion, the target method is the true query model, which is not the ideal estimation of the true relevance model

[11] This reminds us that one main difference between bias-variance analysis and mean-variance analysis is that the former treat both bias and variance as a kind of error, while the latter only often treat variance as a risk/error.

[12] :

$$\begin{aligned} & Bias^2(\text{AP}) + Var(\text{AP}) \\ &= E(\text{AP} - \text{MAP}_T)^2 \\ &= E(\text{AP} - \text{AP}_T + \text{AP}_T - \text{MAP}_T)^2 \\ &= E(\text{AP} - \text{AP}_T)^2 + 2E(\text{AP} - \text{AP}_T)(\text{AP}_T - \text{MAP}_T) + E(\text{AP}_T - \text{MAP}_T)^2 \\ &= Var(\widehat{\rho}) + Bias^2(\widehat{\rho}) + 2E(\text{AP} - \text{AP}_T)(\text{AP}_T - \text{MAP}_T) + Var(\text{AP}_T) \end{aligned} \tag{3.29}$$

The above decomposition also considers $Var(\text{AP}_T)$ which is the variance of AP of the target method and can reflect the query difficulty across queries.

Fourth, the bias-variance based on $\widehat{\rho}$ needs to assume that the upper-bound AP for every query is available. On the other hand, the bias-variance based on AP (in Eq. 3.27) just assumes that we have a target MAP (i.e., MAP_T). Actually, it is even not necessary to have a target method. We just give the bias a MAP target that the user can be satisfied. In addition, the variance of AP can be calculated for any method without knowing the upper bound performance per query. The above discussion means that the bias-variance based on AP (in Eq. 3.27) can be more practical and more flexible.

Finally, the performance variance $Var(\text{AP})$ indicates the stability of actual retrieval performance AP across queries. On the other hand, the additional performance variances $Var(\widehat{\rho})$ and $Var(\widehat{\rho}')$ indicate the stability of performance difference ($\text{AP}_T - \text{AP}$) and regularized performance difference, respectively, across queries. In different scenarios, the best choice of variance to represent the stability might be different. In our scenario, the stability reflected by $Var(\text{AP})$ is the one we mentioned in the introduction, and we will pay more attention to this notion of stability in the rest of this thesis. $Var(\text{AP})$ is defined solely based on the actual performance. It also can be combined with the performance bias $Bias(\text{AP})$ and form a compound indicator of retrieval robustness (see the discussions below Eq. 3.6).

3.1.8 Estimation Bias and Variance

Now, we are going to formulate the estimation bias-variance, in order to *directly* investigate the estimation error (or quality) of an estimated query model with respect to the true query model. The difference between different kinds of bias-variance will be discussed later in Section 3.1.9.

The estimation error or quality can be based on the divergence or similarity between the estimated query model $\widehat{\theta}_q$ and the true one θ_q. We will use both metrics to formulate the estimation bias and variance.

[12] Recall the bias-variance decomposition based on $\widehat{\rho}$ is just one part of the decomposition of $E(\text{AP}-1)^2$ (see Eq. 3.24 and Eq. 3.25), while the bias-variance decomposition based on AP (see Eq. 3.27) is a simple version of the decomposition of $E(\text{AP}-1)^2$.

Bias-Variance based on Divergence between $\widehat{\theta}_{q_i}$ and θ_{q_i}

For each query q_i, we denote the true query model as θ_{q_i} and any estimated query model as $\widehat{\theta}_{q_i}$. The specific formulation of the estimated and true query models are given in the next section. Here, we focus on the main formulation of bias and variance in the estimation process.

For each individual query q_i, the estimation error can be represented by the KL-divergence [13] between the estimated query model and the true query model:

$$\widehat{\eta}_i = D(\widehat{\theta}_{q_i}|\theta_{q_i}) \tag{3.30}$$

Then, the mean estimation error over all concerned queries can be defined as an expected value:

$$E(\widehat{\eta}) = \sum_i \widehat{\eta}_i \times p(q_i) = \frac{1}{m}\sum_i D(\widehat{\theta}_{q_i}|\theta_{q_i}) \tag{3.31}$$

where m denotes the number of queries and $p(q_i)$ is assumed to be uniform, meaning that all queries are treated equally. The expected estimation error in Eq. 3.31 represents the *bias* of the estimation.

More strictly, for each query q_i, we can consider $\widehat{\eta}_i$ to be an estimated value. The true value can be denoted as η_i, which corresponds to the case when the estimated query model $\widehat{\theta}_{q_i}$ (in Eq. 3.30) is the true query model θ_{q_i}. It is obvious that $\eta_i = 0$ for each query as $D(\theta_{q_i}|\theta_{q_i}) = 0$. Therefore, we can denote each η_i as η (=0), which is a constant for each query. Now, we have

$$Bias(\widehat{\eta}) = E(\widehat{\eta}) - \eta \tag{3.32}$$

which is the estimation bias. It equals to the expected value in Eq. 3.31. The smaller bias indicates the smaller expected estimation error, implying the higher expected estimation quality.

For the estimation variance, we can have

$$Var(\widehat{\eta}) = E(\widehat{\eta} - E(\widehat{\eta}))^2 \tag{3.33}$$

which represents the variance of the estimation error for different individual queries (i.e., q_i's). The estimation variance represents estimation stability.

By adding the squared bias and variance, we get

$$\begin{aligned} Bias^2(\widehat{\eta}) + Var(\widehat{\eta}) &= (E(\widehat{\eta}) - \eta)^2 + E(\widehat{\eta} - E(\widehat{\eta}))^2 \\ &= E(\widehat{\eta} - E(\widehat{\eta}) + E(\widehat{\eta}) - \eta)^2 \\ &= E(\widehat{\eta} - \eta)^2 \end{aligned} \tag{3.34}$$

[13] Other Divergence measures (e.g., JS-divergence) could be used in Eq. 3.30.

which can represent the total estimation error.

In addition to KL-divergence, we will adopt another divergence measurement, i.e., JS-divergence, in the formation of estimation bias-variance. Specifically,

$$\widehat{\eta}'_i = JSD(\widehat{\theta}_{q_i}|\theta_{q_i}) = \frac{1}{2}[D(\widehat{\theta}_{q_i}|\theta_{q_i}) + D(\theta_{q_i}|\widehat{\theta}_{q_i})] \tag{3.35}$$

Based on the JS-divergence $\widehat{\eta}'_i$ for every query, we can formulate $Bias(\widehat{\eta}')$ and $Var(\widehat{\eta}')$, in the similar manner to Eq. 3.32 and Eq. 3.33, respectively. JS-divergence is the symmetrized version of KL-divergence. The range of JS-divergence is different from that of KL-divergence, where the former values are in [0,1] and the latter values are in [0,+∞]. We will analyze and evaluate the estimation bias-variance based on both divergence metrics in later sections.

Bias-Variance based on Similarity between $\widehat{\theta}_{q_i}$ and θ_{q_i}

The quality of the estimated query model $\widehat{\theta}_{q_i}$ can also be reflected by the similarity between $\widehat{\theta}_{q_i}$ and the true query model θ_{q_i}. We use Cosine similarity [14], a typical similarity measure in IR. Now, let

$$\widehat{\xi}_i = Sim(\widehat{\theta}_{q_i}, \theta_{q_i}) \tag{3.36}$$

where $\widehat{\xi}_i$ denotes the Cosine similarity between $\widehat{\theta}_{q_i}$ and θ_{q_i}.

We then denote the true similarity value as $\xi = \xi_i = Sim(\theta_{q_i}, \theta_{q_i})$ which corresponds to the case when the estimated query model $\widehat{\theta}_{q_i}$ (in Eq.3.36) is the true query model θ_{q_i}. In order to improve the overall retrieval effectiveness, it is natural to aim at making the expectation $E(\widehat{\xi})$ over all queries approach the true similarity value ξ. Then, the bias of the estimation can be formulated as:

$$Bias(\widehat{\xi}) = \xi - E(\widehat{\xi}) \tag{3.37}$$

The smaller bias here implies the higher expected estimation quality. Note that the true similarity value ξ is 1 as $Sim(\theta_{q_i}, \theta_{q_i}) = 1$.

On the other hand, to improve the stability, it is necessary to reduce the variance

$$Var(\widehat{\xi}) = E(\widehat{\xi} - E(\widehat{\xi}))^2 \tag{3.38}$$

which represents the variance of estimation qualities over different queries. The smaller variance is expected to represent the better estimation stability.

[14] Other similarity measures can also be adopted.

By adding the squared bias and variance, we get

$$\begin{aligned} Bias^2(\widehat{\xi}) + Var(\widehat{\xi}) &= (E(\widehat{\xi}) - \xi)^2 + E(\widehat{\xi} - E(\widehat{\xi}))^2 \\ &= E(\widehat{\xi} - \xi)^2 \end{aligned} \tag{3.39}$$

which can indicate the overall estimation quality.

In the above estimation bias-variance formulation, the smaller bias means that the estimated query model is closer to the true one in the expectation sense, while the smaller variance generally indicates that the qualities of the estimated query models are more stable across all queries. The sum of bias and variance reflects the overall estimation error or quality.

In summary, in this section, we have formulated the performance bias-variance and estimation bias-variance. Next, we will analyze the bias-variance tradeoff for various query language model estimation methods.

3.1.9 Difference between Performance Bias-Variance and Estimation Bias-Variance

Performance bias-variance is directly related to the retrieval performance. The basic variable is $\widehat{P}$ in the performance bias-variance. Before evaluating the performance, a ranking should be obtained based on divergence/similarity values between the estimated query model and each document model. Given a query q_i, if we use KL-divergence as the ranking function, we should compute $D(\widehat{\theta}_{q_i}|\theta_d)$ as the ranking score for each document d in the collection, where document d can be relevant or non-relevant. The KL-divergence values between query and document models, however, are not directly used in the computation of the performance bias-variance.

Estimation bias-variance is directly related to the estimation quality with respect to the true query model. The basic variable is the divergence value $\widehat{\eta}$ or the similarity value $\widehat{\xi}$ between estimated and true query models. Given a query q_i, in order to compute the KL-divergence-based estimation bias-variance, we only need to compute one quantity, i.e., the divergence $D(\widehat{\theta}_{q_i}|\theta_{q_i})$, where θ_{q_i} is the true query model generated only from the truly relevant documents.

Therefore, the performance bias-variance and estimation bias-variance are different and can have different trends. The degree of such difference may vary for different estimated query models or across different collections. Indeed, the two kinds of bias-variance can also have some similar trends for certain query models or across some collections.

3.2 Bias-Variance Analysis of Query Language Models

In this section, we first introduce some background knowledge of the language modeling (LM) approach in Section 3.2.1, in order to see the role of query language model in LM. In Section 3.2.2, we formulate the true query model, as well as present and analyze various estimated query models which reflect different factors that can affect the model estimation. Based on our analysis, we then summarize a number of hypotheses on bias-variance tradeoff and on reducing bias and variance simultaneously. These hypotheses will be tested in our empirical evaluation (Section 3.3).

3.2.1 Background of Language Modeling

The query-likelihood (QL) approach (Ponte & Croft 1998, Zhai & Lafferty 2001), which is a standard LM approach and uses the original query representation, can be formulated as:

$$p(q_i|\theta_d) = \prod_{j=1}^{m_{q_i}} p(q_{i,j}|\theta_d) \tag{3.40}$$

where $p(q_i|\theta_d)$ is the query-likelihood, q_i ($q_{i,1}q_{i,2}\cdots q_{i,m_{q_i}}$) is the given original query, m_{q_i} is q_i's length, and θ_d is the smoothed language model for a document d. The query likelihood tries to estimate the probability that this document d generates this query q_i.

Relevance Model (RM) (Lavrenko & Croft 2001), as a relevance-based language model and a typical query expansion method, is used to estimate an expanded query language model based on relevance feedback:

$$p(w|\widehat{\theta}_{q_i}^{(f)}) = \sum_{d\in D} p(w|\theta_d)\frac{p(q_i|\theta_d)p(\theta_d)}{\sum_{d'\in D} p(q_i|\theta_{d'})p(\theta_{d'})} \tag{3.41}$$

where $\widehat{\theta}_{q_i}^{(f)}$ represents the feedback-based expanded query model, $p(\theta_d)$ represents the prior probability of document d, D denotes a set of feedback documents that generate the expanded query model, $p(q_i|\theta_d)$ computes the query-likelihood (QL) score, and the normalized QL score serves as the document weight:

$$S_{q_i}(d) = \frac{p(q_i|\theta_d)p(\theta_d)}{\sum_{d'\in D} p(q_i|\theta_{d'})p(\theta_{d'})} \tag{3.42}$$

In practice, the documents in D are pseudo-relevant feedback documents, i.e., top-ranked documents retrieved by the QL model (as the first-round retrieval method). After the query expansion, the ranking is based on the second-round retrieval using the expanded query model.

For any estimated query model, the document retrieval can be based on the negative KL-Divergence (Lafferty & Zhai 2001) between the estimated query language model $\widehat{\theta}_{q_i}$

and document language model θ_d:

$$-D(\widehat{\theta}_{q_i}|\theta_d) = -H(\widehat{\theta}_{q_i}, \theta_d) + H(\widehat{\theta}_{q_i}) \tag{3.43}$$

where $H(\widehat{\theta}_{q_i}, \theta_d)$ is the cross entropy between $\widehat{\theta}_{q_i}$ and θ_d, and $H(\widehat{\theta}_{q_i})$ is the entropy of the $\widehat{\theta}_{q_i}$.

According to the deviation in (Lafferty & Zhai 2001, Ogilvie & Callan 2002), if the original query with a maximum-likelihood estimator is used as the estimated query model, the negative KL-divergence is *rank-equivalent* to the query-likelihood approach. In this sense, the original query model (denoted as $\widehat{\theta}_{q_i}^{(o)}$) can be considered as the query model implicitly used in Eq. 3.40, which formulates the query-likelihood (QL) approach. The original query model can also be combined with the expanded query model by RM, and the combined query model is called RM3 in (Abdul-Jaleel et al. 2004).

In summary, given an estimated query language model, the negative KL-divergence in Eq. 3.43 can be used to rank documents and the corresponding retrieval performance (e.g., AP) is usually used to indicate the estimation quality. Each kind of estimated query model can be regarded as one estimation method for the query language model [15].

.

3.2.2 Analyzing Query Language Models

True Query Model

Assuming that the true information need can be reflected or represented by the truly relevant documents, the true query language model should be generated from the truly relevant documents (see also the motivation behind the true query model in the literature review):

$$p(w|\theta_{q_i}) = \sum_{d \in D_R} p(w|\theta_d) \frac{1}{|D_R|} \tag{3.44}$$

where θ_{q_i} represents the true query model, D_R denotes the set of truly relevant documents in the PRF document set D, and $|D_R|$ is the number of documents in D_R. Our experiments also demonstrate its best performance over all the query models evaluated in this thesis.

We do not combine the query model in Eq. 3.44 with the original query model, in order to avoid different possible true query models with different combination parameters. In other words, this true query model is free of adjustable parameters. We need to have a fixed true query model and then study the estimation bias and variance of any estimated query model with respect to this "true" query model.

The true query model in Eq. 3.44 is actually based on the framework of Relevance Model (RM, see Eq. 3.41), but uses the true relevance information (i.e., relevance judge-

[15] In Table 3.1, each query model is associated with a specific query q_i. When we mention query model without specifying any query, it generally refers to an estimation method of query model.

ments) to indicate the document weight in query model estimation. Assume that the value of relevance for each document is binary. The weights of all documents in the set D_R in Eq. 3.44 are the same due to the fact that they have the same relevance judgements (i.e., 1). For non-relevant documents, since their relevant judgements are 0, their weights are 0, meaning that they are excluded from generating the query model.

Accordingly, the differences between the true query model in Eq. 3.44 and the estimated query model in Eq. 3.41 are: 1) The true query model is derived from truly relevant documents D_R, while the estimated one in Eq. 3.41 is derived from pseudo-relevance documents D; 2) For the truly relevant documents, the document weights in the true query model are more smooth than those in RM, since the document weights are the same in the true query model (see Eq. 3.44) while the document weights are normalized QL scores in RM (see Eq. 3.41).

We do not argue that the query model in Eq. 3.44 is the only true one for any framework of query model estimation in the literature. It is based on the RM framework using true relevance information (e.g., relevance judgements). For any other frameworks, we think that the true query model should also adopt the true relevance information.

Factors Affecting Bias and Variance

We first describe various factors that have an influence on the query model estimation. First, the choice to use original query model or expanded query model would result in different kinds of estimated query models. Second, we consider different combinations (with different combination coefficients) of the original and expanded query models. Third, the change of document weight (in Eq. 3.42) in RM can lead to different estimation for the query language model. At last, it is important whether or not we have part of true relevance information, e.g., relevance judgements, in building the expanded query models in Eq. 3.41. Note that in this thesis, our main focus is on the pseudo-relevance feedback and its upper bound, the relevance information we consider is the relevance judgements for only PRF documents, rather than all documents.

The aforementioned factors actually corresponds to the factors that can affect the bias and variance. In Section 3.1.1, we have mentioned three factors. They are model complexity, model design, and training data size. Regarding query model estimation, the difference between original model and expanded model is related to the model complexity. The expanded query model is often more complex in the sense that: 1) it adopts additional assumptions (Lavrenko & Croft 2001), e.g., it assumes that the top-ranked documents are relevant; 2) it often involves more parameters, e.g., the number of expanded query terms or the number of feedback documents. The combination strategy and document weight issue is related to the model design. The use of part of true relevance information can be somewhat considered as use of training data. We emphasize that we do not incorporate any machine learning algorithms (e.g., regression or classification) in our study. Compared

with model design, the use of training data has a more direct and bigger impact on the simultaneous reduction of bias and variance.

We now briefly mention different estimated query models for which we will analyze the bias and variance. These models [16] include: 1) original query model and expanded query model; 2) combined query model by original and expanded query models; 3) expanded query model with smoothed document weights for the feedback documents; 4) expanded query model with true relevance information, e.g., some known non-relevant documents. 5) expanded query model with true relevance information and smoothed document weights.

Note that even for the same query model, the analysis could be different for different kinds of bias and variance, different parameters, or different test collections. We are going to use not only the general principles of bias-variance, but also IR knowledge to analyze why bias-variance occurs and when bias and/or can be reduced. Note, that for the estimation bias-variance, since the KL-divergence is a widely-used metric to measure the divergence between two language models, we will mainly analyze the KL-divergence-based estimation bias-variance.

Original and Expanded Query Models

First, we denote $\widehat{\theta}_{q_i}^{(o)}$ as the original query language model, which is a maximum likelihood estimate of the original query term representation. $\widehat{\theta}_{q_i}^{(f)}$ in RM (see Eq. 3.41) represents a feedback-based expanded query model.

The expanded query model can usually outperform the original one in terms of the retrieval effectiveness over all queries. As a result, the performance bias of the expanded query model will be smaller than that of the original one. However, for some individual queries, the inclusion of non-relevant feedback documents in query expansion can hurt the performance. Intuitively, a poor initial ranking (by original query) would include many non-relevant feedback documents that are mis-ranked highly. Therefore, for those queries with poor initial performance, query expansion is more likely to hurt the performance, than those queries with better initial performance. A possible consequence after query expansion is that a poor initial performance would become even worse, while a better initial performance would become even better. This can result in the performance variance of the expanded query model being bigger than that of the original one.

For the estimation bias-variance, recall that it is directly related to the divergence/similarity between the estimated query model and the true query model (see Section 3.1.9). The original query model $\widehat{\theta}_{q_i}^{(o)}$ is very sparse, in the sense that it only contains the original query terms. On the other hand, the true query model θ_{q_i} (in Eq. 3.44) and the expanded query model $\widehat{\theta}_{q_i}^{(f)}$ by RM (in Eq. 3.41) do not have such a sparsity problem since they are

[16] Since our focus is the bias-variance analysis, we may only adopt some basic methods or some simple versions of concerned models. This can help us reduce the number of parameters in the retrieval models and it is more feasible to adjust no more than one parameter (if possible) to observe the trends of the changing bias and variance.

generated from a set of documents. Due to the range of KL-divergence in [0,+∞] and the sparsity of the original query model, the scale of $D(\widehat{\theta}_{q_i}^{(o)}|\theta_{q_i})$ and the scale of $D(\widehat{\theta}_{q_i}^{(f)}|\theta_{q_i})$ are quite different – the former values are often much larger than latter values. As a result, the estimation bias (based on KL-divergence) of the original query model $\widehat{\theta}_{q_i}^{(o)}$ will often be much bigger than the expanded model $\widehat{\theta}_{q_i}^{(f)}$. In addition, due to the aforementioned big scale difference, the KL-divergence-based estimation variance of the original query model can also be bigger than that of the expanded model. To sum up, in KL-divergence-based estimation bias-variance, the expanded query model often has smaller estimation bias, and can also have smaller estimation variance, compared with the original query model.

The trend of estimation bias-variance can be different when we use JS-divergence and Cosine similarity. Their ranges are [0,1] [17], which can be thought of as a normalized range of [0, +∞]. Hence, it is less likely for them to have the big scale difference like KL-divergence. The range [0,1] is also the same as the range of retrieval performance (e.g., Average Precision (AP) or Precision). Therefore, it is more likely that the bias-variance tradeoff can occur in the estimation bias-variance using JS-divergence or Cosine similarity. We will report detailed observations in the experiments.

Combination between Original and Expanded Query Models

The combination between original and expanded query models was studied in (Abdul-Jaleel et al. 2004, Tao & Zhai 2006, Li 2008, Lv & Zhai 2009). Basically, the combination can be formulated as

$$\widehat{\theta}_{q_i}^{(c)} = \lambda\widehat{\theta}_{q_i}^{(o)} + (1-\lambda)\widehat{\theta}_{q_i}^{(f)} \tag{3.45}$$

where $\widehat{\theta}_{q_i}^{(c)}$ is the combined query model, λ is the combination coefficient of the original query $\widehat{\theta}_{q_i}^{(o)}$, and $1-\lambda$ is the coefficient of the feedback-based expanded query model $\widehat{\theta}_{q_i}^{(f)}$. The combined query model in Eq. 3.45 is often referred as RM3 (Abdul-Jaleel et al. 2004).

In Section 3.1.1, we mentioned that the *combination* method may reduce the bias and variance simultaneously. Therefore, it is expected that the *combined* query model $\widehat{\theta}_{q_i}^{(c)}$ could reduce bias and variance simultaneously, if a proper combination coefficient λ is used. Here, we will investigate how the combined query model can reduce the bias and/or variance, for different kinds of bias-variance formulation.

For performance bias-variance, as discussed previously, one reason why the expanded query model has larger variance is that, for some queries, the performance can be hurt after query expansion when non-relevant terms are brought into query models. One solution can be to combine it with the original query model, which can boost the weights of original query terms while reducing the influence of non-relevant terms in the expanded query model. This can actually prevent the query drifting from the underlying information

[17] 0 and 1 are corresponding to the minimum value and maximum value of JS-divergence and Cosine similarity

need (Zighelnic & Kurland 2008). If the downside performance can be prevented, this could reduce the variance of the expanded query model. On the other hand, the bias can also be reduced if the retrieval performance on average can be improved, given appropriate combination parameters. To sum up, the combined query model with a proper combination coefficient is expected to reduce both bias and variance, which balances the advantages and disadvantages of original and expanded query models.

We observe that a small λ (e.g., 0.1), which is close to 0 but not 0, can be a proper coefficient that reduces bias and variance simultaneously. This is because the combination with the original query using a small λ can adjust the probability of original query terms in the expanded query model.

However, when λ is becoming bigger, the original query terms will tend to dominate the whole term distribution of the combined query model, since the original query terms have very big probability values in the original query model due to its sparsity. In this case, the combined query model will move towards the original query model which has bigger performance bias but smaller performance variance (see Section 3.2.2). Therefore, the trend along with the increasing λ is that the performance bias will be increasing and the performance variance will drop. This means that a performance bias-variance tradeoff will occur.

With regard to the estimation bias-variance, when λ is close to 0 but not 0, it is likely that the bias and variance can be reduced simultaneously. A proper combination can adjust the probability of original query terms in the expanded query model. When λ is approaching 1, the combined query model is getting close to the original query model and will suffer from the sparsity problem like in the original query model. When bias and variance are based on KL-divergence, this can lead to not only the increasing bias but also the increasing variance, as we discussed previously in the analysis of estimation bias-variance for the original and expanded query models.

For the estimation bias-variance using other metrics (e.g., JS-divergence or Cosine similarity), it is more likely that the estimation variance can be reduced and then the bias-variance tradeoff can occur. The range of them is different from the range of KL-divergence. The range of JS-divergence or Cosine similarity is [0,1], which is the same as the range of retrieval performance (e.g., Average Precision (AP) or Precision).

Expanded Query Model with Smoothed Document Weights

Recall that one difference between the true query model in Eq. 3.44 and the estimated query model in Eq. 3.41 is that: for relevant documents, the document weights in the true query model are more smooth than those in the estimated query model by RM. In the true query model, the document weights are the same, leading to the most smooth document weights. For the estimated query model without true relevance information, such uniform document weights may not work. However, it has been shown that properly

smoothing the document weights can improve the effectiveness of feedback-based query expansion (Zhang, Song, Zhao & Hou 2010, Zhang, Song, Wang, Zhao & Hou 2011). We think that it is worthwhile to investigate the bias-variance of the expanded query model with smoothed document weights.

We adopt a simple document weight smoothing method (Zhang, Song, Wang, Zhao & Hou 2011), which can be formulated as:

$$\widetilde{S_{q_i}}(d) = \frac{[S_{q_i}(d)]^{\frac{1}{s}}}{\sum_{d' \in D}[S_{q_i}(d')]^{\frac{1}{s}}} \tag{3.46}$$

where $\widetilde{S_{q_i}}(d)$ is the smoothed document weight, $S_{q_i}(d)$ is the original document weight, and $s(s > 0)$ is a parameter that controls the smooth degree of document weights. When $s = 1$, the document weights are unchanged. The larger the s is, the greater degree of the smoothing would be. For example, assuming the original weights are 0.6250 and 0.3750 for d_1 and d_2, and the parameter s is 3, then the smoothed document weights are 0.5425 and 0.4575, which become more smooth.

Using the smoothed document weights, the estimated query model can be formulated as:

$$p(w|\widehat{\theta}_{q_i}^{(s)}) = \sum_{d \in D} p(w|\theta_d)\widetilde{S}_{q_i}(d) \tag{3.47}$$

where $\theta_{q_i}^{(s)}$ can be referred to as smoothed query model which is the expanded query model with smoothed document weights $\widetilde{S}_{q_i}(d)$ (see Eq. 3.46).

The above smoothing method can improve the document weight smoothness among relevant documents in the pseudo-relevant feedback (PRF) document set. As we discussed in the true query model in Eq. 3.44, such smoothness of relevant documents is important because they have the same relevance judgements. The improved smoothness can also broaden the topic coverage of the expanded query, in order to prevent too many weights on the topics represented in topmost documents which might be non-relevant.

On the other hand, smoothing may affect the discriminativity between the relevant documents and non-relevant document in the PRF document set. For instance, if too much smoothing is imposed and the weights of every PRF documents are the same, no documents will have discriminative weights, even for the relevant ones. Therefore, a moderate smoothing (corresponding to a moderate smoothing parameter s) is needed to preserve the discriminativity of the relevant documents against the non-relevant ones to some extent.

Smoothing the weights of feedback documents may not help improve the query expansion for any queries. In other words, for some queries it may help, but for others it may not. Specifically, when there are many relevant documents ranked top in the initial ranking, a moderate smoothing may improve the performance of query expansion, since

the document weight smoothness among relevant documents can be improved and the aforementioned discriminativity can not be affected much. On the other hand, if there are many non-relevant feedback documents, the smoothing might hurt the query expansion because it may depress the original query terms but boost terms in the non-relevant documents [18]. Recall that in Section 3.2.2, we mentioned that for a query with a better initial ranking (with more relevant feedback documents), the query expansion performance would also be better. Based on the above discussion, give a better initial ranking, the smoothing can be more likely to further improve the query expansion performance.

Now, let us analyze the performance bias and variance, which can be different for different set of queries or on different test collections. According to the above discussions, if the initial ranking of one set of queries is better (i.e., with bigger MAP), it is more likely that the smoothed document weights could further improve the overall effectiveness of query expansion, reducing the performance bias . On the other hand, even if the mean performance of this set of queries is improved, the performance of some individual queries (with relatively poor initial ranking) can be hurt or can not be improved. This would cause the instability of retrieval performance across queries and increase the performance variance, which then results in a performance bias-variance tradeoff. Note that the performance of initial ranking can be also instable/fluctuated across different queries. The more fluctuated performance (i.e., with larger VAP) of initial ranking can cause the more fluctuated smoothing effect on improving performance of query expansion.

The document weight smoothing can play a bigger role in reducing the estimation bias, than in reducing the performance bias. This is because the estimation bias directly computes the mean estimation error of the estimated query model with respect to the true query model. In the true query model, the smoothness of relevant documents is very important and only the relevant documents are involved (see Eq. 3.44). The smoothing method can improve the smoothness among relevant feedback documents in deriving the estimated query model, which makes the estimated query model closer to the true one. Therefore, it is expected that the document weight smoothing can reduce the estimation bias, i.e., the mean estimation error.

The smoothing effect on reducing the estimation error can be different for different individual queries. For the query with more relevant feedback documents, smoothing can have a bigger effect. On the other hand, if most of feedback documents are non-relevant, the document weight smoothing may have small effect. Therefore, for a set of queries which have a bigger performance fluctuation of the initial ranking, it is more likely that the document weight smoothing can increase the estimation variance of query expansion.

[18] A larger initial document weight generally means that the original query terms have higher probability/importance in this document. Smoothing document weights reduces the larger document weight and improve the smaller document weight. This can depress the original query terms in the query model. In addition, smoothing can boost a lot of terms in the non-relevant documents, if there are many non-relevant feedback documents.

On the other hand, for a set of queries which have smaller performance fluctuation of the initial ranking, it is more likely that the estimation variance of query expansion can be reduced. It turns out that the estimation bias and variance is likely to be reduced simultaneously.

Expanded Query Model with Available Non-Relevant Data

One of the reasons for the stability problem of query expansion is that the expanded query model is often generated from a mixture of relevant and non-relevant documents. As a result, the expanded query term distribution is actually a mixture distribution of relevant terms and non-relevant ones (Zhang, Hou & Song 2009)). It is argued, that the retrieval performance can be improved if one can remove the non-relevant distribution from the mixture distribution (Zhang et al. 2009)). In accordance to the assumption in (Zhang et al. 2009), we assume that part of non-relevance information is known. Specifically, we assume that a certain ratio (denoted as parameter r_n) of non-relevant documents is known and then we derive an expanded query model based on RM with part of known non-relevant documents (denoted as D_N) removed:

$$p(w|\widehat{\theta}_{q_i}^{(-n)}) = \sum_{d \in D - D_N} p(w|\theta_d) S_{q_i}(d) \tag{3.48}$$

where $\widehat{\theta}_{q_i}^{(-n)}$ is the estimated query model, $D - D_N$ is the set of remaining documents, and $S_{q_i}(d)$ is the original document weight computed by the normalized QL score (see Eq. 3.42). Note, that the non-relevant documents are selected from top to down in the initial ranking of feedback documents, since the top non-relevant documents with bigger document weights have more influence on the query expansion.

As the non-relevance parameter r_n increases, the more non-relevant documents can be removed from the pseudo-relevant feedback (PRF) documents, meaning the PRF documents are *purer* to be truly relevant. It also means that we have more relevance judgements as r_n increases. It is expected that this method can improve both the effectiveness and stability of the feedback-based query expansion. In the language of bias-variance analysis, it is expected that as the parameter r_n increases, bias and variance can be reduced simultaneously.

Note, that one uncertainty lies in the reduction of performance variance for different set of queries or on different test collections. If there are many queries which have too many non-relevant feedback documents, after removing some non-relevant ones, most remaining documents could be still non-relevant. Therefore, the room for improving the retrieval performance by removing some non-relevant ones would be very small. For some other queries, the non-relevant documents are not too many and then the performance improvement can be bigger. A consequence is that the performance variance will be in-

creased. For example, assume that the query expansion performance of two queries are $\widehat{P}_1 = 0.1$ (for q_1) and $\widehat{P}_2 = 0.4$ (for q_2). After removing the non-relevant documents for query expansion, The performance of both queries increase to $\widehat{P}_1 = 0.12$ and $\widehat{P}_2 = 0.6$. It shows that the performance bias is reduced (as MAP is increased), but the performance variance is increased. In this case, it leads to a bias-variance tradeoff.

The estimation bias and variance are more likely to be reduced simultaneously, compared with the performance bias and variance. This is because the estimated query model in Eq. 3.48 is designed to get closer to the true query model in Eq. 3.44, as the r_n increases and more non-relevant documents are removed in the query model estimation. Note that the estimation bias and variance only relate to the divergence/similarity between the estimated query model and the true query model.

Expanded Query Model with Document Weight Smoothing and Non-Relevant Data

Now, let us consider the ideas of using both relevance information (see Eq. 3.48) and document weight smoothing (see Eq. 3.47). We then come up with the estimated query model as follows.

$$p(w|\widehat{\theta}_{q_i}^{(-ns)}) = \sum_{d \in D - D_N} p(w|\theta_d)\widehat{S}_{q_i}(d) \tag{3.49}$$

If one removes all non-relevant documents (i.e., $r_n = 1$) in Eq. 3.48, the process to smooth the document weights (with increasing smoothing parameter s) can be considered as an attempt to gradually approximate the true query model in Eq. 3.44. The more smooth document weights will match the corresponding relevance judgements better, due to the fact that the relevance judgements of all documents are the same. In fact, if $r_n = 1$ and $s = +\infty$, the query model in Eq. 3.49 will be the true query model in Eq. 3.44.

By using true relevance information (when $r_n = 1$) and document weight smoothing together, it is expected that the performance bias and variance can drop simultaneously along with the increasing smoothing parameter s. We will evaluate the trends of changing performance bias and variance along with the increasing smoothing parameter s.

The estimation bias and variance are more likely to be reduced simultaneously, compared with the performance bias and variance. This is because the estimated query model in Eq. 3.49 is designed to get closer to the true query model in Eq. 3.44, as the s increases when $r_n = 1$.

3.2.3 Hypotheses

Now, we summarize a number of hypotheses. Based on the above analysis, it turns out that the factors which can affect bias and variance include not only various *query model factors* described in Section 3.2.2, but also the *evaluation factors*, e.g., different kinds

of bias-variance (using different metrics), different query sets or test collections. Like in other IR problems, the uncertainty happens in the bias-variance analysis in the sense that even for the same query model factor, the bias-variance trend can be different on different evaluation factors. In the following hypotheses, for better readability, we only include different trends for different kinds of bias-variance, while the term "likely" can indicate the uncertainty on different sets of queries, different test collections, or different parameter values. The uncertainties are explained in the previous analysis for each estimated query model. We will also evaluate the hypotheses and explain different observations on different queries or test collections in the experiments.

$h1$: For the original query model and the expanded model by RM, the performance bias-variance tradeoff will occur . The estimation bias-variance tradeoff may not occur when using KL-divergence as the metric, while the tradeoff is more likely to occur using other metrics (JS-divergence or Cosine similarity) in estimation bias-variance.

$h2$: For the combined query model, the performance bias-variance tradeoff will occur, although bias and variance are likely to be reduced simultaneously when λ is close to 0 but not 0. The KL-divergence-based estimation bias-variance tradeoff may not occur, while the tradeoff is more likely to occur in JS-divergence or Cosine similarity based estimation bias-variance.

$h3$: For the smoothed query model, performance bias-variance tradeoff will occur. Compared with the performance bias and variance, the estimation bias and variance are more likely to be reduced simultaneously, although the tradeoff will still occur.

$h4$: For the expanded query model with available true relevance information (e.g., explicit relevance feedback [19]), it is very likely that performance bias and variance can be reduced simultaneously. Compared with the performance bias and variance, the estimation bias and variance are more likely to be reduced simultaneously.

$h5$: For the expanded query model with available true relevance information and document weight smoothing, there is a trend of performance bias and variance can be reduced simultaneously. Compared with the performance bias and variance, the estimation bias and variance are more likely to be reduced simultaneously.

3.3 Experiments

In this section, for each estimated query model described in the previous section, we are going to evaluate its corresponding hypothesis on the bias-variance tradeoff, and investigate the use of the sum of bias and variance as the indicator/metric for the retrieval robustness and for the total estimation quality.

[19] In this study, we are using the relevance judgements in the test collection to simulate the explicit relevance feedback of users

3.3.1 Evaluation Set-up

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, e.g., newswire articles and 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. Lemur 4.7 (Ogilvie & 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.

The first-round retrieval is carried out by a baseline language modeling (LM) approach, i.e., the query-likelihood (QL) model (Ponte & Croft 1998, Zhai & Lafferty 2001), which uses the original query model. The smoothing method for the document language model is the Dirichlet prior (Zhai & Lafferty 2001) with fixed value $\mu = 700$.

After the first-round retrieval, the top n ranked documents are selected as the pseudo-relevance feedback (PRF) documents for the query expansion task. We report the results with respect to $n = 30$. Nevertheless, we have similar observations on other n (e.g., 50, 70). The Relevance Model (RM) in Eq. 3.41, is used as the basic method for query expansion. The number of expanded terms is fixed as 100. For any query model (including the original one), 1000 documents are retrieved by the negative KL-divergence measure.

3.3.2 Evaluation on Performance Bias and Variance

Recall, that the performance bias and variance are used to study the retrieval effectiveness and stability, respectively. We first describe the experimental results about retrieval effectiveness and stability. We then plot the performance bias and variance to verify and explain our hypotheses in Section 3.2.3.

In our task, average precision (AP) is used as performance metric for each query q_i, and the mean average precision (MAP) is used to measure the overall retrieval effectiveness. Then, in Eq. 3.3, $E(\widehat{P})$ represents MAP, and the larger MAP corresponds to the smaller performance bias (see Eq. 3.4). In Eq. 3.4, P can represent the MAP of the true query model over all queries. The variance of average precision (VAP), which can be represented by $Var(\widehat{P})$ in Eq. 3.5, captures the performance variance and can indicate the retrieval stability. The smaller the VAP, the better the stability.

The summed quantity $E(\widehat{P} - P)^2$ (in Eq. 3.6), which takes into account both bias and variance, can be considered as a retrieval robustness metric which considers both retrieval effectiveness and stability. We will refer $E(\widehat{P} - P)^2$ as $bias^2 + var$. In addition, another robustness metric, i.e., $<Init$ in (Zighelnic & Kurland 2008), which tests the percentage of queries for which the retrieval performance is worse than that of the initial ranking (i.e. QL), is also adopted.

Table 3.3: Retrieval effectiveness-stability of original query model (QL, $\lambda = 1$) and expanded query model (RM, $\lambda = 0$)

Collections	**WSJ8792**			**AP8889**		
Topics	Topics 151-200			Topics 151-200		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
QL	31.25	5.567	—	30.43	6.255	—
RM	37.01*	6.367	30	38.10*	8.368	30

Collections	**ROBUST2004**			**WT10G**		
Topics	Topics 601-700			Topics 501-550		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
QL	29.15	4.121	—	19.78	2.213	—
RM	33.26*	5.550	45	21.59*	2.929	46

*Statistically significant improvements over QL at level 0.05 by Wilcoxon signed rank test

$<Init$ is dependent on the performance of original query model and then is not applicable for the initial ranking (Zighelnic & Kurland 2008). On the other hand, the summed metric $E(\widehat{P} - P)^2$ is independent of the baseline method (i.e., the initial ranking by original query model). We will report the evaluation results for both metrics and analyze the relations and difference between them.

We will also report the results of the additional performance bias-variance based on $\widehat{\rho}$ and $\widehat{\rho}'$ (in Section 3.1.3). Note that Appendix includes the results when the bias and variance can be reduced simultaneously on both notions of variables ($\widehat{\rho}$ and $\widehat{\rho}'$).

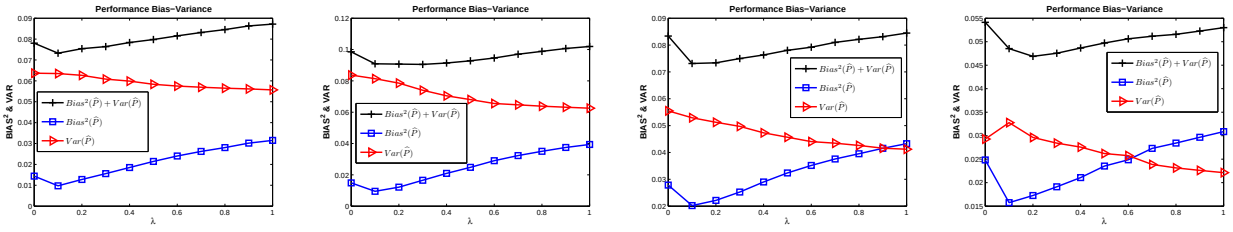
Original and Expanded Query Models

As we can see from Table 3.3, on four collections, the expanded query models computed by RM are more effective than the original ones used in the query likelihood (QL) model. This can be observed from the experimental fact, that RM's MAP significantly outperforms QL's on every collection.

On the other hand, $<Init$ shows that at least 30% queries (or even 46% on WT10G) perform worse after the query expansion. In addition, the variance of average precision (denoted as VAP) over different queries increases on each collection, meaning that query expansion hurts the retrieval stability. Therefore, we can verify that there is a tradeoff between the retrieval effectiveness and stability.

This tradeoff corresponds to the performance bias-variance tradeoff, which can be observed in Figure 3.1, where $\lambda = 0$ corresponds to the expanded query model by RM and $\lambda = 1$ to the original model in the combined query model (see Eq. 3.45). This tradeoff supports our hypothesis $h1$ in Section 3.2.3. The detailed analysis of such tradeoff has been given in Section 3.2.2.

Now, we look at the result for $bias^2$+var plotted in Figure 3.1. $bias^2$+var is a quantity summed over bias and variance, corresponding to the retrieval effectiveness and stability, respectively. Therefore, $bias^2$+var can reflect a combined criteria of retrieval effectiveness and stability, assuming that the retrieval robustness can be decomposed into retrieval



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

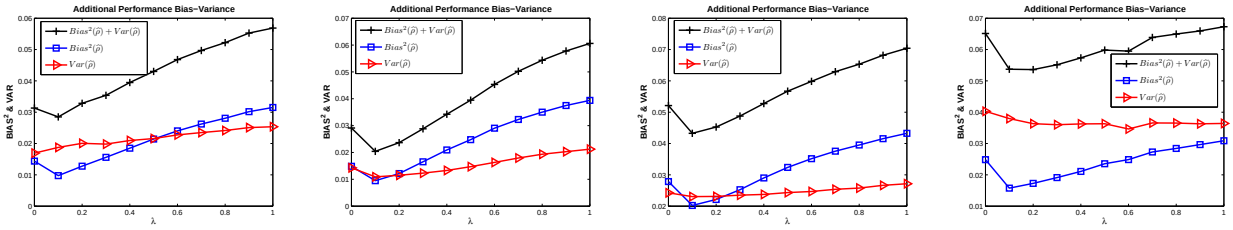
Figure 3.1: Performance bias-variance of the combined query model. The x-axis shows λ values from [0,1] with increment 0.1, and the y-axis represents the bias-variance results. $Bias^2$ (which is proportional to $Bias$) is marked with "blue square", Var is marked with "red triangle" and the sum of $Bias^2$ and Var is marked with "black plus sign".

effectiveness and stability over queries. It shows that the original query model is more robust than the expanded query model on WSJ8792, AP8889 and ROBUST2004. On WT10G, the original query model is slightly less robust than the expanded query model.

The robustness reflected in $bias^2 + var$ is different from the robustness reflected by another robustness metric $<Init$, which shows the percentage of queries for which the performance is worse than the initial ranking by original query model. Therefore, in any case, the $<Init$ for the original query model will always be 0. This means that the original query model will be the most robust query model, no matter how bad its MAP is. Therefore, the metric $<Init$, which is dependent on the initial ranking, is not applicable to the original query model.

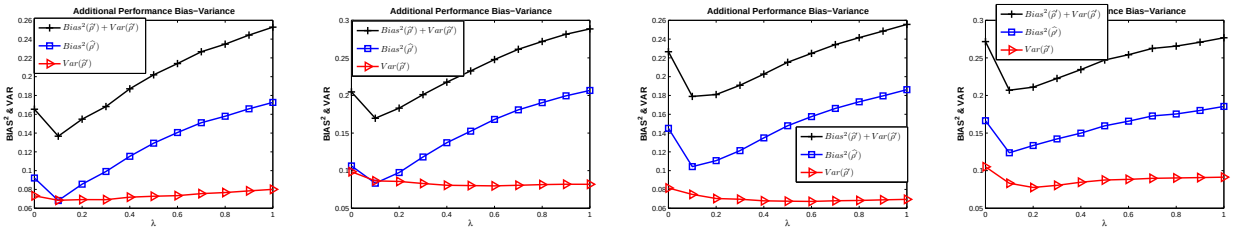
With respect to the additional performance bias-variance based on $\widehat{\rho}$, Figure 3.2 shows that the tradeoff does not happen on WSJ8792, AP8889, ROBUST2004, while the tradeoff happens on WT10G. According to the previous discussions in Section 3.1.3, the reason why the additional performance variance is different from the performance variance is due to the system variance. The system variance is related to the variance of retrieval performance of the true query model across different queries. This variance is shown in Table 3.7, where the true query model is corresponding to NSRM ($r_n = 1, s = +\infty$). It shows that on WT10G, the system variance is the smallest. This explains that the (decreasing) trend of additional performance variance is similar to that of the performance variance on WT10G, while on other collections, the trends between two variances are different.

Now, let us look at the results (shown in Figure 3.3) concerning the additional bias-variance based on $\widehat{\rho}'$, in which the system variance of the (regularized) performance target has vanished. It shows that on AP8889, ROBUST2004, and WT10G, the expanded query model ($\lambda = 0$) has smaller bias but larger variance, than the original query model ($\lambda = 1$). The above results show that compared with the bias-variance based on $\widehat{\rho}$, the bias-variance based on $\widehat{\rho}'$ (i.e., the regularized $\widehat{\rho}$) is more likely to have a tradeoff.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.2: Additional Performance bias-variance (based on $\widehat{\rho}$) of the combined query model. The x-axis shows λ values from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.3: Additional Performance bias-variance (based on $\widehat{\rho}'$) of the combined query model. The x-axis shows λ values from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.

Combined Query Models with Different Combination Coefficient

Here, we evaluate the combined query model (denoted as CRM [20]), which is the combination (see Eq. 3.45) of the original query model and the expanded query model by RM. The experimental results are shown in Table 3.4 and Figure 3.1, where the parameter λ is the combination coefficient with respect to the original query model and λ is chosen from the interval [0,1] [21].

The expanded query model by RM is entailed in CRM with $\lambda = 0$ (i.e., without original query model combined). From Table 3.4, as the λ increases, the mean performance (i.e., MAP) increases first when λ is close to 0, and then drops on the rest of λ values. This indicates, that the retrieval effectiveness is improved first in a small interval of λ, and then becomes worse when the estimated query model approaches the original query model. On the other hand, the performance variance (VAP) usually drops, indicating that the retrieval stability increases along with an increasing λ. To sum up, both the effectiveness and stability can be improved simultaneously using a small λ (e.g., 0.1), and the effectiveness-stability tradeoff will often occur obviously after the λ is getting bigger.

[20] This is often referred to as RM3. We call it CRM here to highlight the combination strategy, and let it be consistent with the superscript (c) in $\widehat{\theta}_{q_i}^{(c)}$ in Eq. 3.45.

[21] In this thesis, for a better presentation, in each table, only part of results are reported. The reported results are enough to describe our observation. In each figure, however, we report more results corresponding to more parameter values.

Table 3.4: Retrieval effectiveness-stability of combined query model

Collections	**WSJ8792**			**AP8889**		
Topics	Topics 151-200			Topics 151-200		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
CRM ($\lambda = 0$)	37.01	6.367	30	38.10	8.368	30
CRM ($\lambda = 0.1$)	39.13*	6.351	14	40.54*	8.140	14
CRM ($\lambda = 0.3$)	36.52	6.084	10	37.43	7.440	8
CRM ($\lambda = 0.6$)	33.50	5.752	10	33.24	6.557	6
CRM ($\lambda = 0.9$)	31.63	5.614	8	30.92	6.319	6
CRM ($\lambda = 1$)	31.25	5.567	—	30.43	6.255	—

Collections	**ROBUST2004**			**WT10G**		
Topics	Topics 601-700			Topics 501-550		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
CRM ($\lambda = 0$)	33.26	5.550	45	21.59	2.929	46
CRM ($\lambda = 0.1$)	35.73*	5.363	32	24.80*	3.428	32
CRM ($\lambda = 0.3$)	34.06	5.291	31	23.52*	3.280	30
CRM ($\lambda = 0.6$)	31.20	4.404	17	21.58	2.575	16
CRM ($\lambda = 0.9$)	29.57	4.158	18	20.13	2.262	14
CRM ($\lambda = 1$)	29.15	4.121	—	19.78	2.213	—

*Statistically significant improvements over RM at level 0.05 by Wilcoxon signed rank test

Now, let us examine the above observation from the perspective of performance bias-variance plotted in Figure 3.1. As λ increases from 0 to 1 with increment 0.1, in most cases, the bias-variance tradeoff happens, evidenced by the fact that the bias and variance change in opposite trends in most cases. Only for a small λ (e.g., 0.1), both the bias and variance can be reduced. The above observation is consistent with our hypothesis $h2$ in Section 3.2.3.

The reason why a small λ performs well is related to the sparsity problem of the original query model which only contains original query terms. Original query terms in the original query model have much bigger probability values than those in the expanded query model. Therefore, a small λ can adjust the probability of original query terms in the expanded query model, while preventing the expanded query model from being dominated by original query terms. In the previous section, we described similar observations.

Evaluation results regarding the robustness metric $bias^2$+var in Figure 3.1 also suggests that the combined query model with a small positive λ can be more robust in the sense of the combined effect of retrieval effectiveness and stability. As we discussed previously, $bias^2 + var$ reflects different aspect of robustness from another robustness metric $<Init$. $<Init$ favors the combined query model with a big λ close to 1, because it moves towards the original query model where $<Init$ is 0. However, $<Init$ does not take into account the retrieval effectiveness. Actually, $<Init$ has similar trends with VAP, which corresponds to the retrieval stability.

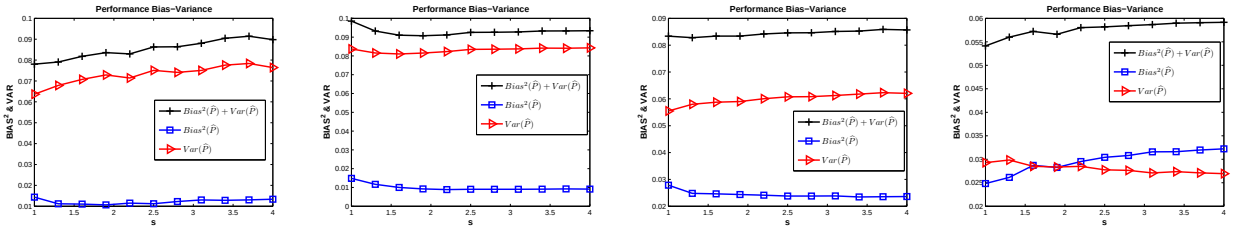
With respect to the additional performance bias-variance based on $\widehat{\rho}$, Figure 3.2 shows that the tradeoff does not happen on WSJ8792, AP8889, ROBUST2004, while the tradeoff happens on WT10G. Now, let us look at the results (shown in Figure 3.3) about the additional bias-variance based on $\widehat{\rho}'$. It shows that on AP8889 and ROBUST2004, the

Table 3.5: Retrieval effectiveness and stability of smoothed query model

Collections	**WSJ8792**			**AP8889**		
Topics	Topics 151-200			Topics 151-200		
Metrics	MAP(%)	VAP (%)	<*Init*(%)	MAP(%)	VAP (%)	<*Init*(%)
QL	31.25	5.567	—	30.43	6.255	—
SRM ($s = 1$)	37.01	6.367	30	38.10	8.368	30
SRM ($s = 1.3$)	38.42*	6.787	24	39.48*	8.160	30
SRM ($s = 1.9$)	38.67*	7.289	26	40.67*	8.151	26
SRM ($s = 2.5$)	38.41*	7.508	28	40.77*	8.348	28
SRM ($s = 100$)	34.76	7.338	38	39.65*	8.306	34

Collections	**ROBUST2004**			**WT10G**		
Topics	Topics 601-700			Topics 501-550		
Metrics	MAP(%)	VAP (%)	<*Init*(%)	MAP(%)	VAP (%)	<*Init*(%)
QL	29.15	4.121	—	19.78	2.213	—
SRM ($s = 1$)	33.26	5.550	45	21.59	2.929	46
SRM ($s = 1.3$)	34.20*	5.798	45	21.18	2.984	48
SRM ($s = 1.9$)	34.34*	5.901	46	20.54	2.835	56
SRM ($s = 2.5$)	34.52*	6.076	47	19.91	2.775	56
SRM ($s = 100$)	32.11	6.046	48	17.52	2.378	60

*Statistically significant improvements over RM at level 0.05 by Wilcoxon signed rank test



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.4: Performance bias-variance of the smoothed query model. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.

bias-variance tradeoff obviously occurs. On WT10G, compared with the expanded query model (when $\lambda = 0$), the combined query model (when $\lambda > 0.5$) has bigger bias but smaller variance, which is also a tradeoff. The above results show, that compared with the performance bias-variance, the additional performance bias-variance (based on $\widehat{\rho}$) is less likely to have a tradeoff. In addition, compared with the bias-variance based on $\widehat{\rho}$, the bias-variance based on $\widehat{\rho}'$ (i.e., the regularized $\widehat{\rho}$) is more likely to have a tradeoff.

Expanded Query Model with Smoothed Document Weights

Now, we evaluate the expanded query model by RM with smoothed document weights (denoted as SRM) described in Section 3.2.2. Recall that the bigger the smoothing parameter s is, the more smoothing the document weights would be. For RM, we can consider its smoothing parameter s as 1, meaning the document weights remain unchanged. Therefore, RM corresponds to SRM ($s = 1$) in Table 3.5 and Figure 3.4.

Results in Table 3.5 show that a moderate smoothing (e.g., $s < 2$) can help improve

the retrieval effectiveness (i.e., MAP) of RM on 3 (i.e., WSJ8792, AP8889, ROBUST2004) out of 4 collections. Regarding the retrieval stability, VAP shows an increasing trend on WSJ8792, AP8889 and ROBUST2004, leading to a dropping stability. On WT10G, however, VAP decreases along with the increasing s. Note that too much smoothing (e.g. $s = 100$) can hurt both the retrieval effectiveness and stability.

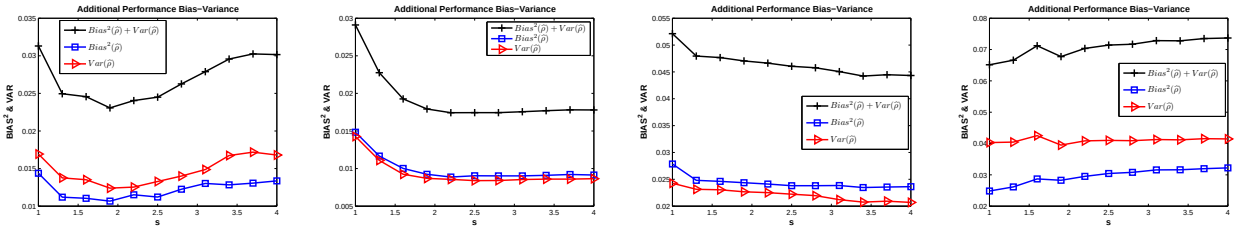
Let us look at the performance bias-variance shown in Figure 3.4, where parameter s is chosen from the range of [1,4] with the increment 0.3. Along with the increasing smoothing parameter s, the performance bias drops on WSJ8792 ($s < 1.9$), AP8889, ROBUST2004, and increases on WT10G. On the other hand, the performance variance increases on WSJ8792, AP8889 ($s > 1.6$) and ROBUST2004, and drops on WT10G. To sum up, we can observe a clear bias-variance tradeoff on each collection. The above evidence support our hypothesis $h3$.

We now explain why the observations on WSJ8792, AP8889 and ROBUST2004 are different from those on WT10G. Please also refer to Section 3.2.2 for more details. Recall that the document weight smoothing is motivated by the smoothness among relevant documents in deriving the true query model. Smoothing can also help improve the smoothness of relevant feedback documents in generating the estimated query model. Intuitively, a better initial ranking can have more relevant feedback documents, which indicates that the smoothing can be more helpful. The initial ranking performance averaged over all queries on WSJ8792, AP8889 and ROBUST2004 is better than that on WT10G (see MAP of QL in Table 3.5). Therefore, on the first three collections, it is more likely that smoothing can improve MAP and reduce performance bias.

Even if the mean performance on WSJ8792, AP8889 and ROBUST2004 is improved, the performance of some individual queries (with relatively poor initial ranking) can be hurt or can not be improved. This can cause the instability of retrieval performance across queries and increase the performance variance on the three collections. As discussed in Section 3.2.2, Higher fluctuation of initial ranking performance over queries is more likely to cause the increasing performance variance. The performance variances of the original query model (see VAP of QL in Table 4) on WSJ8792, AP8889 and ROBUST2004 are all higher than that on the WT10G. That is also why the performance variance on the first three collections is more likely to increase when s increases.

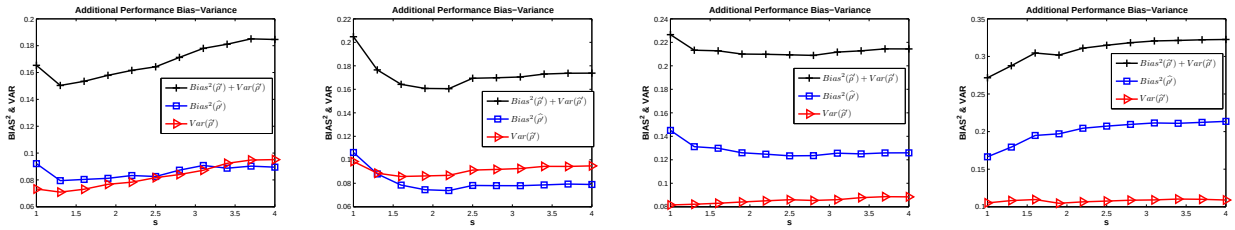
With regard to the robustness metric $<$*Init*, the smoothing ($s < 1.9$) can make it drop on WSJ8792 and AP8889, meaning that the percentage of queries for which the performance are hurt is reduced on these two collections. For the robustness metric $bias^2 + var$, it drops on AP8889 ($s < 1.9$), meaning that smoothing have a good combined effect of effectiveness and stability on this collection.

With respect to the additional performance bias-variance based on $\widehat{\rho}$, Figure 3.5 shows that the tradeoff does not happen on all collections. Now, let us look at the results (shown in Figure 3.6) related to the additional bias-variance based on $\widehat{\rho}'$. It shows that on



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.5: Additional Performance bias-variance (based on $\widehat{\rho}$) of the smoothed query model. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.6: Additional Performance bias-variance (based on $\widehat{\rho}'$) of smoothed query model. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.

ROBUST2004, the bias-variance tradeoff obviously occurs. On WSJ8792, compared with the expanded query model (when $s = 1$), the smoothed query model (when $1.6 < \lambda < 2.5$) has smaller bias but bigger variance, which is also a tradeoff.

Expanded Query Model with Available Non-Relevant Data

In this subsection, we carry out experiments for the expanded query model by RM with part of non-relevant data available and the resulting query model is denoted as NRM in Table 3.6. According to Section 3.2.2. a certain percentage (denoted as r_n in Table 3.6) of non-relevant documents are assumed to be available and we simply remove those non-relevant documents (see Eq. 3.48) in generating the query model. Thus, the expanded query model (by RM only) corresponds to the NRM ($r_n = 0$), meaning that no non-relevant data is available. Table 3.6 and Figure 3.7 summarize the experimental results.

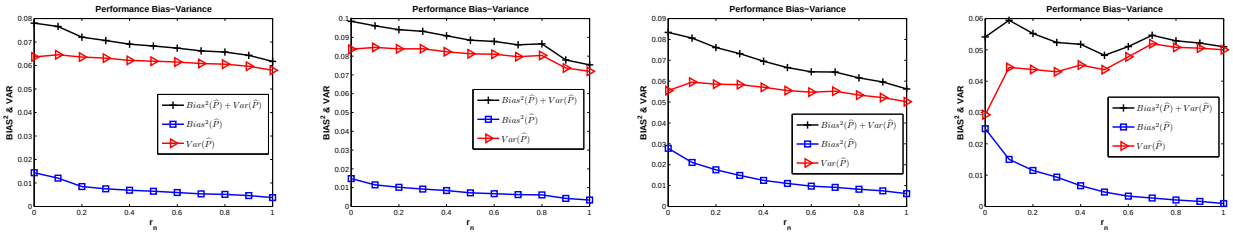
Table 3.6 shows that after we increase the ratio (r_n), both the effectiveness and stability are improved simultaneously on WSJ8792, AP8889 and ROBUST2004. Specifically, with increasing r_n, the MAP values are increasing, and meanwhile VAP generally have a decreasing trend, indicating a better retrieval stability. On WT10G, as the r_n increases, the MAP increases too, showing better retrieval effectiveness. Regarding the stability, VAP often increases. The trend of VAP on WT10G is different from those on other test

Table 3.6: Retrieval effectiveness-stability of the expanded query model by RM with non-relevant data available

Collections	**WSJ8792**			**AP8889**		
Topics	Topics 151-200			Topics 151-200		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
QL	31.25	5.567	—	30.43	6.255	—
NRM ($r_n = 0$)	37.01	6.367	30	38.10	8.368	30
NRM ($r_n = 0.1$)	38.02	6.456	28	39.56*	8.463	32
NRM ($r_n = 0.3$)	40.33*	6.216	22	40.65*	8.393	28
NRM ($r_n = 0.5$)	40.96*	6.184	22	41.77*	8.122	24
NRM ($r_n = 1$)	42.85*	5.796	22	44.42*	7.193	18

Collections	**ROBUST2004**			**WT10G**		
Topics	Topics 601-700			Topics 501-550		
Metrics	MAP(%)	VAP (%)	$<Init$(%)	MAP(%)	VAP (%)	$<Init$(%)
QL	29.15	4.121	—	19.78	2.213	—
NRM ($r_n = 0$)	33.26	5.550	45	21.59	2.929	46
NRM ($r_n = 0.1$)	35.43*	5.956	38	25.09*	4.444	40
NRM ($r_n = 0.3$)	37.75*	5.831	32	27.67*	4.299	28
NRM ($r_n = 0.5$)	39.45*	5.548	28	30.57*	4.364	22
NRM ($r_n = 1$)	42.11*	5.014	21	34.28*	5.002	18

*Statistically significant improvements over RM at level 0.05 by Wilcoxon signed rank test



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.7: Performance bias-variance of the expanded query model with non-relevant data. The x-axis shows non-relevance percentage r_n from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.

collections. We will explain such difference after we describe the bias-variance figure next.

Let us see the performance bias-variance plotted in Figure 3.7, where parameter r_n is in the interval [0,1] with increment 0.1. It clearly shows that on WSJ8792, AP8889, ROBUST2004, performance bias and variance can be reduced simultaneously. The above evidences support our analysis in Section 3.2.2 and the hypothesis $h4$.

The trend of performance variance on WT10G is different from those on the first three collections. Now we are going to explain it (also see Section 3.2.2). The initial ranking on the WT10G is poor (see MAP of QL in Table 3.6) and then for many queries there are a large number of non-relevant feedback documents in the feedback document set. For those queries, after removing some non-relevant ones, most remaining documents could be still non-relevant and the room for performance improvement is very small. At the meanwhile, there may exist some other queries for which the performance improvement can be bigger. As a result, the performance variance will be increased.

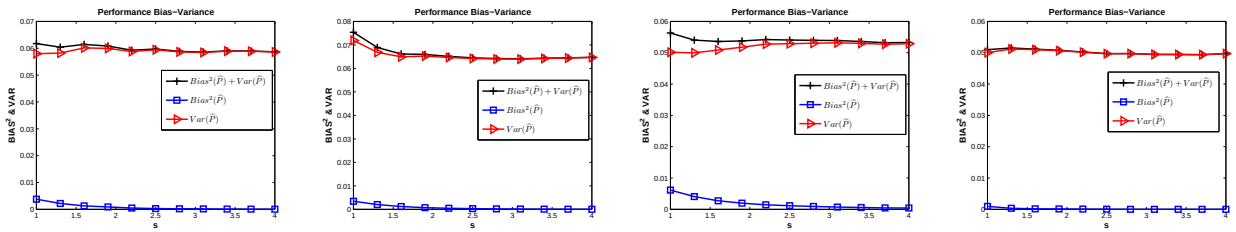
With regard to the robustness metric $<Init$, it drops on each collection, meaning that

Table 3.7: Retrieval effectiveness-stability of the expanded query model by RM on relevant documents with smoothed document weight.

Collections	**WSJ8792**			**AP8889**		
Topics	Topics 151-200			Topics 151-200		
Metrics	MAP(%)	VAP (%)	<*Init*(%)	MAP(%)	VAP (%)	<*Init*(%)
QL	31.25	5.567	—	30.43	6.255	—
NSRM ($r_n = 0, s = 1$)	37.01	6.367	30	38.10	8.368	30
NSRM ($r_n = 1, s = 1$)	42.85*	5.797	22	44.42*	7.193	18
NSRM ($r_n = 1, s = 1.3$)	44.32*	5.820	14	45.76*	6.686	16
NSRM ($r_n = 1, s = +\infty$)	49.00*	6.026	12	50.28*	6.591	8

Collections	**ROBUST2004**			**WT10G**		
Topics	Topics 601-700			Topics 501-550		
Metrics	MAP(%)	VAP (%)	<*Init*(%)	MAP(%)	VAP (%)	<*Init*(%)
QL	29.15	4.121	—	19.78	2.213	—
NSRM ($r_n = 0, s = 1$)	33.26	5.550	45	21.59	2.929	46
NSRM ($r_n = 1, s = 1$)	42.11*	5.014	21	34.28*	5.002	18
NSRM ($r_n = 1, s = 1.3$)	43.57*	4.996	19	35.49*	5.118	16
NSRM ($r_n = 1, s = +\infty$)	49.95*	5.538	7	37.35*	5.034	16

*Statistically significant improvements over RM at level 0.05 by Wilcoxon signed rank test



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.8: Performance bias-variance of the expanded query models on relevant documents with smoothed document wight. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.

the percentage of queries for which the performance is hurt is reduced on every collection. For the robustness metric $bias^2 + var$ Figure 3.7, it also has a dropping trend on each collection, meaning that removing non-relevant documents have a good combined effect of effectiveness and stability on each collection.

Expanded Query Model with Document Weight Smoothing and Non-Relevant Data

Now, we evaluate the query model described in Eq. 3.49 in Section 3.2.2. This query model integrates the ideas of both removing some non-relevant documents and smoothing document weight in RM, and thus is denoted as NSRM in Table 3.6. The true query model in Eq. 3.44 is corresponding to the NSRM ($r_n = 1, s = +\infty$) in Table 3.7, where $r_n = 1$ means that all non-relevant documents have been removed and $s = +\infty$ leads to the case when the weights of all concerned relevant documents are the same. Note that the true query model (see Eq. 3.44) actually does not have any adjustable parameters.

Table 3.7 shows that when $r_n = 1$, the more smooth the document weights (as s increases), the better the retrieval effectiveness (see MAP). With respect to VAP, as s increases, there is dropping trend, indicating an increasing retrieval stability. The reason is that as s increases, the estimated query model in Eq. 3.49 can get closer to the true query model in Eq. 3.44.

We show the performance bias-variance in Figure 3.8, where $r_n = 1$ and s ranges from 1 to 4 with increment 0.3. As s increases, there is an obvious trend that bias and variance can be reduced simultaneously on each test collection. This observation can support the hypothesis $h5$.

From Figure 3.8, we observe that when s starts to increase, the performance VAP can increase a little bit on WSJ8792, AP8889 and WT10G. This is because when s starts to increase, for some queries, the performance improvements are slow, while for other queries, the improvements can be relatively fast, leading to the slightly increased VAP. However, as we can see from Figure 3.8, there is a clear drop of VAP in the end.

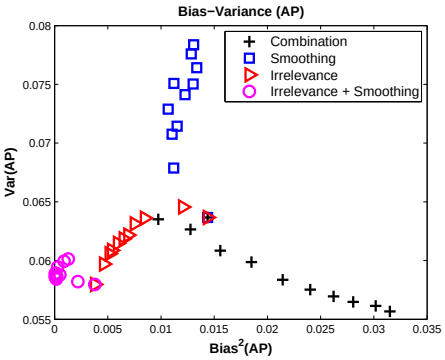
Now, let us look at the robustness metric $<$*Init* in Table 3.7. Unsurprisingly, the true query model, denoted as NSRM ($r_n = 1, s = +\infty$), is the most robust expanded query model on all collections. Another robustness metric $bias^2 + var$ also shows that the true query model is the most robust expanded query model. In Figure 3.8 also shows that as s increases, $bias^2 + var$ keeps dropping. This means that smoothing the weights of relevant documents can have a very good combined effect on retrieval effectiveness and stability on each collection.

Performance Bias-Variance Results for All Query Models

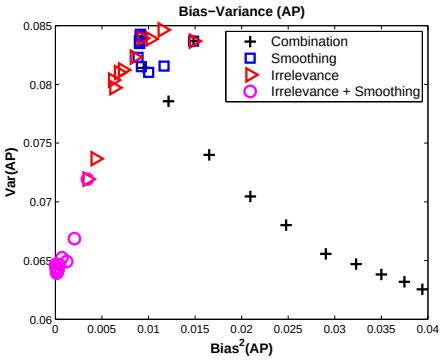
Previously, the bias-variance results of each kind of query model estimation method were plotted subsequently. Now, on each collection, we summarize the bias-variance results for all kinds of query models in each sub-figure in Figure 3.9.

On WSJ8792, it shows in Figure 3.9(a) that by removing irrelevant documents and smoothing document weight (corresponding the query model in Eq. 3.49), the smallest bias can be achieved, while its variance can be smaller than the variance when we only removing irrelevant documents or only smooth the document weights. We can observe that by only removing the irrelevant documents (see the query model in Eq. 3.48), the bias and variance often have the same trends. The document weight smoothing (see the smoothed query model in Eq. 3.42) has the more effects on reducing the variance than reducing the bias. Regarding the combination between the original query model and expanded query model, it shows a clear tradeoff in the results . Recall that the original query model is less complex than the expanded query model. The original query model actually achieves the smallest variance, but the largest bias as well.

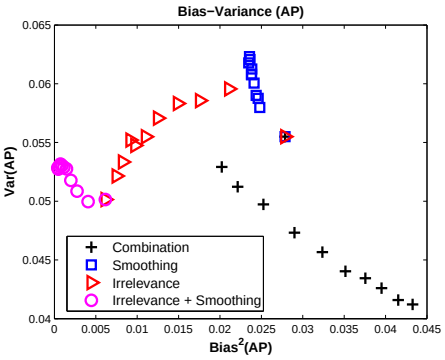
On the other three collections, the query model in Eq. 3.49 still achieves the smallest bias, while the original query model achieves the smallest variance. The bias-variance



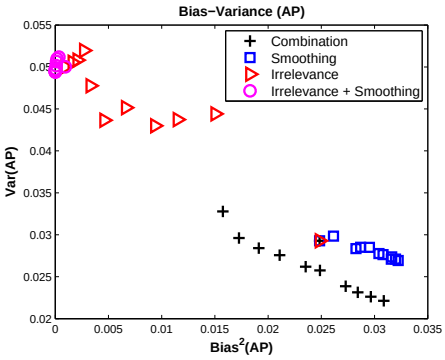
(a) WSJ8792



(b) AP8889



(c) ROBUST2004



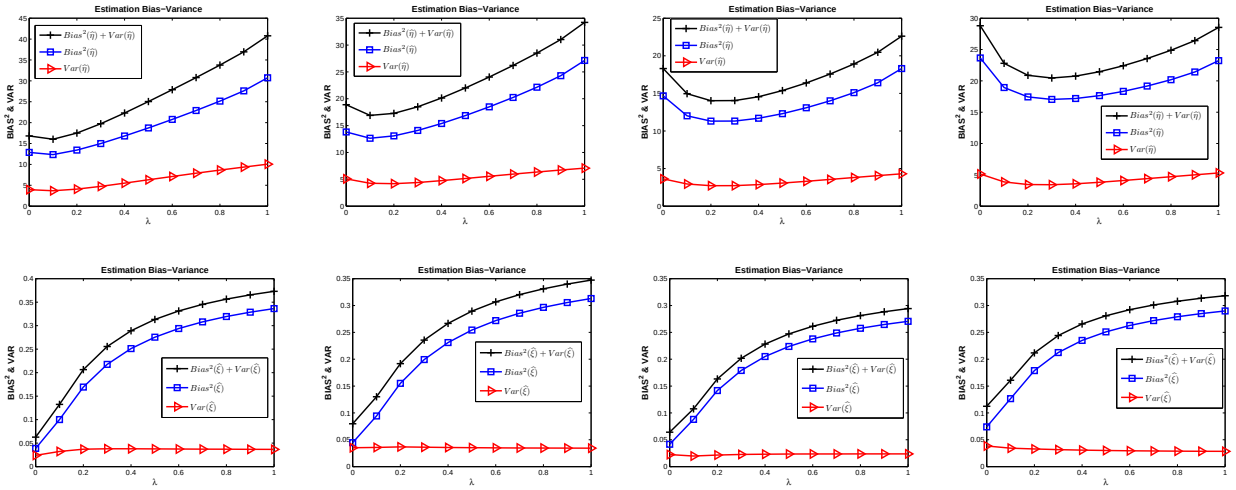
(d) WT10G

Figure 3.9: Performance bias-variance of all the concerned query models. The x-axis shows the squared bias and the y-axis shows the variance.

tradeoff of the smoothed query model becomes more clear on AP8889, ROBUST2004 and WT10G, than on WSJ8792. The bias-variance tradeoff is also very clear in the combination between the original query model and expanded query model on AP8889, ROBUST2004 and WT10G.

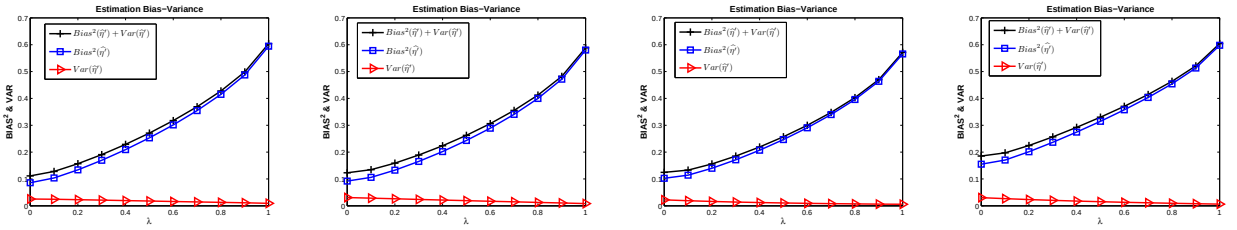
3.3.3 Evaluation on Estimation Bias and Variance

We now evaluate the estimation bias and variance (formulated in Section 3.1.8) of each aforementioned query model. This evaluation directly test the estimation quality of each query model with respect to the true query model. The evaluation metrics are the estimation bias and variance proposed in Section 3.1.8. Specifically, they are KL-divergence based $Bias(\widehat{\eta})$ and $Var(\widehat{\eta})$, and JS-divergence based $Bias(\widehat{\eta}')$ and $Var(\widehat{\eta}')$ formulated in Section 3.1.8, as well as Cosine similarity based $Bias(\widehat{\xi})$ and $Var(\widehat{\xi})$ as formulated in Section 3.1.8. In Appendix, we report the result of JS-divergence based estimation bias-variance when its trend is similar to the KL-divergence based result.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.10: Estimation bias-variance based on $\widehat{\eta}$ (1^{st} row) and $\widehat{\xi}$ (2^{nd} row) of the combined query model. The x-axis shows λ values from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.11: Estimation bias-variance (using $\widehat{\eta}'$, based on JS-divergence) of the combined query model

Original, Expanded and Combined Query Models

Figure 3.10 shows the results about the estimation bias and variance of the original and expanded query models, as well as the combination between them. $\lambda = 1$ corresponds to the original query model used in query likelihood (QL), while $\lambda = 0$ corresponds to the expanded query model by RM.

In Figure 3.10, as λ increases, the KL-divergence based $Bias(\widehat{\eta})$ and $Var(\widehat{\eta})$ drop first when λ is small and then increase when λ is approaching 1 on all collections. For the Cosine similarity based $Bias(\widehat{\xi})$, it increases on all collections, and the corresponding $Var(\widehat{\xi})$ has a dropping trend on AP8889 and WT10G. In Figure 3.11, we also plotted the JS-divergence based $Bias(\widehat{\eta}')$ and $Var(\widehat{\eta}')$, which has a clear tradeoff on all collections. In summary, we do not observe the bias-variance tradeoff regarding the KL-divergence based $\widehat{\eta}$, but we observe the bias-variance tradeoff with respect to the Cosine similarity based $\widehat{\xi}$

and JS-divergence based $\widehat{\eta}'$. This observation supports the hypothesis $h1$ and $h2$.

The reason why there is no tradeoff between $Bias(\widehat{\eta})$ and $Var(\widehat{\eta})$ is mainly because that when λ is becoming to 1, the variance $Var(\widehat{\eta})$ is not reduced. The increased variance is rooted on the sparsity of the original query model and the range of KL-divergence in [0,+∞]. In the original query model, only entries of original query terms have positive probabilities, while other entries are zeros. Hence, the original query model is quite sparse and the positive probabilities are relatively large values. On the other hand, the probabilities are much more smooth in the expanded query model of RM which are generated from the feedback documents (see Eq. 3.41). The true query model in Eq. 3.44 is also generated from the documents. As a result, the KL-divergence between the original query model and the true query model has much bigger scale than that between the expanded query model by RM and true query model. When λ is approaching 1, the combined query model will get closer to the original query model and suffer the sparsity problem, resulting in the increasing (KL-divergence based) bias and variance. The range of JS-divergence and Cosine similarity is [0,1], which can be considered as a normalized range of KL-divergence's one. In this normalized range, the variance is more likely to be smaller as we observed in Figures 3.10 and 3.11.

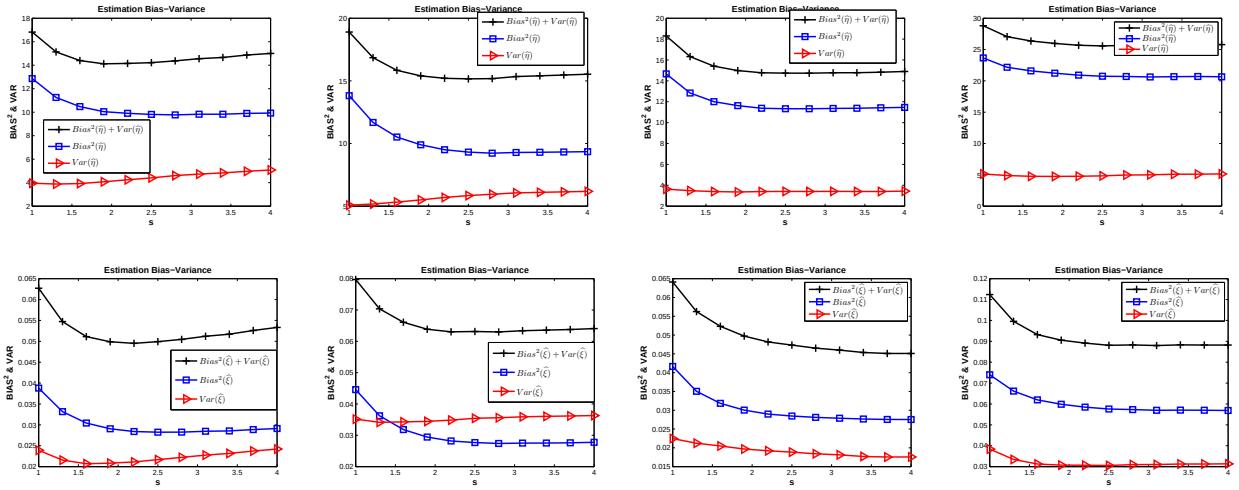
When λ is close to 0 but not 0, meaning that the combined query model is close to the expanded query model, the combination strategy will modestly adjust the probability of original query terms in the expanded query model. This can result in the bias and variance being reduced simultaneously (see the KL-divergence based estimation bias-variance in Figure 3.10). This observation corresponds to the performance bias-variance trend in Figure 3.1 (when $\lambda(\lambda > 0)$ is close to 0).

Expanded Query Model with Smoothed Document Weights

This set of experiments is to test the estimation bias and variance of the expanded query model by RM with smoothed document weights (see Section 3.2.2). The results are plotted in Figure 3.12. $s = 1$ corresponds to expanded query model by RM with its original document weights. Recall that the bigger the smoothing parameter s is, the more smoothing would be imposed on the document weights.

Based on both bias metrics (i.e., KL-divergence and Cosine Similarity), we observe that document weight smoothing can help reduce the estimation bias. Regarding the variance, Figure 3.12 shows that as s increases, both variance metrics have an increasing trend on WSJ8792 and AP8889, and a decreasing trend on the other two collections. We observe an obvious tradeoff on WSJ8792 and AP8889, while bias and variance can be reduced simultaneously on other collections.

As we explained in Section 3.2.2, document weight smoothing can play a bigger role in the reduction of estimation bias. Smoothing can not reduce the performance bias on WT10G (see Figure 3.4), but can reduce the estimation bias on the same collection. This



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.12: Estimation bias-variance based on $\widehat{\eta}$ (1^{st} row) and $\widehat{\xi}$ (2^{nd} row) of the smoothed query model. The x-axis shows smoothing parameter s from [1, 4] with increment 0.3, and the y-axis represents the bias-variance results.

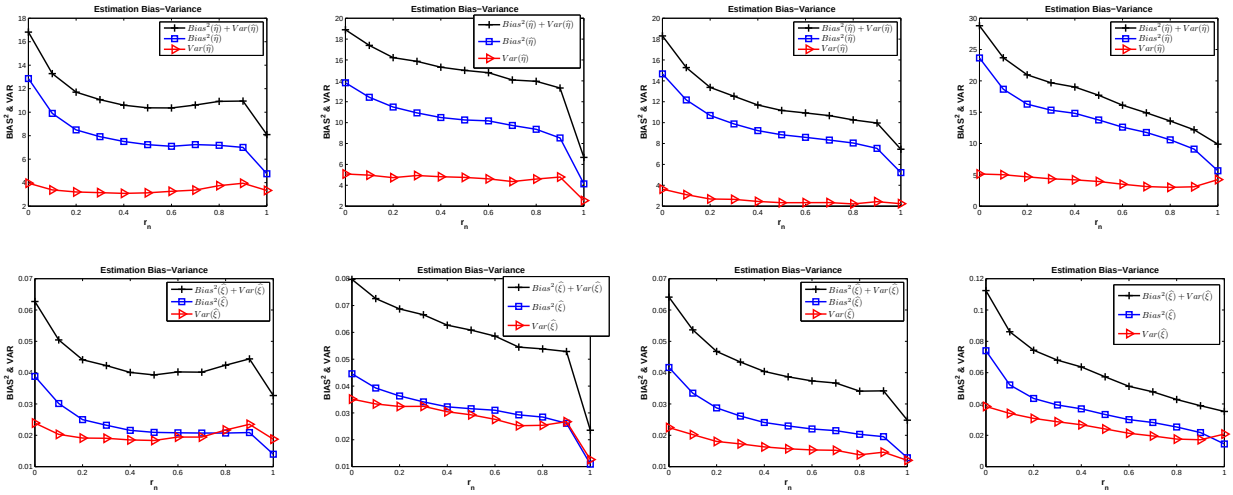
is because that estimation bias is more directly related to the estimation quality *w.r.t.* the true query model. The smoothness in the true query model is important, making the smoothing important for improving the estimation quality. The results also show, that compared with performance bias-variance in Figure 3.4 where 4 out 4 collections experienced bias-variance tradeoff, the estimation bias and variance are more likely to be reduced simultaneously. This can support the hypothesis $h3$.

The trends of estimation variance on WSJ8792 and AP8889 are different from those on ROBUST2004 and WT10G. As we explained in Section 3.2.2, it is because the performance variance of the initial ranking on WSJ8792 and AP8889 is higher than that on ROBUST2004 and WT10G (see VAP of QL in Table 3.5). High fluctuated initial performance would be more likely to cause the increasing estimation variance of the smoothed query model. Therefore, the estimation variance on WSJ8792 and AP8889 is increasing, while the estimation variance on other collections can be decreased.

Moreover, we can observe that $Var(\widehat{\xi})$ can be reduced more obviously than $Var(\widehat{\eta})$ on ROBUST2004 and WT10G. As we explained previously, the range of Cosine similarity $\widehat{\xi}$ is [0,1], which can be thought of as a normalized scale. On the normalized scale, the estimation variance is more likely to be reduced in this experiment.

Expanded Query Model with Available Non-Relevant Data

Here, we evaluate the estimation bias and variance of the expanded query model by RM with non-relevant documents available (see Section 3.2.2). The results are plotted in Figure 3.13, where r_n represents the percentage of known non-relevant documents. It is



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.13: Estimation bias-variance based on $\widehat{\eta}$ (1^{st} row) and $\widehat{\xi}$ (2^{nd} row) of the expanded query model with non-relevant data available. The x-axis shows non-relevance percentage r_n from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.

expected, that by increasing r_n (i.e., removing more non-relevant documents in RM), the estimation quality of the estimated query model can be improved. Note that the expanded query model by RM corresponds to the $r_n = 0$, meaning that no non-relevant data is used.

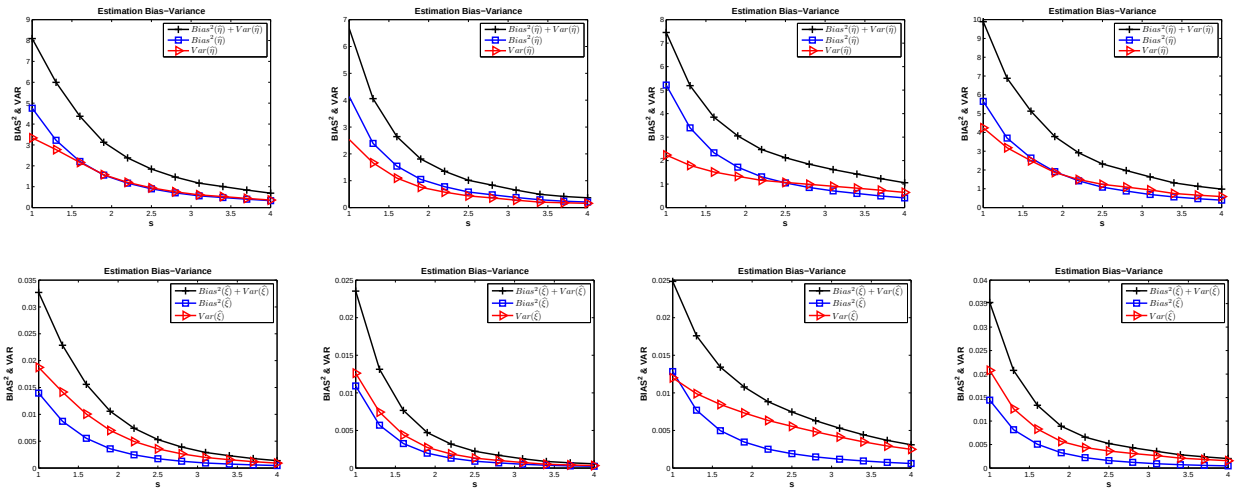
In Figure 3.13, both bias metrics $Bias(\widehat{\eta})$ and $Bias(\widehat{\xi})$ are dropping by removing more non-relevant documents in RM. This means that the estimated query model has better estimation quality in the expectation sense. As for the variance, generally, as r_n increases, both variance metrics $Var(\widehat{\eta})$ and $Var(\widehat{\xi})$ decrease, indicating that the estimation error/quality is reduced/increased stably over all concerned queries.

The estimation bias and variance can be reduced simultaneously on more collections (see Figure 3.13) than the performance bias and variance (see Figure 3.7). In Figure 3.7, the performance variance does not drop on WT10G and there is a bias-variance tradeoff. The above observations support the hypothesis $h4$.

Expanded Query Model with Document Weight Smoothing and Non-Relevant Data

Now, we evaluate the estimated query model in Eq. 3.49, which actually integrates the ideas of both document weight smoothing and removing non-relevant documents in the expanded query model computed by RM.

The experimental results in Figure 3.14 show that if truly relevant documents are used to generate the query model, the more smooth the document weights are, the better the estimation quality can be. The better estimation quality can be reflected by the decrease of bias and variance based on both statistics ($\widehat{\eta}$ and $\widehat{\xi}$). The estimation bias and variance



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 3.14: Estimation bias-variance based on $\widehat{\eta}$ (1^{st} row) and $\widehat{\xi}$ (2^{nd} row) of the expanded query model on relevant documents with smoothed document wight. The x-axis shows smoothing parameter s from [1, 4] with increment 0.3, and the y-axis represents the bias-variance results.

can be reduced simultaneously for all parameters values when s is increasing in Figure 3.14. However, the performance bias and variance can not always be reduced simultaneously for all parameter values in Figure 3.8. The above experimental observations support the hypothesis $h5$.

3.4 Discussion on Potential Impact of Bias-Variance Analysis

We now briefly discuss the potential impact of the proposed bias-variance analysis in the IR context. First, our experimental results support the hypotheses on bias-variance tradeoff, showing that the tradeoff between retrieval effectiveness and stability can be studied through the perspective of bias-variance tradeoff. The current analysis involves four factors in query language modeling, i.e., query model complexity, query model combination, document weight smoothness and non-relevant documents removal. In the future, the analysis can be extended to more general scenarios. For instance, we may be able to study the model complexity of other IR models (e.g., ranking functions). The combination of two query models can be extended to the combination/ensemble of multiple (tens or hundreds) rankers in the web search scenario. As another example, the document weight smoothness can be related to the diversity of topic coverage of feedback documents. Further, we may explore non-relevant documents removal in the implicit feedback or interactive feedback scenario. We expect that the bias-variance analysis in this thesis can potentially serve as a start point for the above interesting research directions.

Second, a better understanding of the above factors through the bias-variance analysis can provide insights on how to improve retrieval effectiveness and stability separately, or simultaneously. In Section 3.2.2-3.2.2 and in the experiments, we have explained when the bias-variance tradeoff can occur, and when the bias and variance can be reduced simultaneously. For example, in Section 3.2.2 and Section 3.3.2, we have explained why a small combination coefficient (e.g., $\lambda = 0.1$) can reduce the performance bias and variance simultaneously. As the λ is approaching 1, the performance bias-variance tradeoff will occur obviously. This shows that we can have an analysis support for the good parameters to reduce bias and variance simultaneously. In the future, we expect that for different applications with specific needs, one can adopt certain strategies for deriving the query models or other IR models based on the insights obtained from the bias-variance analysis. For example, if a system is more concerned about the retrieval stability, one may try to design a simpler method based on the analysis results on model complexity. If a system needs to balance effectiveness and stability, one can try to investigate a proper combination of different models based on the analysis of model combination effect.

Third, this research may potentially lead to a novel evaluation strategy. Specifically, the estimation bias-variance formulation can provide novel metrics (e.g., estimation bias and estimation variance) to evaluate the estimation quality with respect to the true query model. In addition, the summed quantity of performance bias and performance variance (see Eq. 3.6) serve as a kind of robustness metric. The retrieval robustness can be thought of as a criteria combined by the retrieval effectiveness and retrieval stability. The bias-variance formulation can model the decomposition of robustness into effectiveness and stability. In addition, one can give different weights for bias and variance respectively, in the summation of them, to reflect how the retrieval robustness should be decomposed differently in different scenarios. Moreover, the summed quantity of the additional performance bias and variance can also serve as the robustness metric. We will further investigate these issues and potential applications in the future.

3.5 Summary

we propose a novel bias-variance analysis framework to study the tradeoff between the retrieval effectiveness and stability of the query language modeling in the pseudo relevance feedback context. Specifically, we propose a performance bias-variance formulation. This analogy enables us to better analyze and understand the retrieval performance using the bias-variance analysis, which is a fundamental theory in machine learning and statistical estimation. We also go beyond the retrieval performance by directly measuring how closely an estimated query model can approach the true query model derived from the truly relevant documents. This leads to the estimation bias-variance formulation, which is based on the divergence or similarity between the estimated and the true query models.

Based on performance and estimation bias-variance formulations, we analyze a number of representative query model estimation methods and present five hypotheses based on our analysis. We then construct a systematic evaluation on TREC datasets, in order to test the hypotheses. Experimental results on both performance bias-variance and estimation bias-variance support the hypotheses. Based on the above observations, we expect that proposed bias-variance analysis can form an analytical basis and a novel evaluation strategy for the query language modeling, and potential for other IR tasks as well.

Chapter 4

Document Weight Smoothing

As shown in the previous chapter, the document weight smoothing is one of the important factors that affect the bias and variance in the query model estimation. In this chapter, we are going to further investigate document weight smoothing in depth. Recall that the document weights are based on the document relevance scores in the first-round retrieval. Therefore, the document weight is closely related to the document relevance estimation, and the document relevance estimation in the first-round retrieval can influence the query model estimation in the second-round retrieval.

In Chapter 1, we discussed two types of risk in the document relevance estimation. One is *rank-dependent risk* which refers to the relevance estimation risk that can influence the ranking of feedback documents, while the other is the *rank-independent risk* which does not influence the document ranking. It is important to control the estimation risk at the very early stage before it spreads and gets more complicated in the later stages (e.g., query model estimation).

In this chapter, we will explore the risk management in the document relevance estimation, in order to improve the quality of the document weight and the quality of query model estimation. We will first show that the document weight smoothing method can manage the rank-independent risk. After that, we will show how to tackle the rank-dependent risk, by proposing weight allocation methods on top of the weight smoothing method. The rank-independent risk management can be regarded as the micro-level adjustment, as opposed to the re-ranking approaches (tackling the rank-dependent risk). The latter can be regarded as the macro-level adjustment for the document relevance. We will apply the aforementioned risk management methods in the relevance feedback task and evaluate its usefulness in the query model estimation.

Table 4.1: Topmost 4 documents' QL weights ($S(d)$) and relevance judgements (r)

query id	$S(d_1)/r$	$S(d_2)/r$	$S(d_3)/r$	$S(d_4)/r$
#151	0.206/0	0.167/1	0.106/1	0.064/0
#152	0.153/0	0.097/1	0.085/0	0.075/1
#153	0.232/0	0.185/1	0.103/1	0.090/1

4.1 Rank-Independent Risk of Document Weight

Now, we describe the relation between the smoothness of the document weight and the risk of the document relevance estimation. Let us see an example distribution of document weights computed by the normalized query-likelihood (QL) scores, where QL is a typical estimator for document relevance in the first-round retrieval. In Table 4.1, it shows a distribution of QL document weights of topmost 4 documents for three queries[1]. We can observe that the QL document weights drop too rapidly along the ranking. Next, we will explain that this rapid dropping weights indicates a risk in the document relevance estimation, and we can smooth the document weights and make them drop slowly, in order to control such risk.

For those documents with the same relevance judgements, the document weights are often very different in Table 4.1. For example, for query 151, both d_2 and d_3 are relevant, however, the document weight of d_2 is much larger than that of d_3. For query 152, both d_1 d_3 are irrelevant, however, the document weights are quite different. Since the document with the same relevance judgements should have the same scores/weights [2], the above examples indicate that a risk/error occurs in the document relevance estimation. This risk is rank-independent since one can not say the ranking between two documents are wrong when they have the same relevance judgements. If we smooth the document weights and make the document weights drop slowly, the difference between those document with the same document weights can become smaller, leading to a reduced risk.

In addition, the document weight smoothing can alleviate the negative impact of the rank-dependent risk on the query model estimation. As shown in Table 4.1, for all three queries, the weights of d_1 are about twice the d_3's weights and three times the d_4's. All the d_1s, however, are actually irrelevant. It means that the ranks of all d_1s are wrong, indicating the rank-dependent risk in the document relevance estimation. The irrelevant document being mis-ranked highly can hurt the quality of query model estimation. If one smooth the document weights and make them drop slowly, the weight of $d1$ would not be too high and then its negative impact can be alleviated.

The above examples show that both types of risks may exist simultaneously in the

[1]The reported data from the WSJ8792 collection and relevance judgement $r = 1$ represents *relevant* and $r = 0$ represents *irrelevant*. More data can be found in (Zhang, Song, Zhao & Hou 2010)

[2]This is also a requirement in the true query model 2.2.3). We can observe that the risk in the document relevance will spread in the query model estimation.

estimated relevance scores. Therefore, we will first single out the effect of the rank-independent risk associated to different relevance estimators when the resultant ranks are identical (i.e., rank-equivalent). In this scenario, the document weight smoothing is a suitable method to manage the rank-independent risk since it can preserve the original ranking of feedback documents but yield new document weights which are different from the original ones. Bear in mind that the document weight smoothing may not be helpful in any cases for any queries. In Section 4.1.5, we will provide an entropy-bias explanation to show the rationality (in the expectation sense) of the document weight smoothing. Next, we first describe the formulation of the document weights obtained by the relevance estimation, and show two rank-equivalent relevance estimators.

4.1.1 Document Weight obtained by Relevance Estimation

Now, we introduce a formulation of document weight obtained by estimating document relevance. The probability of relevance of each document corresponds to one basic retrieval question (Lafferty & Zhai 2003): what is the probability of this document d being relevant to a query q? Accordingly, it can be formulated as $p(r|d,q)$ (Robertson & Zaragoza 2009). Let $\hat{p}(r|d,q)$ denote the estimate of $p(r|d,q)$. $\hat{p}(r|d,q)$ is usually not a probability, but rather a score.

Once we obtain $\hat{p}(r|d,q)$, assuming a uniform prior $p(d)$, we can normalize it as

$$S_q(d) = \frac{\hat{p}(r|d,q)}{\sum_{d' \in D} \hat{p}(r|d',q)} \tag{4.1}$$

where D is the document set. The normalized score $S_q(d)$ can be considered as a probability value which forms a relevance distribution S_q over all the documents in D. We can refer to $S_q(d)$ as an estimated relevance probability of document d with respect to the query q. $S_q(d)$ is actually used as the document weight used in the relevance feedback.

Our proposed rank-independent risk management is expected to be applicable to most retrieval models that can estimate the probability of relevance. In this thesis, our focus is on the language modeling (LM) approaches. As explained in our motivation, we are going to explore the rank-independent risk associated with any two rank-equivalent relevance estimations. Now, we are going to present two representative LM approaches which are rank-equivalent.

4.1.2 Rank-Equivalent LM Approaches

The query-likelihood (QL) approach (Ponte & Croft 1998, Zhai & Lafferty 2001) is a standard LM approach for the first-round retrieval. It is formulated as:

$$p(q|\theta_d) = \prod_{j=1}^{m_q} p(q(j)|\theta_d) \tag{4.2}$$

where $p(q|\theta_d)$ is the query-likelihood, $q = q(1)q(2)\cdots q(m_q)$ is the given query, m_q is q's length, and θ_d is a smoothed language model for a document d.

The Negative KL-Divergence (ND) (Lafferty & Zhai 2001) between the query language model θ_q and document language model θ_d is formulated as

$$-D(\theta_q|\theta_d) = -H(\theta_q, \theta_d) + H(\theta_q) \tag{4.3}$$

where $H(\theta_q, \theta_d)$ is the cross entropy between θ_q and θ_d, and $H(\theta_q)$ is the entropy of the θ_q. According to the derivation in (Lafferty & Zhai 2001, Ogilvie & Callan 2002), if a maximum-likelihood estimator is used to estimate the query language model θ_q, then

$$-H(\theta_q, \theta_d) = \frac{1}{m_q} \log p(q|\theta_d). \tag{4.4}$$

The above equation shows that $-H(\theta_q, \theta_d)$ is logarithmically proportional to the query-likelihood $p(q|\theta_d)$. This means that $-H(\theta_q, \theta_d)$ and $p(q|\theta_d)$ are equivalent in terms of ranking documents. Since in Eq. 4.3, the $H(\theta_q)$ is independent of document ranking, it turns out that negative KL-divergence is rank-equivalent to the query-likelihood approach.

4.1.3 Difference between the Two Rank-Equivalent Estimations

We now present the difference between the two document relevance distributions estimated by the QL model and ND model. For a given q, the document relevance distribution estimated by the QL model is denoted as:

$$S_q^{QL}(d) = \frac{p(q|\theta_d)}{\sum_{d' \in D} p(q|\theta_{d'})} \tag{4.5}$$

where D is a set consisting of all concerned documents.

The document relevance distribution estimated by the ND model can be defined as the normalized exponential of the negative KL-divergence:

$$S_q^{ND}(d) = \frac{\exp\{-D(\theta_q|\theta_d)\}}{\sum_{d' \in D} \exp\{-D(\theta_q|\theta_{d'}))\}} \tag{4.6}$$

The exponential transformation (i.e exp{}) is to transform the divergence value to a probability value. Since the $H(\theta_q)$ in Eq. 4.3 is a constant for every $d \in D$, it can be eliminated in the normalization process of Eq. 4.6. We then get

$$S_q^{ND}(d) = \frac{\exp\{-H(\theta_q, \theta_d)\}}{\sum_{d' \in D} \exp\{-H(\theta_q, \theta_{d'})\}} = \frac{[p(q|\theta_d)]^{\frac{1}{m_q}}}{\sum_{d' \in D} [p(q|\theta_{d'})]^{\frac{1}{m_q}}} \tag{4.7}$$

After normalizing $p(q|\theta_d)$ by $\sum_{d' \in D} p(q|\theta_{d'})$ (denoted as Z_{QL}), we have

$$S_q^{ND}(d) = \frac{[p(q|\theta_d)/Z_{QL}]^{\frac{1}{m_q}}}{\sum_{d' \in D}[p(q|\theta_{d'})/Z_{QL}]^{\frac{1}{m_q}}} = \frac{[S_q^{QL}(d)]^{\frac{1}{m_q}}}{\sum_{d' \in D}[S_q^{QL}(d')]^{\frac{1}{m_q}}} \tag{4.8}$$

It shows that in the estimated ND distribution S_q^{ND}, the relevance probabilities are raised to the powers of $\frac{1}{m_q}$ of $S_q^{QL}(d)$, turning to $[S_q^{QL}(d)]^{\frac{1}{m_q}}$ before normalization. Compared with the QL relevance distribution in Eq. 4.5, the ND relevance distribution in Eq. 4.6 is often more smooth. For example, if the normalized QL scores are 0.05 and 0.03 for two documents, given $m = 3$, after being raised to the powers of (1/3), the scores will become 0.7937 and 0.6694 (before normalization), respectively, meaning the relative difference between two ND scores becomes smaller. Note that the relative difference between any two scores are independent of the normalization step, since every score will be divided by the same normalization factor. Next, we are going to explain this smoothness in depth by proposing a powers-based risk management method in order to smooth scores/weights. The powers-based distribution remodeling is actually motivated by the powers-based idea in Eq. 4.8.

4.1.4 Powers-based Risk Management (PRM) Method

We will present a novel risk management method and provide a theoretical analysis to show that the method can make every pair of probabilities in an estimated distribution become more smooth so as to reduce (overall) rank-independent risk (without changing the original document rank). This method can remodel an estimated distribution and the remodeling method is motivated by the powers-based idea described in Eq. 4.8. Specifically, given a retrieval model and its estimated document relevance distribution S_q, the remodeling method will raise every probability in S_q to the powers ($f(q)$) and then normalize the revised probabilities. It can be formulated as:

$$\widetilde{S_q}(d) = \frac{[S_q(d)]^{\frac{1}{f(q)}}}{\sum_{d' \in D}[S_q(d')]^{\frac{1}{f(q)}}} \tag{4.9}$$

where $\widetilde{S_q}$ denotes the remodeled distribution, and the powers $f(q)(> 0)$ is a function of the query q. $f(q)$ can be m_q in Eq. 4.8,or can be other functions (detailed later). Here, we first explain the relations between this remodeling method and the rank-independent risk measurement. This remodeling algorithm preserves the original document rank and has a property described in Proposition 1. The proposition proves that in Eq. 4.9, the bigger $f(q)$ value (i.e. b in the Proposition), the smaller the relative difference between any two probabilities in the distribution $\widetilde{S_q}$ and thus the higher degree of overall smoothness of the distribution.

Proposition 1. *Given a distribution S_q, suppose $S_q(d_i)$ and $S_q(d_j)$ are the estimated relevance probabilities of any two document d_i and d_j, respectively. If $0 < a < b$, then the relative difference between $[S_q(d_i)]^{\frac{1}{b}}$ and $[S_q(d_j)]^{\frac{1}{b}}$ should be smaller than that between $[S_q(d_i)]^{\frac{1}{a}}$ and $[S_q(d_j)]^{\frac{1}{a}}$.*

Proof. For simplicity, in this proof, let S_i and S_j denote $S_q(d_i)$ and $S_q(d_j)$, respectively. Without loss of generality, we assume that $S_i > S_j > 0$. Then, we have

$$\frac{(S_i^{\frac{1}{b}} - S_j^{\frac{1}{b}})/S_j^{\frac{1}{b}}}{(S_i^{\frac{1}{a}} - S_j^{\frac{1}{a}})/S_j^{\frac{1}{a}}} = \frac{(S_i/S_j)^{\frac{1}{b}} - 1}{(S_i/S_j)^{\frac{1}{a}} - 1} \tag{4.10}$$

Since $(S_i/S_j) > 1$ and $0 < \frac{1}{b} < \frac{1}{a}$, we get $1 = (S_i/S_j)^0 < (S_i/S_j)^{\frac{1}{b}} < (S_i/S_j)^{\frac{1}{a}}$. This means that the right hand side of Eq. 4.10 is less than 1. Therefore, we have

$$\frac{(S_i^{\frac{1}{b}} - S_j^{\frac{1}{b}})/S_j^{\frac{1}{b}}}{(S_i^{\frac{1}{a}} - S_j^{\frac{1}{a}})/S_j^{\frac{1}{a}}} < 1 \tag{4.11}$$

The proposition then follows. □

A bigger $f(q)$ value will give more smoothing effects on too large or too small probabilities in the distribution $\widetilde{S_q}$, making the distribution become smoother.

In this paper, we adopt two options for $f(q)$, each corresponding to an instantiated algorithm of our method. The first option is m_q as used in the Eq. 4.8, where m_q is the length of query q. We denote this option as

$$f_{ND}(q) = m_q \tag{4.12}$$

Since m_q is often greater than 1, it turns out that the estimated distribution (i.e. $S_q^{ND}(d)$ in Eq. 4.8) by the ND model is often more smooth than the one (i.e. $S_q^{QL}(d)$ in Eq. 4.8) by the QL model.

The second option of $f(q)$ can be an adjustable parameter s as follows:

$$f_s(q) = s \ (s > 0) \tag{4.13}$$

This option allows us to have different remodeled distributions and a bigger s generally leads to a smoother remodeled distribution. For example, assuming the original weights are 0.6250 and 0.3750 for d_1 and d_2, and the parameter s is 3, then the smoothed document weights are 0.5425 and 0.4575, which becomes more smooth. Note that the original ranking (i.e., d_1 and d_2) reflected by the document weights is preserved. This powers-based smoothing method is actually the smoothing method discussed in the previous chapter. Next, we will show an entropy-bias explanation of the document weight smoothing and

the rank-independent risk.

4.1.5 An Entropy-Bias Explanation

Now, let us analyze the entropy associated with the document weight distribution (or called document relevance distribution) S_q (see Eq.4.1). The entropy of S_q is defined as:

$$H(S_q) = -\sum_{d \in D} S_q(d) \log S_q(d) \tag{4.14}$$

where H is the Shannon entropy of the distribution S_q, and D is the concerned document set, which includes the top n documents in our problem. The entropy $H(S_q)$ can represent the smoothness of the distribution S_q. The larger entropy of S_q generally means that the corresponding distribution is more smooth. An extreme case is that a uniform distribution is the one with the maximum entropy in all possible distributions. Indeed, the estimated S_q is not a uniform distribution when it represents a ranking.

Generally speaking, there is a bias in the entropy estimation (Miller 1955, Carlton 1969, Paninski 2003, Hou, Yan, Zhang, Song & Li 2010). Assume that there is a true underlying distribution P_q, from which the actual S_q is sampled. Then, based on the bias analysis of the entropy estimation (Paninski 2003, Hou et al. 2010), we can have

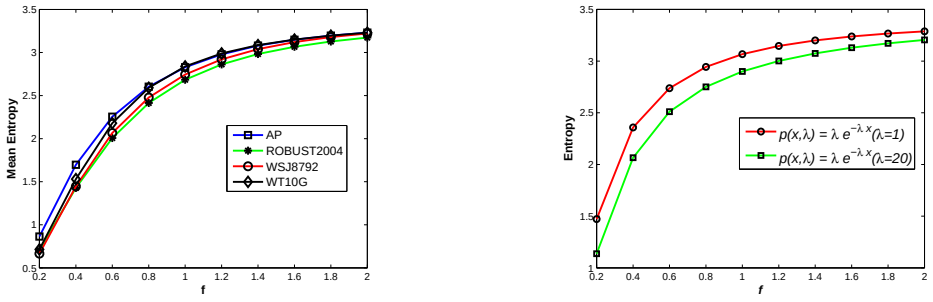
$$E_{S_q}(H(S_q)) \leq H(P_q) \tag{4.15}$$

where $E_{S_q}(H(S_q))$ shows the expectation of the entropy of the sampled distribution. The above inequation shows that the expected entropy of the sampled distribution is always smaller than the entropy of the true distribution.

One main reason about the entropy bias is that there are only limited sample data for constructing the sampled distribution S_q (Paninski 2003, Hou et al. 2010). As a result, there would be some too large probability values in the sampled S_q, making the entropy of the sampled distribution be smaller than that of the true one. One can smooth the distribution S_q to avoid these too large probability values. This can reduce the risk of the distribution and prevent "putting all eggs into the same basket".

In our problem, as we see in Table 4.1, there are also some too big document weights. This is also because that we have limited relevance evidence/data of the information need. What we have is the original query terms to represent the information need in the first-round retrieval. Therefore, some documents which have large document weights (corresponding to large similarity value between the document and the query), may not be relevant. We also mentioned that the document weights should not be different for the relevant documents with the same relevance judgements/status. To smooth document weights can to some extent solve above problems.

In Figure 4.1, we show the effect of smoothing using the entropy of the document



(a) (b)

Figure 4.1: The trend of the entropy ($y - axis$) when the $f = s$ value (x-axis) increases from 0.2 to 2 in the powers-based remodeling algorithm.

weight distribution by the QL model on TREC collections (see Figure 4.1(a)), as well as the entropy of the simulated discrete probabilities from the exponential distribution (see Figure 4.1(b)). In both cases, it shows that the more smoothing (with bigger f value in Eq.4.9) is, the larger the entropy $H(\widetilde{S_q})$ of the distribution $\widetilde{S_q}$ will be.

Indeed, we do not argue that smoothing can always help improve the accuracy of document weight and the performance of the query expansion. As we discussed in the previous chapter, we assume that a moderate smoothing can help the effectiveness of the query expansion on some collections, and the smoothing can usually help the effectiveness and stability of the query expansion when the feedback documents are all relevant. We will evaluate the effectiveness and stability of the smoothing method in the experiments reported later.

4.2 Tackling Rank-Dependent Risk

In the previous section, we described a powers-based smoothing method which smooths the document weights while preserves the original rank. The smoothness we referred to is the smoothness along the document ranking. Now, we are going to address the smoothness of the document weights among (topically) similar documents. This kind of smoothness takes into account the inter-document dependency and has shown effective re-ranking ability (Diaz 2005, Diaz 2007, Diaz 2008). The re-ranking method can tackle the rank-dependent risk as it can improve the ranking performance. We also build up a weight allocation method to re-rank the feedback documents and this method outperforms other methods for the pseudo-relevance feedback (Zhang, Song, Zhao & Hou 2010). In addition, the weight allocation method can run on top of the powers-based smoothing method.

In our method, we aim to allocate the weights of topmost-ranked $k(k<n)$ documents to the lower-ranked documents, according to the similarity between these two parts of documents. This is not only to further smooth the document weights, but also to improve

the ranks [3] of those documents which are truly relevant but have lower weights. Recall that usually the topmost-ranked documents (e.g, the first 5 documents) are more likely to be truly relevant, since the corresponding retrieval precision (e.g., P@5) is often relatively higher compared with the average precision of all the PRF documents. According to the clustering hypothesis (Tombros & van Rijsbergen 2004), the weight allocation methods, in which the allocation is actually based on the similarity value with respect to the topmost-ranked documents, could boost the weights of the truly relevant documents which may have lower initial weights. In the following, we present two weight allocation methods (WAs) with different smoothing effects. Note that in the formulation of WAs, $\widehat{S}(d)$ is the smoothed document weight. Note that we dropped the subscript q for simplicity, as the document weight allocation method is actually carried out for each query q.

4.2.1 Linear Weight Allocation (LWA)

To illustrate the basic idea, let us consider one topmost-ranked document d_t, and a lower-ranked document d_l. Our basic idea is to keep d_t's weight unchanged, and meanwhile improve d_l's weight based on the similarity between d_t and d_l, which is measured by $sim(d_l, d_t)$ [4]. Specifically, for d_l, LWA assigns it $(1 - sim(d_l, d_t))$ proportion of its own weight and $sim(d_l, d_t)$ proportion of d_t's weight, and the allocation can be formulated as:

$$\widetilde{S}_{LWA}(d_l) = (1 - sim(d_l, d_t))\widetilde{S}(d_l) + sim(d_l, d_t)\widetilde{S}(d_t) \tag{4.16}$$

where $\widetilde{S}_{LWA}(d_l)$ is the LWA weight for the d_l. For the d_t, LWA retains its own weight, meaning that $\widetilde{S}_{LWA}(d_t) = \widetilde{S}(d_t)$. Therefore, the Equation 4.16 can also represent the LWA weight of d_t due to the fact that $sim(d_t, d_t) = 1$.

Next, if considering all the k topmost documents, for any PRF document d, we have

$$\widetilde{S}_{LWA}(d) = \frac{1}{Z} \times \sum_{d_t \in M_t} (1 - sim(d, d_t))\widetilde{S}(d) + sim(d, d_t)\widetilde{S}(d_t) \tag{4.17}$$

where $\widetilde{S}_{LWA}(d)$ denotes the LWA weighting function, Z is the normalization factor, and the M_t is the set of the topmost k documents.

4.2.2 Nonlinear Weight Allocation (NLWA)

In addition to LWA, we propose a nonlinear version of weight allocation, called NLWA, which has the same basic idea as LWA. The difference between NLWA and LWA is the specific allocation strategy. For a topmost document d_t and a lower one d_l, the NLWA weights are formulated as:

$$\widetilde{S}_{NLWA}(d) = \sqrt{\widetilde{S}(d)}\sqrt{\widetilde{S}(d_t)}sim(d, d_t) \tag{4.18}$$

[3] Here, we assume that the higher rank corresponds to the higher weight.
[4] Generally, *sim* can be any similarity metric with values on [0, 1].

where d can be d_t or d_l. In a similar manner as for the LWA, if considering all the topmost documents, for any PRF document d, the NLWA weighting function is:

$$\widetilde{S}_{NLWA}(d) = \frac{1}{Z} \times \sum_{d_t \in M_t} \sqrt{\widetilde{S}(d)} \sqrt{\widetilde{S}(d_t)} sim(d, d_t) \tag{4.19}$$

4.2.3 Analyzing Difference between LWA and NLWA

Now, we analyze the different between the above two weight allocation methods by investigate how many weights the topmost documents can be allocated to the lower-ranked documents. For simplicity, our analysis is based on any two documents d_t and d_l, where d_t is ranked higher than d_l. Let $s = sim(d_t, d_l)$, $\widetilde{S}(l) = \widetilde{S}(d_l)$ and $\widetilde{S}(t) = \widetilde{S}(d_t)$, where $0 < \widetilde{S}(l) < \widetilde{S}(t)$. According to Equation 4.16, we have $\widetilde{S}_{LWA}(l) = (1-s)\widetilde{S}(l) + s\widetilde{S}(t)$, and from the Equation 4.18, we can obtain $\tilde{f}_{NLWA}(l) = s\sqrt{\widetilde{S}(l)\widetilde{S}(t)}$. Then, the quotient of d_l's LWA weight and d_l's NLWA weight is:

$$\frac{\widetilde{S}_{LWA}(l)}{\widetilde{S}_{NLWA}(l)} = \frac{1-s}{s} \sqrt{\frac{\widetilde{S}(l)}{\tilde{f}(t)}} + \sqrt{\frac{\widetilde{S}(t)}{\widetilde{S}(l)}} \tag{4.20}$$

Since $\frac{1-s}{s}\sqrt{\frac{\widetilde{S}(l)}{\widetilde{S}(t)}} > 0$ and $\sqrt{\frac{\widetilde{S}(t)}{\widetilde{S}(l)}} > 1$, we can get:

$$\frac{\widetilde{S}_{LWA}(l)}{\widetilde{S}_{NLWA}(l)} > 1 \tag{4.21}$$

It turns out that d_l's LWA weight is larger than its NLWA weight. Since d_t's weight is unchanged in both LWA and NLWA, we can conclude that LWA makes the weight difference between d_t and d_l smaller than NLWA does. In general, we observe that LWA can allocate more weights from the topmost documents to the lower-ranked documents than NLWA. Hence, the discriminativity of LWA between weights of topmost documents and weights of lower-ranked documents is smaller than that of NLWA. In (Zhang, Song, Zhao & Hou 2010), we have observed that this kind of difference can lead to different re-ranking performance. Specifically, NLWA is more effective in re-ranking the feedback documents.

4.3 Application

The applications of the proposed document weight smoothing and allocation methods include those tasks where the initial estimation of the document relevance is not the final decision. In this chapter, the task we focus on is the pseudo-relevance feedback (PRF), where the relevance estimation in the first-round retrieval can indicate feedback documents' weights used in the second-round retrieval.

Relevance Model (RM) (Lavrenko & Croft 2001) is a typical language modeling approach to a feedback-based expanded query model for the second-round retrieval. For each

query q, based on the given document set D ($|D| = n$), the RM[5] is formulated as:

$$p(w|\theta_q^{(f)}) = \sum_{d \in D} p(w|\theta_d) S_q^{QL}(d) \tag{4.22}$$

where $p(w|\theta_q^{(f)})$ is the estimated query model, $S_q^{QL}(d)$ is the document weight computed by the normalized QL score (see Eq. 4.5) of document d. The proposed document weight smoothing and allocation methods will run on the original document weight $S_q^{QL}(d)$, and then influence the query expansion and the second-round retrieval performance. In the next section, we are going to test the proposed document weight smoothing and allocation method in the context of PRF task to estimate an expanded query model.

4.4 Empirical Evaluation

4.4.1 Evaluation Configuration

The evaluation data and set-up are the same as those in the previous chapter.

Evaluation Data 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, e.g., newswire articles and 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. Lemur 4.7 (Ogilvie & 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.

Evaluation Set-up The first-round retrieval is carried out by a baseline language modeling (LM) approach, i.e., the query-likelihood (QL) model (Zhai & Lafferty 2001, Ponte & Croft 1998) in Eq. 4.2. The smoothing method for the document language model is the Dirichlet prior (Zhai & Lafferty 2001) with $\mu = 700$, which is a default setting in the Lemur toolkit, and also a typical setting for query-likelihood model.

After the first-round retrieval, the top n ranked documents are selected as the pseudo-relevance feedback (PRF) documents for the PRF task. We report the results with respect to $n = 30$. Nevertheless, we have similar observations on other n (e.g., 50, 70, 90). The Relevance Model (RM) in Eq. 4.22, is selected as the second baseline method, where the document prior is set as uniform. The number of expanded terms is fixed as 100. 1000 retrieved documents by the KL-divergence model are used for performance evaluation in both the first-round retrieval and second-round retrieval.

The Mean Average Precision (MAP), which reflects the overall rank performance, is

[5]This formulation is equivalent to RM1 in (Lavrenko & Croft 2001)

adopted as the primary evaluation metric. The Wilcoxon signed rank test is the measure of the statistical significance of the improvements over baseline methods. We will also report the performance bias-variance and estimation bias-variance of the proposed methods used in the query model estimation.

Next, we describe the evaluation procedure and parameter configurations.

Evaluation Procedure We will first evaluate the proposed document weight smoothing method, which can be considered as a Powers-based Risk Management (PRM) method (see Section 4.1.4). These methods corresponding to $f_{ND}(q)$ and $f_s(q)$ are denoted as PRM_ND and PRM_s, respectively. We will compare PRM methods (dealing with the rank-independent risk) with the state-of-the-art re-ranking methods (tackling the rank-dependent risk), which will be used to change the original document weights as well as their order in the feedback document list. These re-ranking methods include the score regulation (SR) approach (Diaz 2005, Diaz 2008) and DSDG method in (Mei et al. 2008), which are based on the clustering hypothesis, as well as the rank-related priors (RRP)[6] (Li 2008), which is to revise the document weight in RM.

Next, we will compare the proposed document weight smoothing method in Section 4.1 and the proposed document weight allocation method in Section 4.2. The latter will be run after the former method. Our aim is to see if the weight allocation methods can improve the retrieval performance of the weight smoothing method. A systematic evaluation based on the retrieval effectiveness as well the bias-variance analysis will be reported.

Parameter Configuration For the proposed smoothing method PRM_s, we test λ in [1, 4] with the increment 0.3. For PRM_ND, there is no adjustable parameter. For the proposed document weight allocation methods in Section 4.2, we test different k in [2, 5] with the increment 1, where k is the number of topmost documents used in the weight allocation. We report the result with respect to $k = 5$, while we find similar results on other k values. Recall that the weight allocation is run after the weight smoothing. Therefore, the weight smoothing will be still involved.

For SR in (Diaz 2005, Diaz 2008), we tuned three parameters: the α is in [0.1, 0.9] with the increment 0.1, the t^{-1} in [0.1, 0.9] with the step 0.1, and the number of nearest neighbor kNN in [5, 10] with the step 1. For DSDG in (Mei et al. 2008), we tuned two parameters: the λ of DSDG in [0.1, 0.9] with the increment 0.1 and the nearest neighbor kNN in [5, 10] with the step 1. The iteration number is fixed to be 3. Basically, the above parameter settings for both SR and DSDG are consistent with those in the original papers (Diaz 2005, Mei et al. 2008). The values of kNN are smaller than values used in (Diaz 2005, Mei et al. 2008), since the total number of documents in the PRF task is smaller than that in the re-ranking task. We also tested both the normalized or unnormalized QL scores for both SR and DSDG. As for the RRP, the α is set as 140 and

[6] We do not consider the other two components of the work in (Li 2008), one about automatic combination of original query model, and the other about word discounting.

Table 4.2: Overall PRF performance over 30 PRF documents.

MAP% (chg% over LM)	**WSJ8792**	**AP8889**	**ROBUST04**
LM	31.25	30.43	29.15
RM	37.01 ($+18.4^{\alpha}$)	38.10 ($+25.2^{\alpha}$)	33.26 ($+14.1^{\alpha}$)
RRP	36.76 ($+17.6^{\alpha}$)	37.54 ($+23.4^{\alpha}$)	31.56 ($+8.2^{\alpha}$)
SR	38.51 ($+23.2^{\alpha\beta}$)	38.70 ($+27.1^{\alpha}$)	34.29 ($+17.6^{\alpha}$)
DSDG	38.26($+22.4^{\alpha}$)	39.44($+29.6^{\alpha\beta}$)	34.37($+17.9^{\alpha}$)
PRM_ND	37.98 ($+21.5^{\alpha}$)	40.84 ($+34.2^{\alpha\beta}$)	34.66 ($+18.9^{\alpha\beta}$)
PRM_s	**38.67**($+23.7^{\alpha\beta}$)	**40.86**($+34.3^{\alpha\beta}$)	**34.67**($+18.9^{\alpha\beta}$)

The improvements (at significance level 0.05) over LM and RM are marked with α and β, respectively.

the β is set as 50, both the optimal values reported in (Li 2008).

4.4.2 Evaluation on Effectiveness of Powers-based Risk Management

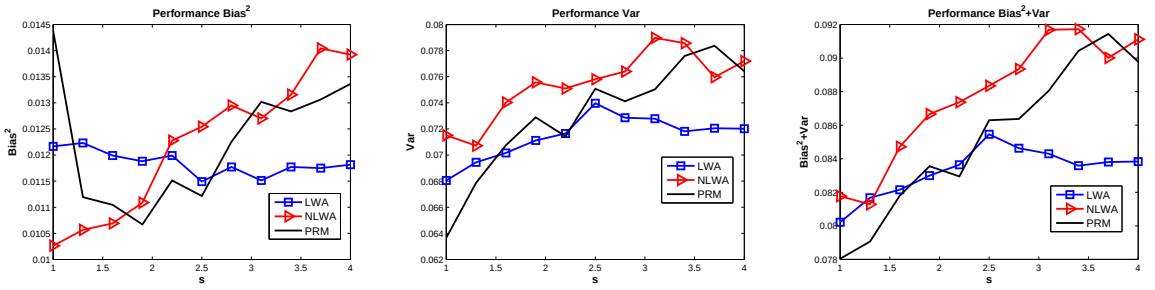
In this chapter, we investigate the rank-independent risk and regard the proposed powers-based smoothing method as a micro-adjustment for the relevance estimation, as opposed to the macro-adjustment, i.e., the re-ranking methods which can tackle the rank-dependent risk. Therefore, in this set of experiments, we are going to compare the retrieval effectiveness of the power-based risk management (PRM) methods with that of the state-of-the-art re-ranking methods. These comparative methods [7] include the score regulation (SR) (Diaz 2005), the DSDG method (Mei et al. 2008)) as well as the rank-related priors (RRP) (Li 2008). The parameters configuration is described in the previous subsection. The experimental results are summarized in Tables 4.2.

For the RRP, its performance using the reported parameters in (Li 2008) can improve RM on only one collection, i.e., WSJ8792, and the improvement is not significant. We think that this approach can may perform better when a large number (e.g., 500) of feedback documents are involved, since it effectively depress the weights of lower-ranked documents, according to its formulation in Eq. 2.13. However, if the number of feedback documents is relatively small (e.g., 30 in our settings), we observe that it can make the weights of feedback documents less smooth, and hence hurt feedback performance.

In Table 4.2, we show the best performance (among the all tested parameters) of both SR and DSDG. They are able to improve RM, but the performance improvements are not significant on most collections. This observation suggests that these re-ranking methods may not be sufficient enough to improve RM significantly in the context of pseudo-relevance feedback, where a second-round retrieval is involved and re-ranking the feedback-documents is only an intermediate step.

Now, let us look at the powers-based risk management methods which smooth the document weights while preserving the original rank. For the PRM_ND, we can observe that it can improve RM on every collection, and the improvements are significant on AP

[7] Please refer to their formulations in the literature.



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.2: Performance bias-variance Result (on WSJ8792) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA)

.

and ROBUST2004 collections. For the PRM_λ [8], it can significantly improve RM on most collections. It also outperforms the best performance of SR and DSDG. The above observations suggest that the micro-adjustment (rank-independent) smoothing methods have the ability to outperform the macro-adjustment (rank-dependent) re-ranking methods, when both are used to revise the document weights in the context of pseudo-relevance feedback.

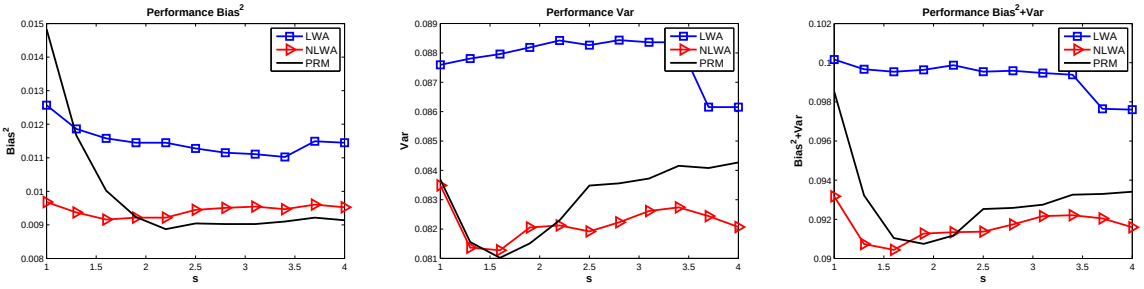
We should mention that the document weight smoothing can not be always helpful on any smoothing parameters or for any queries. Recall that we have systematically tested the smoothing methods and showed a retrieval effectiveness-stability tradeoff on a range of smoothing parameters. In the next subsection, we will evaluate the proposed weight allocation methods in terms of both retrieval effectiveness and robustness, and show the results of bias and variance analysis on a range of smoothing parameters. Note that the weight allocation methods can also be regarded as a re-ranking method. One difference between the proposed weight allocation method and the existing re-ranking methods [9] is that the former is run after the document weight smoothing methods in our experiments.

4.4.3 Evaluation on Bias-Variance of Weight Smoothing and Allocation

This set of experiments evaluates if the weight allocation methods (WAs) can further reduce the bias and variance of the weight smoothing method (denoted as PRM). The evaluation will be based on the performance bias-variance and estimation bias-variance. The lower bias and variance indicates a better effectiveness and stability, respectively. Recall that the WAs run on the smoothed document weights by the weight smoothing method (see Eq. 4.16 and Eq. 4.18). Therefore, we report the bias-variance results on

[8] In Table 4.2, we show the best performance of PRM_λ over all smoothing parameters s. We also derived an ad-hoc method to adjust the smoothing parameter adaptively for different queries. This method shows some positive results in our prior study (not reported here).

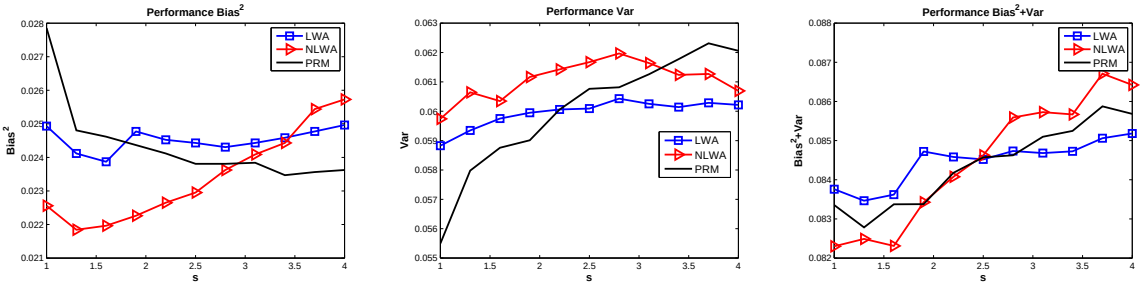
[9] For the detailed experimental comparison between them, please refer to the experiments shown in (Zhang, Song, Zhao & Hou 2010)



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.3: Performance bias-variance Result (on AP8889) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA)

.



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.4: Performance bias-variance Result (on ROBUST2004) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA)

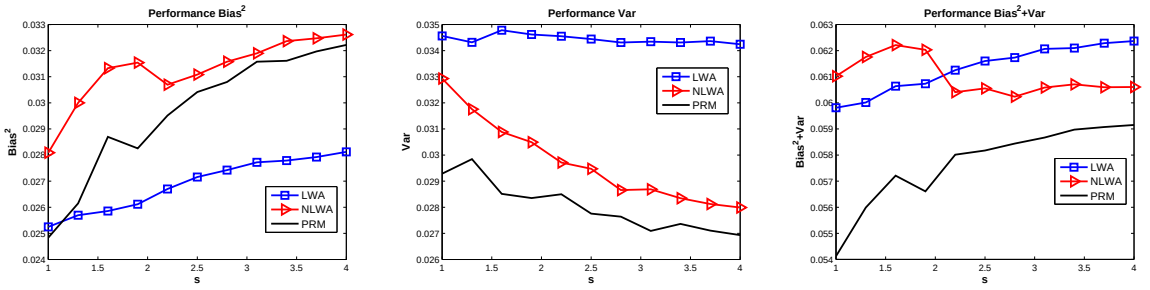
.

every smoothing parameter s ranging from [1, 4] with the increment 0.3.

Performance Bias-Variance Analysis

The performance bias-variance results are illustrated in Figures 4.2–4.5. On WSJ8792 (see Figure 4.2), NLWA can reduce the bias [10] of PRM when $s < 1.6$, and LWA can reduce the bias of PRM when $s > 2.8$. NLWA achieves the lowest bias in the corresponding figure. For the variance, LWA can be more effective to reduce the variance than NLWA. However, the smoothing method has the lowest variance. For $Bias^2+Var$, it shows that LWA can do a better job than NLWA. The PRM, however, has the smallest $Bias^2+Var$. On AP8889 (see Figure 4.3), it shows that NLWA can be more effective to reduce both bias and variance than LWA. NLWA also achieves the lowest $Bias^2+Var$. On ROBUST2004 (see Figure 4.4), NLWA can be more effective to reduce the variance than LWA, and NLWA achieves the smallest bias. For the variance, the situations are different and LWA can do better. Overall, NLWA can have smaller $Bias^2+Var$ when $s < 2.5$ but LWA has smaller

[10] In this section, we may use "reduce the bias/variance" as the simplification of "reduce the bias/variance of PRM".



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.5: Performance bias-variance Result (on WT10G) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA)

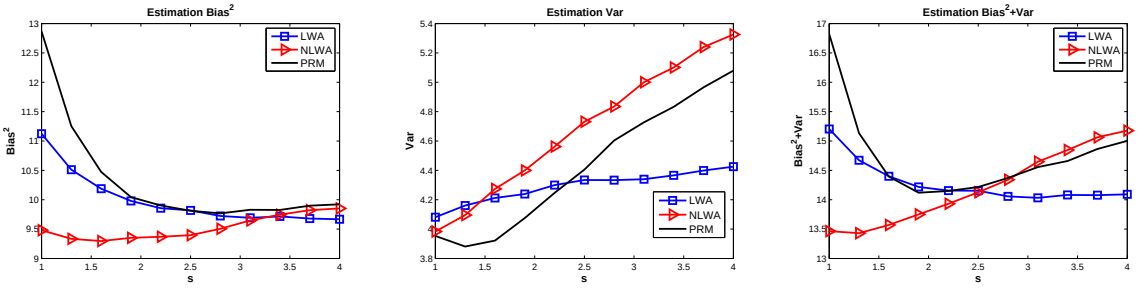
.

$Bias^2+Var$ after $s = 2.5$. NLWA can achieves the lowest $Bias^2+Var$ on ROBUST2004 in the corresponding figure. On WT10G (see Figure 4.5), it shows that LWA performs better for reducing the bias while NLWA can have smaller variance than LWA. Neither of them can reduce the $Bias^2+Var$ of the PRM. This might be because that on WT10G, the initial ranking is very poor and the topmost documents are not as relevant as we expect. Thus, the re-ranking function of the weight allocation does not help the weight smoothing and the retrieval performance of the pseudo-relevance feedback.

To sum up, both weight allocation methods can reduce the bias and/or variance of the weight smoothing method on many cases. It sometimes shows that a tradeoff between two methods, i.e., one method may perform better in reducing the bias while the other may perform better in reducing the variance. Although no single method can have dominating result, NLWA is more likely to achieve the lowest bias or the lowest $Bias^2+Var$ in the corresponding figures. We mentioned earlier that the re-ranking performance of NLWA is better than that of LWA. In addition, the lowest values are often achieved when s is relatively small. The reason could be that if s is relatively small and the document weights have not been smoothed much, then the re-ranking can be more helpful. On the other hand, if the document weights of feedback documents are smoothed to uniform, then the re-ranking will not be helpful since the topmost documents will have the same weights as the lower-ranked documents.

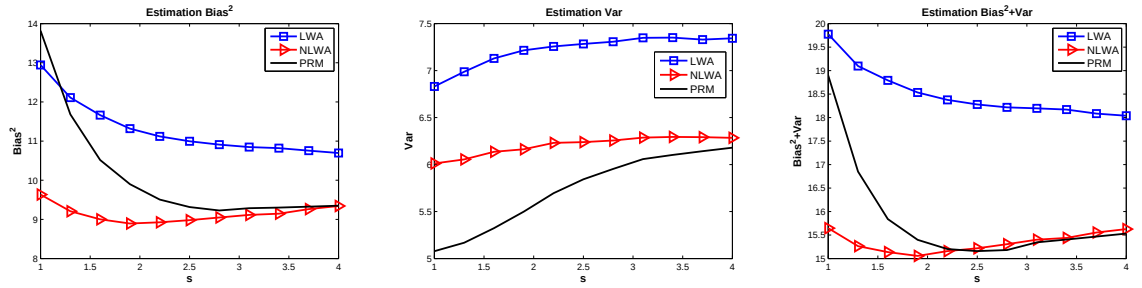
Estimation Bias-Variance Analysis

The results of the estimation bias-variance based on KL-divergence are illustrated in Figures 4.6–4.9, from which we can observe similar trends as in Figures 4.2–4.5, respectively. For example, on AP8889, the trends of all three sub-figures in Figure 4.7 are similar to those in Figure 4.3, in the sense that the lines of NLWA are all below those of LWA. On WSJ8792, the trends of variance are similar in Figure 4.6 and in Figure 4.2, where LWA performs better. When the s is small, the trends of two different kinds of bias is similar



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.6: Estimation bias-variance (on WSJ8792) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA).



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.7: Estimation bias-variance (on AP8889) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA).

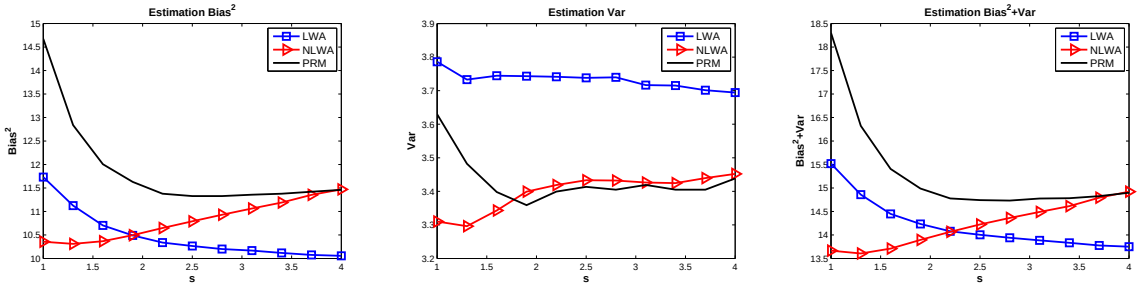
as well – NLWA's bias is smaller than LWA's. On ROBUST2004, the trends for bias are similar in Figure 4.8 and in Figure 4.4 when s is small. For $Bias^2+Var$, like the result in Figure 4.4, two lines will meet at the middle area of the range of s in Figure 4.8. On WT10G, like the result shown in Figure 4.5, we observe that NLWA is good at reducing bias in Figure 4.9. Still, no weight allocation methods can reduce the variance of the PRM. However, NLWA can reduce the $Bias^2+Var$ of the PRM on WT10G.

To sum up, the results about the estimation bias-variance also suggests that the weight allocation methods can further improve the estimation quality of the query model, since it can further reduce the bias and/or variance of the weight smoothing method denoted as PRM. Like in the performance bias-variance figure, we can observe that the NLWA is more likely to achieve the smallest estimation bias, variance and $Bias^2+Var$.

4.5 Summary

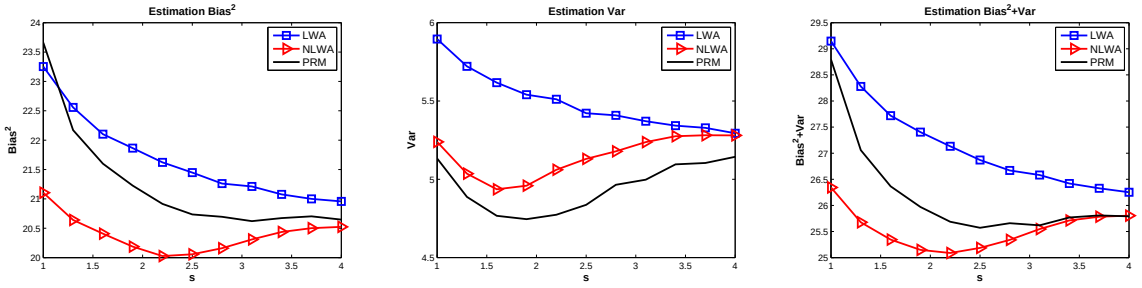
In this chapter, we investigate the rank-independent risk in estimating the document relevance. The proposed rank-independent risk management method is actually the document weight smoothing method described in the previous chapter.

In this chapter, we first show that even though two language modeling approaches



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.8: Estimation bias-variance (on ROBUST2004) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA).



(a) $Bias^2$ (b) Var (c) $Bias^2+Var$

Figure 4.9: Estimation bias-variance (on WT10G) of the weight smoothing method (PRM) and two weight allocation methods (LWA and NLWA).

(i.e., QL and ND models) are rank-equivalent, their estimated relevance distributions are different and the distribution of the ND model is more smooth than the one of the QL model. In addition, a risk management method, which is based on the powers-based remodeling idea motivated from the distribution difference (see Eq. 4.8) of QL and ND models, is proposed to generally manage the rank-independent risk for a given retrieval model. An entropy-bias explanation is provided to support the rationality of the proposed risk management method. We apply the proposed risk management method to the pseudo-relevance feedback. Experimental results on several TREC collections demonstrate the effectiveness of the proposed method against the state-of-the-art re-ranking methods that are used to tackle the rank-dependent risk in pseudo-relevance feedback.

Based on the document weight smoothing method, we also propose two weight allocation methods (linear and nonlinear), which can tackle the rank-dependent risk by re-ranking the feedback documents. We constructed a systematic bias-variance evaluation, which shows that the weight allocation methods are able to further improve the weight smoothing method, evidenced by the result that the weight allocation methods are able to further reduce the performance bias-variance and estimation bias-variance over the weight smoothing methods. Compared with the linear counterpart, the nonlinear weight

allocation methods are more likely to achieve the lowest bias and/or variance values.

Chapter 5

Distribution Separation Method - Removing Irrelevant Distribution

According to the bias-variance analysis described earlier, removing irrelevant documents is another important factor to approximate the true relevance model. Based on both theoretical analysis and experimental results in Chapter 3, this factor is more important than smoothing document weights in terms of reducing bias and variance simultaneously. In this chapter, we will investigate it in depth by further exploring two research problems associated to removing irrelevant documents.

The first research problem is: can we go beyond the document level and remove the irrelevant term distribution directly from the mixture term distribution obtained by relevance model (RM in Eq. 3.41)? Recall that the pseudo-relevant feedback (PRF) document set D in RM is often a mixture set of both relevant and irrelevant documents. Thus, RM yields a mixture distribution of both relevant and irrelevant terms. In this thesis, we propose a novel Distribution Separation Method (DSM) to separate the true relevance distribution from the mixture one, by removing the irrelevance distribution.

DSM is based on a linear combination assumption, which assumes that the mixture distribution is a linear combination between the relevant distribution and the irrelevant distribution. This assumption is generally valid in our problem. In Eq. 3.41, the mixture distribution by RM based on the whole document set D is a linear combination between the relevant and irrelevant distributions, which are corresponding to two partitions of the whole document set D (see more details in Section 5.2.1). In addition, the linear combination assumption is also consistent with the fact that linear combination is a commonly used technique in IR. Based on the linear combination assumption, the proposed DSM can obtain the relevant distribution, by automatically identifying the combination coefficient of the relevant distribution in the mixture distribution.

The second research problem is that in practice we often do not have the irrelevant distribution corresponding to all the irrelevant documents in D. Users in general may

be constrained or reluctant to provide so much explicit feedback information to identify all irrelevant documents. Therefore, we should endeavor to obtain a *seed* irrelevant term distribution. One way is via explicit relevance feedback. Specifically, we can assume that a small number (or a small portion, e.g., 10%) of irrelevant ones in D can be obtained via explicit relevance feedback. In addition to explicit feedback, another solution is to automatically identify a small number of irrelevant documents . To this aim, we adopt outlier detection methods by treating the irrelevant documents as outliers. Moreover, the seed irrelevant term distribution can be derived directly by detecting outlier terms. For example, we can get outlier terms that are far away from the query terms, based on the term position in the text, or based on the term similarity.

Now, we have two distributions to start with: a mixed distribution and a seed irrelevant distribution. Then, the research question is: Given the mixed distribution and the seed irrelevance distribution, how to automatically derive an optimal approximation of the true relevance distribution? To this aim, we need to assume that the seed irrelevance distribution and the optimal relevance distribution have a minimum correlation [1]. Imagine that if the irrelevance distribution had strong positive correlation with the relevance distribution, then the irrelevance distribution would have resulted in a good retrieval performance.

Based on these two assumptions (linear combination and minimum correlation), a unified framework for distribution separation is proposed and theoretical justifications of the proposed algorithm are given. We systematically evaluate the effectiveness of the proposed DSM on several large-scale TREC data sets. Evaluation results from extensive experiments demonstrate the effectiveness of our method. Our approach outperforms the RM for pseudo relevance feedback, as well as the use of RM on D with the seed negative documents directly removed.

Our approach is distinct in the following aspects:

(1) It uses the distribution separation idea to make full use of the irrelevant data, in order to approximate the true relevant information.

(2) The proposed distribution separation method can automatically get the right/optimal combination coefficient of the true relevance distribution in the mixture distribution, subject to the amount of irrelevant data we can have.

(3) It deals with separating probabilistic distributions directly, thus is more general and applicable to many other cases where the seed irrelevance information is available in other forms (e.g., terms excluded from previous queries in a query modification history) that can be converted into a probability distribution.

(4) It is feasible in practice as it requires only a seed irrelevant distribution to achieve an effective and stable performance;

(5) It can include the scenarios when no explicit feedback data is available, and then one

[1] This minimum correlation assumption is similar to the assumption in Independent Component Analysis (ICA) (Hyvärinen & Oja n.d.).

Table 5.1: 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

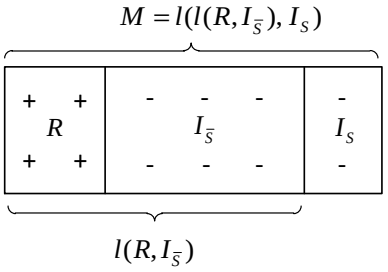


Figure 5.1: An illustration of the linear combination $l(\cdot,\cdot)$ between two distributions. "+" and "−" stand for the relevance and irrelevance, respectively.

can automatically find the seed irrelevant documents or seed irrelevant term distribution.

5.1 The Distribution Separation Method (DSM)

5.1.1 Notations and Task Definition

Table 5.1 lists some major notations used in our paper. For simplicity the $l(\cdot,\cdot)$ omits the specific linear coefficient. Figure 5.1 illustrates how the distribution M w.r.t. the pseudo-relevant document set, is a linear combination of two distributions I_S and $l(R, I_{\overline{S}})$ w.r.t two partitions of the whole set, where $l(R, I_{\overline{S}})$ is also a linear combination of R and $I_{\overline{S}}$.

Our task can be then defined as follows: given the mixture distribution M and a seed irrelevance distribution I_S, find a R^* which approximates the R. To this aim, it raises three problems:

1) How to separate the distribution $l(R, I_{\overline{S}})$, which is less noisy but is still a mixture of the true relevance and the unknown irrelevance distributions, from the seed irrelevance distribution I_S in M?

2) How to further find an optimal R^* that approximates R as closely as possible?

3) How to design a framework which can comprise previous steps and have a unified theoretical explanation?

5.1.2 Deriving a Less Noisy Distribution $l(R, I_{\overline{S}})$

Now, given the mixture distribution M and the seed irrelevance distribution I_S, we need to solve the less noisy distribution $l(R, I_{\overline{S}})$ (see Figure 5.1). Recall that M is a nested linear combination $l(l(R, I_{\overline{S}}), I_S)$, which can be represented as [2]:

$$M = \lambda \times l(R, I_{\overline{S}}) + (1 - \lambda) \times I_S \tag{5.1}$$

where λ ($0 < \lambda \leq 1$) is the (real) linear coefficient w.r.t. the desired distribution $l(R, I_{\overline{S}})$.

The problem to obtain $l(R, I_{\overline{S}})$ only based on Eq. 5.1, however, is not well-posed in the sense that it 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. According to Eq. 5.1, we have

$$\hat{l}(R, I_{\overline{S}}) = \frac{1}{\hat{\lambda}} \times M + (1 - \frac{1}{\hat{\lambda}}) \times I_S. \tag{5.2}$$

There can be infinite possible choices of $\hat{\lambda}$ and its corresponding $\hat{l}(R, I_{\overline{S}})$. To obtain a $\hat{\lambda}$ which can estimate the real coefficient λ as well as possible, we introduce a constraint

$$\hat{l}(R, I_{\overline{S}}) \succcurlyeq 0, \tag{5.3}$$

which means that all the values in distribution $\hat{l}(R, I_{\overline{S}})$ should be not less than 0. Based on Eq. 5.2 and Eq. 5.3, we have

$$\frac{1}{\hat{\lambda}} \times M \succcurlyeq (\frac{1 - \hat{\lambda}}{\hat{\lambda}}) \times I_S.$$

Then, we get

$$\hat{\lambda} \succcurlyeq (\mathbf{1} - M./I_S) \tag{5.4}$$

where ($\mathbf{1}$ is 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 $\hat{\lambda} > 1 - \infty$, which is still hold since $\hat{\lambda} > 0$.

The Eq. 5.4 sets a lower bound of $\hat{\lambda}$:

$$\lambda_L = \max(\mathbf{1} - M./I_S) \tag{5.5}$$

where $\max(\cdot)$ denotes the max value in the resultant vector $\mathbf{1} - M./I_S$. This 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. We will discuss it in-depth in

[2] In Eq. 5.1, M, I_S and $l(R, I_{\overline{S}})$ are vectors in which the i^{th} entry is the probability of the i^{th} term

the rest of the paper. We first investigate how the reduction of $\hat{\lambda}$ affects its corresponding $\hat{l}(R, I_{\overline{S}})$ in Lemma 1. For simplicity, we use some simplified notations listed in Table 5.2.

Table 5.2: 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 $\hat{\lambda}$)

Lemma 1. *If $M(i) < I_S(i)$ for term i, then $\hat{\lambda}_1 < \hat{\lambda}_2$ implies that $\hat{l}_1(i) < \hat{l}_2(i)$, and vice versa.*

Proof. Suppose that $M \neq I_S$. Since both M and I_S are distributions and sum to 1, we can always find a term i that satisfies $M(i) < I_S(i)$ [3]. According to Eq. 5.2, we can get:

$$\hat{l}_1(i) - \hat{l}_2(i) = (\frac{1}{\hat{\lambda}_1} - \frac{1}{\hat{\lambda}_2}) \times (M(i) - I_S(i)) \tag{5.6}$$

This equation means that $\hat{l}_1(i) - \hat{l}_2(i)$ and $\hat{\lambda}_1 - \hat{\lambda}_2$ have the same sign. Hence, if $\hat{\lambda}_1 < \hat{\lambda}_2$, then accordingly $\hat{l}_1(i) < \hat{l}_2(i)$, and vice versa. □

Lemma 1 basically tells us that if $M(i) < I_S(i)$, then the reduction of λ can result in the reduction of $\hat{l}(i)$. However, it does not tell us which terms can guarantee the condition $M(i) < I_S(i)$. Now we introduce an underlying situation when there exists a zero value in the desired distribution $l(R, I_{\overline{S}})$. We will show that in this case, the corresponding term i with zero value (i.e., $l(i) = 0$) ensures $M(i) < I_S(i)$.

Lemma 2. *Given a term i, $l(i) = 0$ can entail that $0 < M(i) < I_S(i)$.*

Proof. We will show that if $M(i) < I_S(i)$ is not true, other cases about the relation between $M(i)$ and $I_S(i)$ are not applicable or possible in our task.

First, if $M(i) = I_S(i) = 0$, then it means the term i does not appear in the documents or collections we considered. Therefore, we can ignore term i in distribution M [4] and in our task. Actually, this case is not applicable for our task.

Second, if $M(i) = I_S(i) > 0$, according to Eq. 5.1 $l(i)$ will equal to $M(i)$, which is bigger than 0. This contradicts the precondition that $l(i) = 0$.

Third, if $M(i) > I_S(i)$, according to Eq. 5.1, which shows that $M(i)$ is a linear combination between $I_S(i)$ and $l(i)$ and $0 < \lambda \leq 1$. This means $M(i)$ is a value between $I_S(i)$ and $l(i)$, implying that $l(i) > M(i) \geq 0$. It turns out that $l(i) > 0$, which also contradicts the precondition that $l(i) = 0$.

In summary, $l(i) = 0$ entails $0 < M(i) < I_S(i)$. □

[3] In this subsection, we only consider the case when $M(i) < I_S(i)$. Other cases will be discussed later.

[4] We do not ignore the terms which only satisfies $l(i) = 0$ or $I_S(i) = 0$, because they may appear in the considered documents.

In pseudo-relevance feedback, the zero values often exist in $l(R, I_{\overline{S}})$ if there is no smoothing step involved. We will also consider the case when the smoothing step is involved later. Note that in the estimation of λ, we do not need to know whether or not there is a zero value in the desired distribution. The involvement of this underlying condition is mainly to show the property of the lower bound λ_L.

Next, we present an important property of λ_L in Lemma 3, which guarantees if there exists zero value in $l(R, I_{\overline{S}})$, then $\lambda = \lambda_L$. In this case, $\lambda = \lambda_L$, meaning that the distribution $l_L(R, I_{\overline{S}})$ *w.r.t.* λ_L is the desired distribution $l(R, I_{\overline{S}})$ *w.r.t.* λ.

Lemma 3. *If there exists a zero value in $l(R, I_{\overline{S}})$, then $\lambda = \lambda_L$, leading to $l(R, I_{\overline{S}}) = l_L(R, I_{\overline{S}})$.*

Proof. Let probability of the term i in $l(R, I_{\overline{S}})$ be zero, i.e., $l(i) = 0$. According to Lemma 2, it turns out means $0 < M(i) < I_S(i)$. Based on Eq. 5.1, we have $M(i) = \lambda l(i) + (1 - \lambda) I_S(i)$. Recall that $l(i) = 0$. It turns out that $\lambda = 1 - M(i)/I_S(i)$.

This λ is actually the lower bound λ_L in Eq. 5.5. If λ is not the lower bound, then one can reduce λ. According to Lemma 1, reducing λ will make $l(i)$ also reduced to negative value. This contradicts that $l(i) = 0$;

Therefore, $\lambda = \lambda_L$ and correspondingly $l(R, I_{\overline{S}}) = l_L(R, I_{\overline{S}})$, since the value of the coefficient determines the corresponding distribution based on Eq. 5.2. □

Let us consider a simple example. Suppose that in Eq. 5.1, M =[0.16, 0.12, 0.18, 0.22, 0.06, 0.26]^T, I_S = [0.2, 0, 0.1, 0.3, 0.1, 0.3]^T, $\lambda = 0.4$ and correspondingly $l(R, I_{\overline{S}})$ = [0.1, 0.3, 0.3, 0.1, 0, 0.2]^T, where a zero value exists. Now, given M and I_S only, according to Eq. 5.5, we can get $\lambda_L = 0.4$, which equals to λ. Based on this coefficient and Eq. 5.2, we can then have $l_L(R, I_{\overline{S}})$ which is actually $l(R, I_{\overline{S}})$.

Now, we consider the case when a smoothing step is involved. In PRF background, after applying smoothing (usually with the collection model), there will not be zero values, but instead a lot of small values exist, in $l(R, I_{\overline{S}})$. In this context, Remark 1 guarantees the approximate equality between λ_L and λ.

Remark 1. *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 $l(R, I_{\overline{S}})$.*

Since λ_L is the lower bound, then $\lambda_L \leq \lambda$. Also because there exist zero elements in $l_L(R, I_{\overline{S}})$, then $\lambda_L \neq \lambda$, implying that $\lambda_L < \lambda$.

However, if λ_L is not close to λ, according to Lemma 1, a few small values in $l_L(i)$ are likely to be negative. This violates the fact that all $l_L(i)$ are not less than zero. Thus λ_L should be quite close to λ. According to Lemma 1, $l_L(R, I_{\overline{S}})$ should then be approximately equal to $l(R, I_{\overline{S}})$. Therefore, the proper estimate of $l(R, I_{\overline{S}})$ can be $l_L(R, I_{\overline{S}})$.

5.1.3 Approximating the True Relevance Distribution R

Using the above strategy, a less noisy but still mixed distribution $l(R, I_{\overline{S}})$ can be derived. In this section, we investigate how to compute a distribution R^* that can approximate the true relevance distribution R. Our method is based on the following two observations on the difference between $l(R, I_{\overline{S}})$ and R.

1) The linear coefficient of R w.r.t M ($M = l(R, I)$) is less than that of $l(R, I_{\overline{S}})$ w.r.t M ($M = l(l(R, I_{\overline{S}}), I_S)$). This means that the corresponding linear coefficient λ should be reduced accordingly for R.

2) If the pseudo feedback documents do not share the same or a very similar distribution with each other, which is true in most cases, then the true relevance distribution R should be less smooth than the mixture distribution $l(R, I_{\overline{S}})$.

To further explain the first observation, let us consider a simple example. Assume the probability of each term w in a specific document set D_F (e.g., D_F could be D or D_{I_S}) is defined as

$$F(w) = tf(w, D_F)/tf(D_F), \tag{5.7}$$

where $tf(w, D_F)$ denotes the number of times of w occurring in D_F, and $tf(D_F) = \sum_w tf(w, D_F)$. Then, it turns out

$$M = \frac{N_F}{N} \times F + \frac{N_G}{N} \times G \tag{5.8}$$

where N, N_F and N_G are the numbers of terms in D ($D = D_F \bigcup D_G$), D_F and D_G, respectively; M, F and G are computed by Eq. 5.7 on document sets D, D_F and D_G, respectively. From this example, we can find that the numbers of terms, e.g., N_F or N_G, determine the linear coefficient of F or G, respectively. Therefore, if we generate distribution using Eq. 5.7, since the number of terms in R is less than the number in $l(R, I_{\overline{S}})$, the linear coefficient of R should be less than that of $l(R, I_{\overline{S}})$. More generally, this observation can still hold by just imagining that the linear coefficient of R or $l(R, I_{\overline{S}})$ w.r.t. M can be equivalent to the weights of them w.r.t. M.

As for the second observation, it is also the number of terms and their distribution that play a key role. If each top-ranked document does not have the same distribution with each other, the more terms will usually mean a more smooth distribution. This is also an explanation why in many methods the distributions are smoothed by the collection model.

Now, given the above two observations, the problem is how to refine the distributions M and I_S by reducing terms properly, in order to further compute a relevance distribution R^* that can bridge the gap between $l(R, I_{\overline{S}})$ and R. To this end, we propose a strategy

as follows: If any term i meets the following condition:

$$\frac{M(i)}{I_S(i)} < 1 - \lambda_L \times \eta \tag{5.9}$$

then, term i will be deleted in both M and I_S, where λ_L is the pervious lower bound, and $\eta < 1$ is a parameter in our model to control the refinement step. This refinement step can be explained in an intuitive way. Specifically, if term i satisfies Eq. 5.9, it means that $M(i)$ is very small while $I_S(i)$ is relatively large, and then we can safely consider this term as irrelevant.

After this refinement, we let the current $l_L(R, I_{\overline{S}})$, which is computed by Eq. 5.2 using current M, I_S and λ_L, be the solution to the relevance distribution R^*. Next, we will describe that λ_L has been reduced, and accordingly the current $l_L(R, I_{\overline{S}})$ becomes less smooth than the previous one.

To demonstrate the reduction of λ_L, we will consider two cases. First, if current M and I_S are not normalized, according to Eq. 5.5 and 5.9, the current λ_L is not greater than previous λ_L multiplied by η. Second, if current M and I_S are normalized, we need to normalize the previous $l_L(R, I_{\overline{S}})$ only concerned with remaining terms, and compute its linear coefficient w.r.t. M. This coefficient is also larger than current λ_L, which is the lower bound of all possible linear coefficients. In other words, λ_L has been reduced.

For a better clarity, let us look at the example before Lemma 3. Here, we set $\eta = 0.6$ to control the refinement. According to Eq. 5.9, the fourth and fifth terms in M and I_S will be deleted. After this refinement, if M and I_S are not normalized, the current λ_L will be 0.2 which is less than the previous λ_L 0.4, and the current $l_L(R, I_{\overline{S}})$ is [0, 0.6, 0.5, 0.1]^T which is equivalent to [0, 0.5, 0.4167, 0.0833]^T after normalization. If M and I_S are normalized, the current λ_L is 0.3333, and the corresponding $l_L(R, I_{\overline{S}})$ is still [0, 0.5, 0.4167, 0.0833]^T; the previous $l_L(R, I_{\overline{S}})$ with remaining terms is [0.1111, 0.3333, 0.3333, 0.2222]^T after normalization. By slightly changing Eq. 5.2, we can get the linear coefficient of this previous $l_L(R, I_{\overline{S}})$ w.r.t. M is 0.5, which is greater than the current λ_L, 0.3333.

With regard to the less smoothness of current $l_L(R, I_{\overline{S}})$, firstly, it involves less number of terms than the previous one. Secondly, even for the remaining terms, the current $l_L(R, I_{\overline{S}})$ is still less smooth than the previous one (see the above example). Now, only considering these remaining terms in current normalized M and S, we propose Lemma 4 to demonstrate that along with reducing $\hat{\lambda}$ ($\lambda_L < \hat{\lambda} \leq 1$), the probability values in the corresponding $\hat{l}(R, I_{\overline{S}})$ will approach to 0 or 1 (see Figure 5.2). In this sense, the current $l_L(R, I_{\overline{S}})$ should be less smooth than the previous one since the former corresponds to a smaller linear coefficient.

Lemma 4. *The more we reduce $\hat{\lambda}$ ($\lambda_L < \hat{\lambda} \leq 1$), the more values in the corresponding $\hat{l}(R, I_{\overline{S}})$ approaches to 0 or 1.*

Proof. This proof is based on Lemma 1. When $M(i) < I_S(i)$, if $\hat{\lambda}_1 < \hat{\lambda}_2$, then the

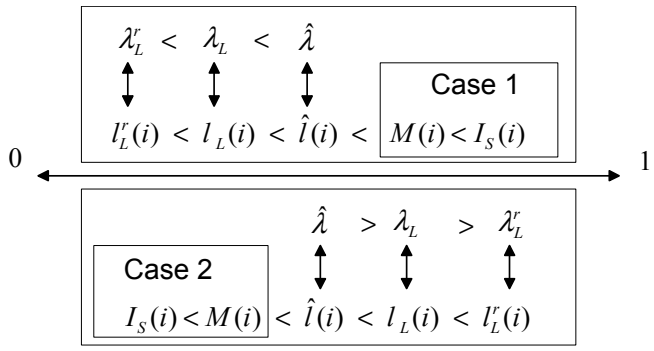


Figure 5.2: The effect of reducing $\hat{\lambda}$ ($\lambda_L < \hat{\lambda} \leq 1$) on the corresponding $\hat{l}(R, I_{\overline{S}})$ computed by Eq. 5.2.

corresponding $\hat{l}_1(i) < \hat{l}_2(i)$. Similarly, when $M(i) > I_S(i)$, from Eq. 5.6, it turns out that if $\hat{\lambda}_1 < \hat{\lambda}_2$, then $\hat{l}_1(i) > \hat{l}_2(i)$.

Now, we can keep reducing $\hat{\lambda}$ ($\hat{\lambda}_1$ or $\hat{\lambda}_2$) (see Figure 5.2). When $M(i) < I_S(i)$, it turns out that the more we reduce $\hat{\lambda}$, the more $\hat{l}(i)$ approaches to 0. On the other hand, when $M(i) > I_S(i)$, the more we reduce $\hat{\lambda}$, the more $\hat{l}(i)$ approaches to 1; □

Till now, we can conclude that $l_L(R, I_{\overline{S}})$ is less smooth than before and corresponds to a smaller linear coefficient. This complies with the two observations mentioned earlier in this subsection. To sum up, it is expected that if η in Eq. 5.9 is properly adjusted, the refined $l_L(R, I_{\overline{S}})$ can be R^* that approximates R.

5.1.4 A Unified Framework

We have shown the important role of the lower bound λ_L and the corresponding $l_L(R, I_{\overline{S}})$. Indeed, they have good properties as discussed in the previous sections. However, an underlying risk is that $l_L(R, I_{\overline{S}})$ (a $\hat{l}(R, I_{\overline{S}})$) could go to be singular (i.e., too unsmooth). A subsequent problem is: which criterion can we adopt to control this risk and choose the proper $\hat{l}(R, I_{\overline{S}})$?

From Figure 5.2, we observe that for any[5] term i , the more we reduce $\hat{\lambda}$, the more $\hat{l}(i)$ will be further away from $I_S(i)$. Intuitively, this can make the correlation between $\hat{l}(R, I_{\overline{S}})$ and I_S smaller. In this paper, we propose to use Pearson product-moment correlation coefficient (Rodgers & Nicewander 1988) ρ ($-1 \leq \rho \leq 1$), which is a standard correlation measurement, as the criterion to control the linear coefficient $\hat{\lambda}$. At first, we will analyze how this correlation coefficient changes while reducing $\hat{\lambda}$, by Proposition 2.

Proposition 2. *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.*

[5]Here, we do not consider the case when $M(i) = I_S(i)$

Proof. Let $E(M) = E(I_S) = E(\hat{l}(R, I_{\overline{S}})) = \frac{1}{m}$, where m is the number of terms concerned. Let $a = \sum_i^m (I_S(i) - \frac{1}{m})(M(i) - I_S(i))$, $b = \sum_i^m (I_S(i) - \frac{1}{m})^2$ and $c = \sum_i^m (M\ (i) - I_S(i))^2$. Without loss of generality, let $a \neq 0$, $b > 0$ and $c > 0$; and assume $\rho(\hat{l}(R, I_{\overline{S}}), I_S) = 0$ if $\hat{l}(R, I_{\overline{S}}) = [\frac{1}{m}, \frac{1}{m}, \ldots, \frac{1}{m}]^T$.

From the definition of correlation coefficient, we obtain

$$\begin{aligned}&\rho(\hat{l}(R, I_{\overline{S}}), I_S)\\ &= \frac{\sum_i^m (\hat{l}(i) - E(\hat{l}(R, I_{\overline{S}})))(I_S(i) - E(I_S))}{\sqrt{\sum_i^m (\hat{l}(i) - E(\hat{l}(R, I_{\overline{S}})))^2}\sqrt{\sum_i^m (I_S(i) - E(I_S))^2}}\end{aligned}$$

To facilitate the proof, let $\xi = 1/\hat{\lambda}$ and $\rho(\xi) = \rho(\hat{l}(R, I_{\overline{S}}), I_S)$. By expanding $\hat{l}(i)$ based on Eq. 5.2, it turns out that

$$\rho(\xi) = \frac{a\xi + b}{\sqrt{b}\sqrt{c\xi^2 + 2a\xi + b}} \tag{5.10}$$

The derivative of $\rho(\xi)$ w.r.t. ξ is

$$\rho'(\xi) = \frac{(a^2 - bc)\xi}{\sqrt{b}(c\xi^2 + 2a\xi + b)^{\frac{3}{2}}} \tag{5.11}$$

By Cauchy-Schwarz inequality, $(a^2 - bc) \leq 0$. Hence it turns out $\rho'(\xi) \leq 0$. When $(a^2 - bc) < 0$, $\rho(\xi)$ is strictly monotonically decreasing with increasing ξ.

When $(a^2 - bc) = 0$, $\xi = -\frac{b}{a}$ is a discontinuity point in $\rho(\xi)$, since $c\xi^2 + 2a\xi + b = 0$, which means $\hat{l}(R, I_{\overline{S}}) = [\frac{1}{m}, \frac{1}{m}, \ldots, \frac{1}{m}]^T$, leading to $\rho(\hat{l}\ (R, I_{\overline{S}}), I_S) = 0$ based on the assumption of this proof. If $\xi \neq -\frac{b}{a}$, from Eq. 5.10, we can get $\rho(\xi) \in \{1, -1\}$. If $a > 0$, then $\rho(\xi) = 1$ ($\xi > -\frac{b}{a}$, equivalently $\xi > 0$); and $\xi < -\frac{b}{a}$ is not in the domain of ξ. If $a < 0$, then $\rho(\xi) = 1$ ($0 < \xi < -\frac{b}{a}$) and $\rho(\xi) = -1$ ($\xi > -\frac{b}{a}$).

In conclusion, $\rho(\xi)$ decreases when ξ increases. Therefore, $\rho(\hat{l}(R, I_{\overline{S}}), I_S)$ is decreasing with decreasing $\hat{\lambda}$ ($\hat{\lambda} = \frac{1}{\xi}$). □

According to Proposition 2, 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 minimum correlation coefficient could be negative. However, we think that negative correlation does not often exist between true relevance distribution R and irrelevance distribution I_S, especially in the PRF context. In general, most terms in R are independent of those in I_S, while some terms are expected to be positively correlated, such as query terms and common terms. Only a small number of terms may be negatively correlated.

Therefore, it is natural to change the minimum correlation coefficient (i.e., $\min(\rho)$) to minimum correlation (i.e., $\min(\rho^2)$), in order to avoid negative coefficients and control the singularity of $\hat{l}(R, I_{\overline{S}})$. This idea can be formulated as the following optimization problem:

Algorithm 1 Framework for Distribution Separation

Input: Distribution: mixed M, seed Irrelevant I_S
Output: Approximately true relevance distribution: R^*
Parameter: η $(0 < \eta \leq 1)$
Step 1: Compute the initial λ_L using Eq. 5.5 based on input M and I_S.
Step 2: Given η, according to Eq. 5.9, refine M, I_S, and λ_L.
Step 3: Based on the refined M, I_S, and λ_L, solve the optimization problem in Eq. 5.12, and get the optimal λ^*.
Step 4: Using λ^*, obtain the R^* based on Eq. 5.2.

$$\begin{aligned} &\min_{\hat{\lambda}} [\rho(\hat{l}(R, I_{\overline{S}}),\ I_S)]^2 \\ &s.t. \quad \lambda_L \leq \hat{\lambda} \leq 1 \end{aligned} \tag{5.12}$$

To solve this optimization problem, we need to first solve such a $\hat{\lambda}$ that the corresponding $\rho(\hat{l}(R, I_{\overline{S}}),\ I_S) = 0$. According to the proof in Proposition 2, this $\hat{\lambda} = -\frac{a}{b}$. Then, we need to check whether $\lambda_L \leq -\frac{a}{b} \leq 1$ holds. If it holds, the optimal linear coefficient λ^* for the optimization problem in Eq. 5.12 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 λ^*. The corresponding $l^*(R, I_{\overline{S}})$, called the optimal R^*, is the relevance distribution obtained by our model.

Now, we will present a unified framework of our distribution separation method (DSM) in Algorithm 1. When $\eta = 1$ in DSM, according to Eq. 5.9, there is no refinement step for M, I_S and λ_L. By contrast, when $\eta < 1$, DSM involves refinement. Note that the refinement cannot be performed alone, i.e., after refinement, the distribution separation (see steps 3 and 4 in Algorithm 1) will still need to be involved.

Compared with $\min(\rho)$, the objective function $\min(\rho^2)$ in Eq. 5.12 can make our model more general and also control the risk of reducing λ_L due to the facts that: 1) if an unreasonable λ_L (i.e., leading to a negative correlation) occurs, $\min(\rho^2)$ can result in a $\lambda^* \in [\lambda_L, 1]$ that has a minimum but non-negative correlation; 2) if λ_L does not deduce a negative correlation, then, λ_L, corresponding to $\min(\rho)$, will also be the solution of $\min(\rho^2)$; in this case, λ_L is equivalent to the one computed in Section 5.1.2 ($\eta = 1$) or in Section 5.1.3 ($\eta < 1$), and so does its corresponding relevance distribution.

Practically, experimental results about the two objective functions are similar. We observed that the negative correlation value seldom exists when the refinement step in not involved. In our experiments, when the refinement step is not involved, the λ^* computed by DSM usually equals to the lower bound λ_L.

In summary, DSM can encompass the different strategies discussed earlier. In the next section, we will report the experimental results on this unified framework.

5.2 Formulation of Utilized Distributions

In this section, we are going to present the specific formulations of the distributions utilized in evaluating the proposed distribution separation method. Most of the distributions are based on the Relevance Model (RM in Eq. 2.7).

5.2.1 Linear Combination of Distributions

We first describe the linear combination assumption by formulating the mixed distribution, relevance distribution and irrelevant distribution obtained by Relevance Model (RM) in the context of relevance feedback.

The term distribution derived by RM is actually a mixed distribution M corresponding to all the pseudo-relevance feedback documents D. Specifically, the mixed distribution M by RM can be formulated as:

$$p(w|M) = \sum_{d \in D} p(w|\theta_d) \frac{p(q|\theta_d)}{Z_M} \tag{5.13}$$

where $p(q|\theta_d)$ is the query likelihood (QL) score, and $Z_M = \sum_{d' \in D} p(q|\theta_{d'})$ is the summed QL scores over all documents in D. In this formulation, the document prior is uniform, which is often assumed in RM.

The true relevance distribution R should be derived from all the relevant feedback documents D_R:

$$p(w|R) = \sum_{d \in D_R} p(w|\theta_d) \frac{p(q|\theta_d)}{Z_R} \tag{5.14}$$

where $Z_R = \sum_{d' \in D_R} p(q|\theta_{d'})$. Note that the true relevance distribution in this chapter is slightly different from the estimation of the true relevance model described in Chapter 3. The difference is on the document weight. We do not smooth the document weight in Eq. 5.14 to focus on the effect of removing irrelevant documents on RM. In the bias-variance evaluation described in the experiments (Section 5.3), we will still use the upper-bound performance of true query model described in the Chapter 3.

In addition to distribution R in Eq. 5.14, we can obtain the irrelevant distribution I:

$$p(w|I) = \sum_{d \in D_I} p(w|\theta_d) \frac{p(q|\theta_d)}{Z_I} \tag{5.15}$$

where $Z_I = \sum_{d' \in D_I} p(q|\theta_{d'})$ and D_I corresponds to all the irrelevant documents in D. Now, we can observe the linear combination as follows:

$$p(w|M) = \frac{Z_R}{Z_M} p(w|R) + \frac{Z_I}{Z_M} p(w|I) \tag{5.16}$$

It turns out that

$$M = \frac{Z_R}{Z_M} R + \frac{Z_I}{Z_M} I \tag{5.17}$$

which shows that the mixed distribution M is a linear combination between the relevance distribution R and the irrelevance distribution I. The linearity can be seen by the fact that $\frac{Z_R}{Z_M} + \frac{Z_I}{Z_M} = 1$. Note that the linear combination assumption will be still hold even when the document weights in deriving each above distribution are smoothed.

5.2.2 Obtaining Seed Irrelevant Distribution

In practice, we often do not have the irrelevant distribution corresponding to all the irrelevant documents D_I in feedback document set D. As discussed earlier, we need to obtain *seed* irrelevant documents/distribution.

Explicit Relevance Feedback

We can assume that a small number of irrelevant documents in D can be obtained via explicit relevance feedback. Once we have the seed irrelevant documents D_{I_S}, we can build the seed irrelevance distribution as:

$$p(w|I_S) = \sum_{d \in D_{I_S}} p(w|d) \frac{p(q|d)}{Z_{I_S}} \tag{5.18}$$

where $Z_{I_S} = \sum_{d' \in D_{I_S}} p(q|d')$ and the seed irrelevant documents in D_{I_S} can be selected from the a small percentage (e.g., 10%-30%) of top-ranked irrelevant documents in D.

In implementation, we simulate the explicit feedback by using the relevant judgements (from ground-truth test collections) of a small number of top-ranked irrelevant judgements. In addition to the explicit feedback manner, we introduce the automatic approaches to irrelevant documents/distribution. Recall that the automatic approaches can be regarded as the simulation of the implicit feedback.

Outlier Document Detection

To automatically obtain the seed irrelevant documents, we adopt outlier detection methods by treating outlier documents as irrelevant documents. There are various outlier detection methods in the literature. In this thesis, motivated by the $k-nn$ distance scores (Ramaswamy, Rastogi & Shim 2000, Angiulli & Pizzuti 2002), we adopt the $k-nn$ similarity scores as the indicator for the irrelevant documents. According to the clustering hypothesis (Tombros & van Rijsbergen 2004), the topically-relevant documents tend to cluster together, while the irrelevant documents would be scattered. Therefore, the less the k–nn similarity score is, the more likely a document would be an irrelevant documents. Based on the above ideas, we build the outlier document detection method as follows:

$$Outlier(D,\pi) = \left\{ d \in D | \frac{|\{d' \in D | knn(d') < knn(d)\}|}{|D|} < \pi \right\} \tag{5.19}$$

In the above equation, $Outlier(D,\pi)$ is a set of detected outlier documents which can be used as the seed irrelevant documents. This set is computed by selecting a number (i.e., $|D| \times \pi$) of documents which has the lowest $k-nn$ scores as represented by $knn(d)$, where π is a percentage and $knn(d)$ is the summed Cosine similarity values of the k nearest neighbors of document d. After obtaining the outlier document set $Outlier(D,\pi)$, we can derive the seed irrelevant distribution by:

$$p(w|I_S) = \sum_{d \in Outlier(D,\pi)} p(w|d) \frac{p(q|d)}{Z_{I_S}} \tag{5.20}$$

Here, $Z_{I_S} = \sum_{d' \in Outlier(D,\pi)} p(q|d')$.

Outlier Term Detection

The seed irrelevant term distribution can also be derived though detecting the irrelevant terms in any document. For example, we can regard those terms that are far away from the query terms as the outlier terms. For the document d, the outlier term detection method can be formulated as

$$Outlier(d,\epsilon) = \{w_{i,j} \in d | Distance(w_{i,j}, q) > \epsilon\} \tag{5.21}$$

In the above formulation, $w_{i,j}$ denotes a term token for w_i, and $Distance(w_{i,j}, q)$ measures the distance between the $w_{i,j}$ and the query q. In this thesis, we adopt a simple distance measurement based on the positional difference between $w_{i,j}$ and q. We calculate $Distance(w_i, q)$ by counting the number of term tokens between $w_{i,j}$ and $w_{i,j}$'s nearest query terms in the text. If the number is above a threshold ϵ (e.g., 3), then the corresponding term token is considered as an outlier. After we obtain the set $Outlier(d,\epsilon)$, we can compute the term frequency to obtain an irrelevant term distribution for each document d:

$$p(w_i|Outlier(d,\epsilon)) = \frac{\#(w_i, Outlier(d,\epsilon))}{|Outlier(d,\epsilon)|} \tag{5.22}$$

where $\#(w_i, Outlier(d,\epsilon))$ is the number of tokens of term w_i occurring in the set $Outlier(d,\epsilon)$. Then, we can obtain a seed irrelevant distribution:

$$p(w|I_S) = \sum_{d \in D - Outlier(D,\pi)} p(w|Outlier(d,\epsilon)) \frac{p(q|d)}{Z_{I_S}} \tag{5.23}$$

where $Z_{I_S} = \sum_{d' \in D-Outlier(D,\pi)} p(q|d')$. In the above equation, we extract the outlier term distribution from $D-Outlier(D,\pi)$, which is a document set that contains those remained documents after removing the outlier documents in D.

The reason why we extract irrelevant term distribution from $D - Outlier(D,\pi)$ is because we will construct two-round DSM. In the first round, DSM will automatically remove the term distribution corresponding to $Outlier(D,\pi)$ from the mixture distribution M. In the second round, DSM will then automatically remove the distribution generated by the outlier term detection (based on the remaining documents in $D - Outlier(D,\pi)$). In the experiments, we will also report the results for each round of DSM.

Note that we can use other metrics to calculate the distance between a term w and the query q. For example, we can use domain knowledge or the sematic ontology. In addition, the similarity between w and q can also be used in the detection of outlier terms. We will look into these alternative methods in the future work.

5.2.3 Smoothing with Collection Term Distribution

Recall that the input distributions of DSM are the mixed distribution M and a seed irrelevant distribution I_S. One can smooth each distribution with the collection term distribution C. Specifically, we can obtain

$$p(w|M') = (1-\mu_C)p(w|M) + \mu_C\ p(w|C) \tag{5.24}$$

and

$$p(w|I'_S) = (1-\mu_C)p(w|I_S) + \mu_C\ p(w|C) \tag{5.25}$$

After that, the two resulting distributions M' and I'_S will be the input distributions of DSM. Recall that in Section 5.1.2, the smoothing is a factor that can affect of conditions the theoretical results. In the experiments, we will evaluate the performance of DSM in different smoothing settings.

5.3 Empirical Evaluation

5.3.1 Test Collections

The test collections are the same as those in the previous chapters. Specifically, experiments are conducted on four standard TREC collections, including WSJ (87-92, 173,252 docs), AP (88-89, 164,597 docs) in TREC Disk 1 & 2, ROBUST 2004 (528,155 docs) in TREC Disk 4 & 5, and WT10G (1,692,096 docs). These data sets respectively involve a variety of texts, e.g., newswire articles and Web/blog data. Title queries are used for retrieval. 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.

Lemur (Ogilvie & Callan 2002) 4.7 is used for indexing and retrieval. All collections are stemmed using the Porter stemmer and stop words are removed in the indexing process.

5.3.2 Evaluation Set-up

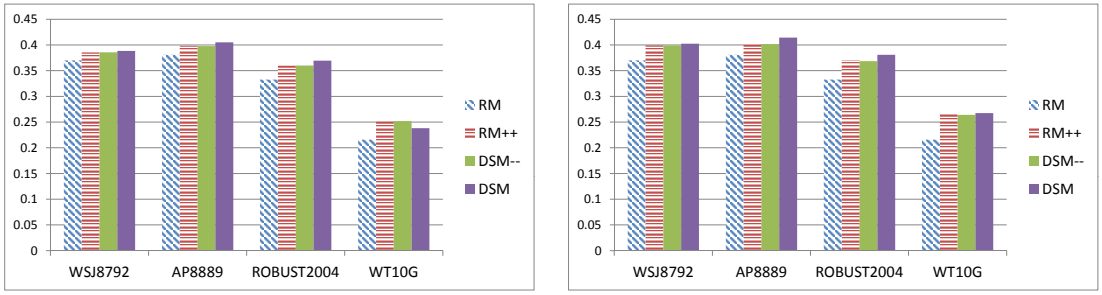
Initially, the top n pseudo feedback documents D are retrieved by the KL-divergence based language model (LM), with Dirichlet prior (Zhai & Lafferty 2001) set as a fixed value 700. RM is selected as the baseline due to the facts that: first, RM is a successful PRF model and consistently outperform the standard LM model by 10%-15% (Lavrenko & Croft 2001); second, it has been demonstrated in (Wang, Fang & Zhai 2008) and in our experiments (Zhang et al. 2009) as well that only based on given irrelevant documents, Rocchio's model does not perform well (see (Zhang et al. 2009) for details).

Next, based on the relevance judgements, a small portion (10% and 20%, denoted as r_n) of irrelevant documents in D are selected as the seed irrelevant documents D_{I_S}. We denote $|D_I|$ as the number of irrelevant feedback documents and choose a number (i.e., $round(|D_I| \times r_n)$) of top-ranked irrelevant documents as the seed irrelevant documents. Note that in Chapter 3, the number of irrelevant documents is $floor(|D_I| \times r_n)$, which is less than $round(|D_I| \times r_n)$. We use $round(|D_I| \times r_n)$ in this chapter is to see more experimental results about RM after removing seed irrelevant documents. We use RM++ to denote RM running on the documents $D - D_{I_S}$, which is a feedback document set with seed irrelevant documents directly removed in D.

Now, based on the mixed distribution M [6] and the irrelevance distribution I_S (see Section 5.2), we run the distribution separation method (DSM) without the refinement step (i.e., step 2 in Algorithm 1), to see whether it can outperform the baseline (RM on D), and whether it can produce a comparable performance with RM++, i.e., RM on $D - D_{I_S}$. We denote DSM without a refinement step as DSM$--$. After that, DSM (with refinement step) is tested to evaluate the effect of the refinement step. In the output distribution by DSM, the top-100 terms with the highest probability values will be the expanded query terms. In all the aforementioned query expansion models, the number of expanded terms is fixed as 100, and 1000 retrieved documents are used for performance evaluation.

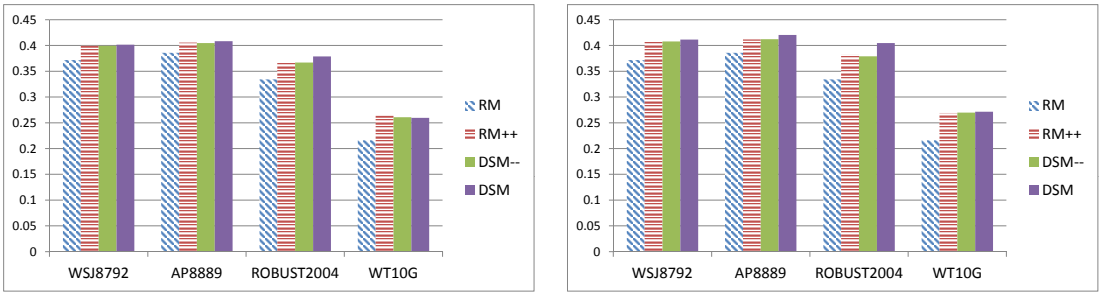
In addition to explicit feedback to obtain the seed irrelevant distribution, we also presented automatic approach in the previous section. In the experiments, we are going to test the outlier document detection (denoted as OutlierD), which can be used to detect seed irrelevant documents, and the outlier term detection (denoted as OutlierDT), which are used to extract the irrelevant terms from the remaining documents in D after outlier documents detected by OutlierD are removed.

[6] For sake of efficiency and feasibility, we just keep the terms whose probabilities are greater than 0.0001 in the mixed distribution, and use the same set of terms in the seed irrelevant distribution.



(a) $r_n = 0.1$ (b) $r_n = 0.2$

Figure 5.3: Performance (MAP) of RM, RM++, DSM−− and DSM, when $n = 30$.



(a) $r_n = 0.1$ (b) $r_n = 0.2$

Figure 5.4: Performance (MAP) of RM, RM++, DSM−− and DSM, when $n = 50$.

Evaluation Metrics

As for the evaluation metric, we use the Mean Average Precision (MAP), which reflects the overall ranking accuracy. In addition, we use the Wilcoxon significance test to examine the statistical significance of the improvements of the DSM models over the baseline. We will also report the performance bias-variance of the proposed methods.

5.3.3 Evaluation on Effectiveness of DSM with Seed Irrelevant Documents Available

Now, we test the retrieval effectiveness of the proposed Distribution Separation Method (DSM) when the seed irrelevant documents can be obtained from the explicit relevance feedback. The evaluation results are summarized in Figure 5.3 when $n = 30$ and Figure 5.4 when $n = 50$, where n is the number of all feedback documents.

In both figures, we can see that RM++ (i.e., RM on $D - D_{I_S}$) can significantly [7] improve RM on all collections. RM++ is a document-level method which corresponds to RM running on the feedback documents with seed relevant documents removed. By contrast, DSM is a distribution-level method whose input are two distributions.

[7] The significant improvement we refer to in this chapter is based on the significance test we conducted.

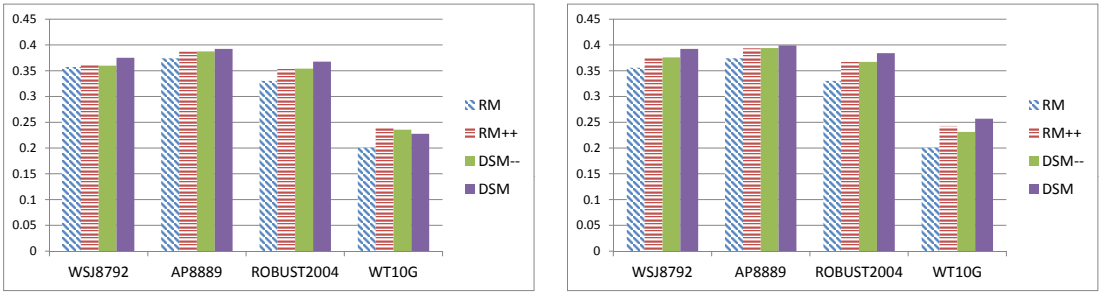
The results in both figures show that DSM−− has very similar performance with RM++. This supports our theoretical justification in Lemma 3 and demonstrates that DSM can derive a less noisy mixture distribution. Recall that DSM−− denotes DSM without the refinement step, meaning that in this case the λ^* computed by DSM often equals to the lower bound λ_L (see the discussions after Algorithm 1). In Lemma 3, it proves that when there is a zero value in $l(R, I_{\overline{S}})$, then the derived distribution $l_L(R, I_{\overline{S}})$ by DSM is the distribution $l(R, I_{\overline{S}})$ which is actually corresponding to the distribution derived by RM++. In our implementation of RM, while the document language model in computing the query likelihood score $p(q|\theta_d)$ is smoothed, the document language model $p(w|\theta_d)$ in RM is unsmoothed [8]. Therefore, $l(R, I_{\overline{S}})$ often has some zero values, which makes DSM−− have very similar performance as RM++. The slight difference can be due to the computation error. For the computation efficiency, in DSM, we only select terms whose probabilities in the mixed distribution M are higher than 0.0001.

Now, we are going to test the DSM with the refinement step (see Algorithm 1). There is only one parameter η to adjust in the DSM. In this chapter, we report the optimal results by selecting $\eta \in [0.4, 1]$ with increment 0.1. The results in Figure 5.3 and Figure 5.4 show that DSM not only significantly outperform RM, but also outperforms RM++ in many cases, for instances, by 1.73% and 2.63% on AP8889 and ROBUST2004, respectively when $n = 30$ and $r_n = 0.1$. Also, an trend is that when r_n increase to 0.2, the larger improvements (e.g., by 2.8% and 2.9% on AP8889 and ROBUST2004) over RM++ can be achieved by DSM. In addition, An exciting trend is that the larger number (when $n = 50$) of feedback documents, the larger improvements over RM++ can be obtained. For example, the improvement over RM++ is 5.91% on ROBUST2004 when $r_n = 0.2$.

Next, we evaluate DSM when its input term distributions are smoothed with the collection term distribution (denoted as C). The smoothing is conducted by a linear interpolation between each input distribution (i.e., M and I_S) and the collection term distribution C. Accordingly, the distribution M corresponding to RM on D and the distribution $l(R, I_{\overline{S}})$ corresponding to RM++ (i.e., RM on $D - D_{I_S}$) are also smoothed in the same way. The smoothing parameter (i.e., the interpolation parameter) is set as 0.5. The results are reported in Figure 5.5 when $n = 30$ and Figure 5.6 when $n = 50$.

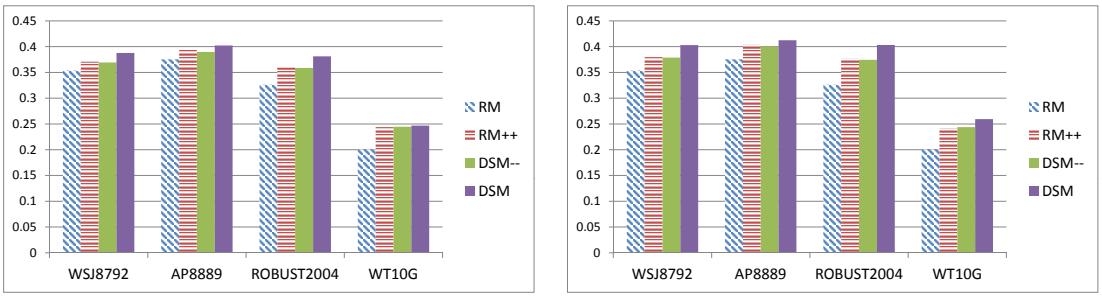
The results in both figures are showing that RM++ can significantly improve RM and RM++ has similar performance with DSM−−. The performance difference between RM++ and DSM−− is small in Figure 5.5 and Figure 5.6. The results about RM++ and DSM−− support Remark 1 and also demonstrate that our distribution separation method is able to derive a less noisy distribution. Remark 1 suggests that when the smoothing is involved and there may not exist zero values but exist small values in $l(R, I_{\overline{S}})$, the output distribution by DSM (without refinement step) should be close to $l(R, I_{\overline{S}})$ corresponding to RM++. This leads to the similar performance between DSM−− and RM++.

[8] This implementation is consistent with the implementation in Lemur 4.7.



(a) $r_n = 0.1$ (b) $r_n = 0.2$

Figure 5.5: Performance (MAP) of RM, RM++, DSM−− and DSM, when $n = 30$, and $\mu_C = 0.5$



(a) $r_n = 0.1$ (b) $r_n = 0.2$

Figure 5.6: Performance (MAP) of RM, RM++, DSM−− and DSM, when $n = 50$, and $\mu_C = 0.5$

At last, we evaluate the DSM with the refinement step (see Algorithm 1 and Section 5.1.3). The results show that DSM not only significantly outperform RM, but also outperforms RM++, for instance, by 6.1%, 2.0%, 6.9% and 7.2% on WSJ8792, AP8889, ROBUST2004 and WT10G, respectively when $n = 50$ and $r_n = 0.2$. The improvements of DSM over RM++ in Figure 5.5 and Figure 5.6 are often larger than the improvements over RM++ plotted previously in Figure 5.3 and Figure 5.4. The reason can be that when the more smoothing is involved, there would be more correlated terms in M and I_S and thus the de-correlation by DSM can be more important. In addition, as shown in Figure 5.5 and Figure 5.6, the larger ratio of irrelevant documents available and the larger number of feedback documents, the larger improvements over RM++ can often be achieved.

5.3.4 Evaluation on Effectiveness of DSM with Automatic Approaches to Seed Irrelevant Distribution

This set of experiments is to test the automatic approaches to the seed irrelevant distribution. The DSM runs on the mixed distribution and this seed irrelevant distribution.

Table 5.3: Evaluation on DSM when $n = 50$ and $\mu_C = 0$

MAP (chg% over RM)	$r_n = 0.1$	$r_n = 0.2$	$r_n = 0.3$
	WSJ8792		
RM (baseline)	0.3719	0.3719	0.3719
RM++ (RM on D-D_{I_S})	0.3996(+7.45%)**	0.4072(+9.49%)**	0.4128(+11.00%)**
DSM−− (no refinement)	0.3993(+7.37%)**	0.4078(+9.65%)**	0.4142(+11.37%)**
DSM	**0.4015**(+7.96%)**	**0.4114**(+10.62%)**	**0.4194**(+12.77%)**
	AP8889		
RM (baseline)	0.3860	0.3860	0.3860
RM++ (RM on D-D_{I_S})	0.4058(+5.13%)*	0.4118(+6.68%)*	0.4179(+8.26%)**
DSM−− (no refinement)	0.4047(+4.84%)*	0.4123(+6.81%)**	0.4169(+8.01%)**
DSM	**0.4083**(+5.78%)*	**0.4204**(+8.91%)*	**0.4258**(+10.31%)**
	ROBUST2004		
RM (baseline)	0.3343	0.3343	0.3343
RM++ (RM on D-D_{I_S})	0.3669(+9.75%)**	0.3820(+14.27%)**	0.3906(+16.84%)**
DSM−− (no refinement)	0.3670(+9.78%)**	0.3790(+13.37%)**	0.3855(+15.32%)**
DSM	**0.3789**(+13.34%)**	**0.4046**(+21.03%)**	**0.4158**(+24.38%)**
	WT10G		
RM (baseline)	0.2163	0.2163	0.2163
RM++ (RM on D-D_{I_S})	**0.2639**(+22.01%)**	0.2693(+24.50%)**	0.2813(+30.05%)**
DSM−− (no refinement)	0.2608(+20.57%)**	0.2699(+24.78%)**	0.2793(+29.13%)**
DSM	0.2596(+20.02%)**	**0.2714**(+25.47%)**	**0.2835**(+31.07%)**

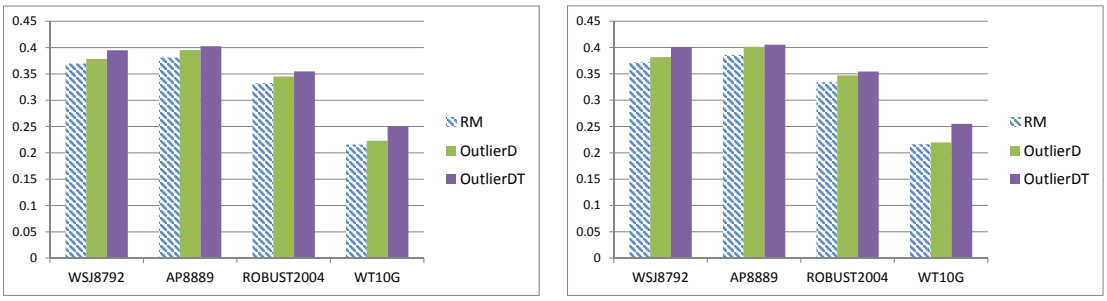
**Statistically significant improvement at level 0.01 according to Wilcoxon signed rank test.
*Statistically significant improvement at level 0.05 according to Wilcoxon signed rank test.

Table 5.4: Evaluation on DSM when $n = 50$ and $\mu_C = 0.5$

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++ (RM on D-D_{I_S})	0.3710(+4.86%)**	0.3798(+7.35%)**	0.3872(+9.44%)**
DSM−− (no refinement)	0.3693(+4.38%)**	0.3787(+7.04%)**	0.3776(+6.73%)**
DSM	**0.3878**(+9.61%)**	**0.4030**(+13.91%)**	**0.4056**(+14.64%)**
	AP8889		
RM (baseline)	0.3755	0.3755	0.3755
RM++ (RM on D-D_{I_S})	0.3939(+4.90%)*	0.4042(+7.64%)**	0.4109(+9.43%)**
DSM−− (no refinement)	**0.3898**(+3.81%)*	**0.4004**(+6.63%)**	**0.4050**(+7.86%)**
DSM	0.4024(+7.16%)*	0.4125(+9.85%)*	0.4218(+12.33%)**
	ROBUST2004		
RM (baseline)	0.3262	0.3262	0.3262
RM++ (RM on D-D_{I_S})	0.3599(+10.33%)**	0.3772(+15.63%)**	0.3853(+18.12%)**
DSM−− (no refinement)	0.3586(+9.93%)**	0.3745(+14.81%)**	0.3797(+16.40%)**
DSM	**0.3813**(+16.89%)**	**0.4033** (+23.64%)**	**0.4106**(+25.87%)**
	WT10G		
RM (baseline)	0.2004	0.2004	0.2004
RM++ (RM on D-D_{I_S})	0.2439(+21.71%)**	0.2419(+20.71%)**	0.2631(+31.29%)**
DSM−− (no refinement)	0.2447(+22.11%)**	0.2438(+21.66%)**	0.2609(+30.19%)**
DSM	**0.2468**(+23.15%)**	**0.2594**(+29.44%)**	**0.2724**(+35.93%)**

**Statistically significant improvement at level 0.01 according to Wilcoxon signed rank test.
*Statistically significant improvement at level 0.05 according to Wilcoxon signed rank test.

Recall that one method is the outlier document detection (denoted as OutlierD) to detect the seed irrelevant documents. The results on OutlierD in Figure 5.7 corresponds to the



(a) $n = 30$ (b) $n = 50$

Figure 5.7: Performance (MAP) of RM and DSM using OutlierD and OutlierDT

performance of DSM running on the seed irrelevant distribution generated from the outlier documents. This is the first-round DSM and the output distribution is used as the mixed distribution for the next-round DSM. In the next round, the outlier term detection (denoted as OutlierDT) is used to extract the irrelevant terms from the remaining documents in D after outlier documents detected by OutlierD are removed. After that, the DSM will remove this irrelevant term distribution from the mixed distribution obtained from the first-round separation.

The results about OutlierD and OutlierDT are summarized in Figure 5.7. The baseline is RM when the corresponding mixed distribution M is not smoothed with the collection model. The reason is that RM (without smoothing) in Figure 5.3 and Figure 5.4 usually outperform RM (with smoothing) in Figure 5.5 and Figure 5.6, respectively. We are interested to see if DSM using OutlierD and OutlierDT can outperform this better baseline.

Figure 5.7 shows that OutlierD can outperform the baseline RM. OutlierD can improve RM by 2.26%, 3.7%, 3.75% and 3.28% on WSJ8792, AP8889, ROBUST2004 and WT10G, respectively, when $n = 30$, and by 2.63%, 3.94%, 3.86% and 1.62% on the corresponding collections when $n = 50$. OutlierDT can significantly outperform RM by 6.7%, 5.64% and 6.67% on WSJ8792, AP8889, ROBUST2004, respectively when $n = 30$, and by 7.9%, 5% and 5.98% on these three collections when $n = 50$. An exciting result is that OutlierDT can even outperform RM++ by 2.38% and 1.05% on WSJ8792 and AP8889, respectively when $n = 30$ and $r_n = 0.1$ (see Figure 5.7, and is comparable to RM++ on other cases. We would mention that DSM using automatic outlier detection can have comparable performance with DSM using explicit feedback (when $r_n = 0.1$). The above observations show that the outlier methods can be useful in automatically deriving the seed irrelevant distribution.

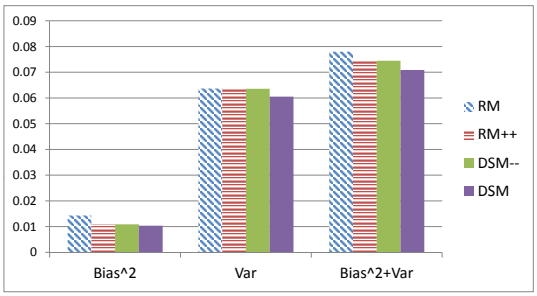
5.3.5 Evaluation on Performance Bias-Variance

Now, we will report the evaluation results about the performance bias-variance of aforementioned methods. The true query model is the same as the one used in Chapter 3. The

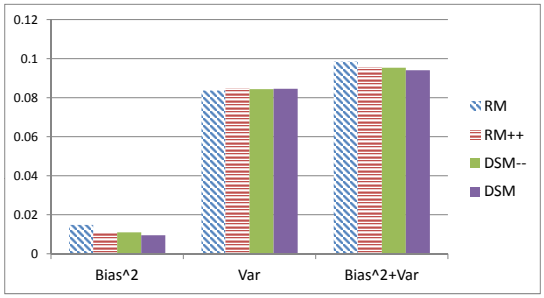
Table 5.5: Evaluation on DSM using Outlier Detection when $\mu_C = 0$ and $n = 50$

MAP (chg%)	WSJ8792	AP8889	ROBUST2004	WT10G
RM (baseline)	0.3719	0.3860	0.3343	0.2163
OutlierD	0.3817(+2.64%)	0.4012(+3.94%)*	0.3472(+3.86%)*	0.2198(+1.62%)
OutlierDT	0.4011(+7.85%)**	0.4053(+5.00%)*	0.3543(+5.98%)*	0.2550(+17.89%)**
RM++ ($r_n = 0.1$)	0.3996(+7.45%)**	0.4058(+5.13%)*	0.3669(+9.75%)**	**0.2639**(+22.01%)**
DSM ($r_n = 0.1$)	**0.4015**(+7.96%)**	**0.4083**(+5.78%)*	**0.3789**(+13.34%)**	0.2596(+20.02%)**

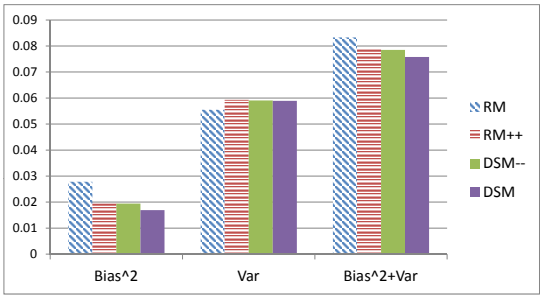
**Statistically significant improvement at level 0.01 according to Wilcoxon signed rank test.
*Statistically significant improvement at level 0.05 according to Wilcoxon signed rank test.



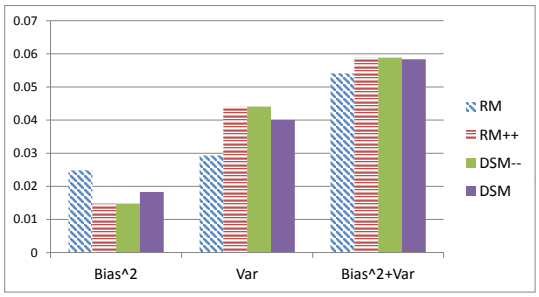
(a) WSJ8792



(b) AP8889



(c) ROBUST2004



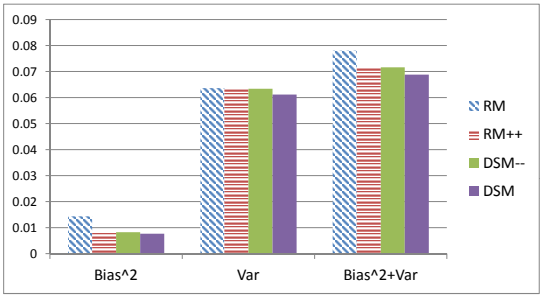
(d) WT10G

Figure 5.8: Performance bias-variance of RM, RM++, DSM−− and DSM, when $r_n = 0.1$

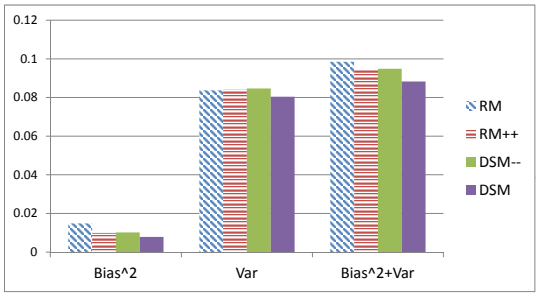
number of feedback documents is 30 and the mixed distribution by RM is not smoothed with the collection model, which are the same settings as those in Chapter 3. We report the results of RM++ when r_n is 0.1 and 0.2. The results are similar when $r_n = 0.3$.

Let us first see the results when a small portion of irrelevant documents are available. When $r_n = 0.1$, the results are plotted in Figure 5.8. We can observe that RM++ can always reduce the performance bias, however, it can not reduce the performance variance and even increase the performance variance on ROBUST2004 and WT10G. Note that the larger the r_n is, the more likely that the performance variance can be reduced by RM++, according to the results reported in Chapter 3. In this chapter, we are particularly interested in the case when r_n is small, since in practice it is more feasible to obtain a small number of irrelevant documents.

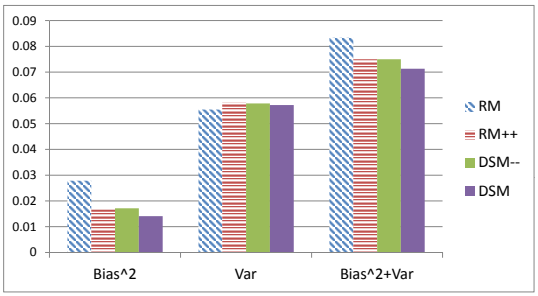
RM++ and DSM−− have very similar results on both performance bias and variance, which supports our theoretical analysis in Section 5.1.2. Both RM++ and DSM−− can



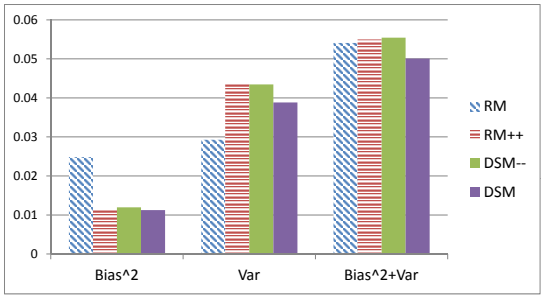
(a) WSJ8792



(b) AP8889

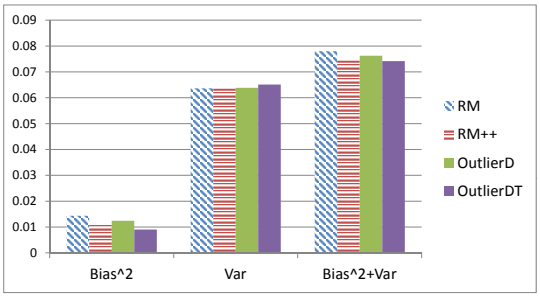


(c) ROBUST2004

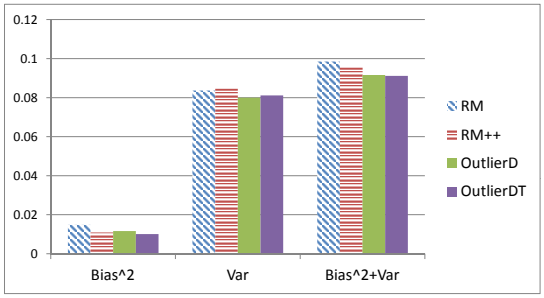


(d) WT10G

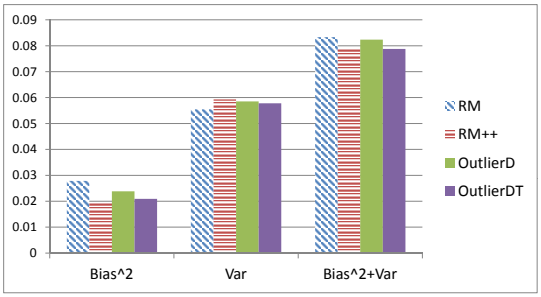
Figure 5.9: Performance bias-variance of RM, RM++, DSM−− and DSM, when $r_n = 0.2$



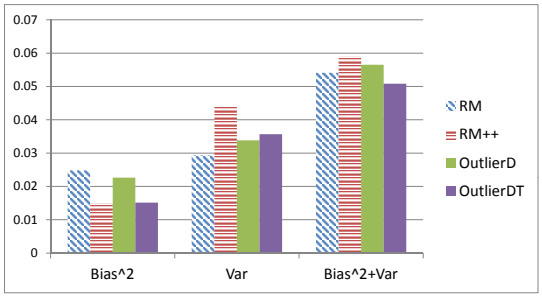
(a) WSJ8792



(b) AP8889



(c) ROBUST2004



(d) WT10G

Figure 5.10: Performance bias-variance of RM, RM++ ($r_n = 0.1$) and DSM using OutlierD and OutlierDT

reduce $Bias^2 + Var$, indicating that they are more robust than RM. With regard to DSM, it can reduce the performance variance on WSJ8792 and WT10G over RM, while RM++ and DSM−− fail to do so. DSM can further reduce the $Bias^2 + Var$ and has the smallest $Bias^2 + Var$ on WSJ8792, AP8889, ROBUST2004, indicating DSM is the most robust method on these collections. When r_n increases to 0.2 (see Figure 5.9), the performance bias and variance can be further reduced by DSM. On each collection, DSM has the smallest $Bias^2 + Var$ and can be considered as the most robust method.

We also evaluate DSM using the automatic approaches, i.e., OutlierD and OutlierDT. The evaluation results are illustrated in Figure 5.10. It shows that OutlierD can reduce the performance bias of RM. The performance bias of OutlierD is larger than RM++, while the performance variance of OutlierD is better (i.e., smaller) than that of RM++. In terms of $Bias^2 + Var$, OutlierD can outperform RM++ on AP8889 and WT10G, since it has smaller $Bias^2 + Var$.

As for OutlierDT, its performance bias is smaller than that of RM++ on WSJ8792 and AP8889 and the results are similar between them on ROBUST2004 and WT10G. Its performance variance is also smaller than that of RM++ on AP8889, ROBUST2004 and WT10G. The results about $Bias^2 + Var$ show that DSM using OutlierDT performs the best on AP8889 and WT10G and can be regarded as the most robust method on these two collections. On other collections, DSM using OutlierDT also have comparable $Bias^2 + Var$ with RM++. In addition, DSM using automatic outlier detection can have comparable performance with DSM using explicit feedback (when $r_n = 0.1$). The above results show that although OutlierDT is simple, it can help DSM automatically obtain the seed irrelevant distribution and improve both the retrieval effectiveness and retrieval stability of RM.

5.4 Summary

In this chapter, we focus on the second factor (i.e., removing irrelevant documents) that can affect the bias-variance of the query model estimation based on relevance feedback. We proposed a novel distribution-level method, i.e., DSM, which can derive an optimal approximation of the true relevance distribution from a mixture distribution (corresponding to pseudo relevance feedback documents), based on seed irrelevant distribution. The proposed model is neat due to the theoretical justification and guarantee of its solutions. Practically, DSM consistently and significantly outperforms RM in terms of both retrieval effectiveness and stability as indicated by the performance bias and variance, respectively. Moreover, DSM can also outperform RM when the seed irrelevant documents have been directly removed in the feedback documents. Our evaluation based on performance bias-variance shows that DSM are often the most robust method among the tested method in terms of the combined effect of the retrieval effectiveness and robustness.

In addition to the explicit feedback to obtain a small number of irrelevant documents, we also develop automatic approaches to obtaining seed irrelevant distribution. We present the outlier document detection method (denoted as OutlierD) to detect seed irrelevant documents. We also propose an outlier term detection (denoted as OutlierDT) to extract the irrelevant terms from feedback documents. Experiments have shown that DSM using OutlierD can improve the retrieval effectiveness of RM, and the improvements are significant when using OutlierDT to generate irrelevant distribution for DSM. Moreover, DSM using OutlierDT can be more robust than RM (with seed irrelevant documents removed), on some collections, e.g., AP8889 and WT10G. These observations suggest that the automatic outlier detection can make DSM be more practical and automatic. DSM using automatic outlier detection can have comparable performance with DSM using explicit feedback (when $r_n = 0.1$). We will also keep investigating this line of research in the future work.

It is worth mentioning that our method is based on distributions rather than documents, and is thus more general. It can treat the distribution generation method (e.g., RM) as a black box and can separate the needed distribution from the original mixture distribution, by removing another irrelevant/noise distribution automatically. In many cases, for example, we may have discarded old relevant or irrelevant documents after a number of search iterations. Nevertheless, we may still be able to keep updating relevant or irrelevant term distribution incrementally, as well as keep tracking other items or features, such as query modifications, from which a term probability distribution can be formed. Generally speaking, the DSM method is expected to be applied to other IR tasks or other fields, since there is no such restrictions in DSM that the distribution should only be query term distribution.

Chapter 6

Further Explorations

In addition to our main contributions described in previous chapters, we start to explore further directions on the application of the bias-variance framework to personalization task. We also try to explore the problem of the estimation of relevance (including document relevance and query relevance) via the Quantum theory perspective.

6.1 Application of Bias-Variance in Personalization

We first try to apply the bias-variance analysis in the personalization task. We can consider different users as different queries. To evaluate a personalization technique, there are two criteria, i.e., the mean user satisfaction over all users (corresponding to mean retrieval performance over all queries) and the variance of user satisfaction across all users (corresponding to the variance of retrieval performance across all queries).

The potential of personalization corresponds to the gap between how well search engines can satisfy users by providing a single rank list, and how well search engines can satisfy every user by providing personalized rank lists (Teevan, Dumais & Horvitz 2010).

We use bias and variance decomposed from the mean squared error of the actual user satisfaction, to analyze the potential of personalization, and discuss the bias-variance tradeoff in designing personalization techniques. Initial analysis could be done by using average precision to simulate the user satisfaction.

We first look at the mean squared error of actual user satisfaction:

$$\mathbb{E}_u(S_u - S)^2 = \sum_{u \in \mathcal{U}} (S_u - S)^2 p(u) \tag{6.1}$$

where S_u can be actual satisfaction value [1] of user u, and S can be the ideal or maximum satisfaction value (a constant) of any user after viewing the ideally personalized rank lists.

[1] The measurement can be average precision (AP) given a single query, or mean average precision (MAP) given a group of queries. Or, it can be other IR performance metrics.

To see the potential of personalization, one can think that S_u represents the user satisfaction after user u viewing a single rank list. Then, the quantity represented in Eq. 6.2 represents the room for improvement over the single rank list by the personalization techniques. Such a room corresponds to the maximum potential of personalization.

The above mean squared error can be decomposed by:

$$\begin{aligned} \mathbb{E}_u(S_u - S)^2 &= \mathbb{E}_u(S_u - \mathbb{E}_u S_u)^2 + (\mathbb{E}_u S_u - S)^2 \\ &= Var(S_u) + Bias^2(S_u) \end{aligned} \tag{6.2}$$

where variance $Var(S_u)$ represents the variance of the user satisfaction across users, and $Bias(S_u)$ represents the gap between the mean user satisfaction value $\mathbb{E}_u S_u$ and the ideal satisfaction value S.

Now, let us look at the actual user satisfaction value (denoted as S_u^p) of personalization techniques in practice. The target of S_u^p is also S. In fact, the actual user satisfaction S_u in Eq. 6.2 can also represent S_u^p. The mean square error of S_u^p can measure how well the corresponding personalization technique is. Now, we use S_u^s denote the user satisfaction *w.r.t.* the single rank list. Both S_u^p and S_u^s are the actual user satisfaction in Eq. 6.2.

The personalization's potential we refer to is not only concerned about user satisfaction comparison between the single rank list and personalized rank list, but also related to issues in designing and optimizing the personalization techniques. Specifically, we aim to use the mean squared error in Eq. 6.2 and its bias-variance decomposition to answer:

1) When will the personalization be helpful? In other words, when S_u^p is better than S_u^s?

2) When there will be a bias-variance tradeoff of the user satisfaction using personalization techniques?

3) How to design personalization techniques to balance such a tradeoff to optimize the user satisfaction.

We have presented the basic formulation of the bias-variance in the personalization tasks. Three research questions are also provided. In the future work, we will mainly deal with these research questions.

6.2 The Analogy of Photon Polarization in Relevance Feedback

The bias-variance tradeoff idea was initially somehow inspired by one quantum theory – the Heisenberg's Uncertainty Principle, which states a tradeoff between the momentum and the position of an electron or photon and one can not precisely measure them simultaneously. We were initially seeking an analogy of such a tradeoff in IR. Bias-variance tradeoff can be considered as a kind of the uncertainty principle (Grenander 1952, Ge-

man et al. 1992). The tradeoff between the retrieval effectiveness and retrieval stability can be considered as a tradeoff in performance evaluation. In addition, the relevance estimation also has a nature of uncertainty (Mizzaro 1996). Specifically, even though the statistic information (e.g., TF, IDF, document length) has been calculated precisely, there is still a high level of uncertainty associated to the relevance of document. For instance, the relevance estimation can be varied for different users, in different contexts, and during different times (Zhang, Beresi, Song & Hou 2010, Zhang, Song, Hou, Wang & Bruza 2010).

We believe that in order to investigate Heisenberg's Uncertainty Principle in IR in depth, we should build analogies of the quantum phenomenon in IR problems. We propose an analogy of photon polarization (a key Quantum experiments) in IR, particularly for relevance feedback.

The photon polarization experiment (Rieffel & Polak 2000) involves the probability measurement of photons that can pass through a polarization filter. We can view documents as photons, and the retrieval process as measuring the probability of each document that can pass through the query's retrieval filter (as polarization filter). Then, the measured probability can be regarded as the estimated probability of relevance of each document. This QM experiment usually inserts an additional filter between the original filter and the photon receiver (e.g. a screen). Similarly, in query expansion, the expanded query is constructed for the second-round retrieval.

In QM, the probability that a photon can pass through an additional filter is the combined effect of probability measurement on both filters (i.e., the original and the additional ones). This inspires us, in IR, to fuse (i.e. combine) the retrieved results from the original query and the expanded one. Indeed, such fusion-based method has been shown as an effective approach to tackling the query-drift problem (Zighelnic & Kurland 2008). Recall that In the literature review, we built the connections between the combination methods and the fusion approaches. In Chapter 3, we also show that the combination method could be an effective way to reduce both performance bias and variance.

Photon polarization provides a new perspective and a novel mathematical framework to look at the problem by considering the representation of the additional filter under the same basis as the original filter. This means that the expanded query can be implicitly observed with respect to the original one. Based on the above idea, we can formulate the query expansion process under the QM framework.

6.2.1 Photon Polarization

We first briefly introduce the idea of photon polarization (Rieffel & Polak 2000). A photon's state can be modeled by a unit vector $\varphi = a\,|{\rightarrow}\rangle + b\,|{\uparrow}\rangle$, which is a linear combination of two orthogonal basis vectors $|{\rightarrow}\rangle$ (horizontal polarization) and $|{\uparrow}\rangle$ (vertical polarization). The amplitudes a and b are complex numbers such that $|a|^2 + |b|^2 = 1$. Suppose

the original filter is a horizontal polarization filter. Each photon will be measured by the basis $|\rightarrow\rangle$ and the probability is $|a|^2$, i.e., the squared norm of corresponding amplitude a in the horizontal direction. After the measurement, the photon's state will collapse to the original basis vector $|\rightarrow\rangle$. If we now insert an additional filter (e.g. with direction $\nearrow$ of 45-degree angle), then the new basis vectors become $|\nearrow\rangle$ and its orthogonal basis $|\nwarrow\rangle$. Now, to measure the probability of the collapsed photon under this new basis, the relation between the new and the original basis should be considered (Rieffel & Polak 2000). Next, we will describe it in detail in the light of our proposed approach.

6.2.2 QM-Inspired Fusion Approach

In the first-round retrieval, under the QM formulation, a document d's state can be formulated as:

$$|\varphi_d\rangle = a_d\,|q\rangle + b_d\,|\neg q\rangle \tag{6.3}$$

where q is the original query, $|q\rangle$ denotes the basis vector for relevance, $|\neg q\rangle$ denotes the basis for irrelevance which is orthogonal to $|q\rangle$, and $|a_d|^2 + |b_d|^2 = 1$. $|a_d|^2$ can denote the estimated relevance probability of the document d with respect to q. If we do not consider the state collapse after the first-round retrieval, d's state with respect to the expanded query q^e can be represented as

$$|\varphi_d^e\rangle = a_d^e\,|q^e\rangle + b_d^e\,|\neg q^e\rangle \tag{6.4}$$

where $|a_d^e|^2 + |b_d^e|^2 = 1$ and $|a_d^e|^2$ denotes the estimated relevance probability of document d with respect to q^e.

To prevent query-drift, the existing fusion models in (Zighelnic & Kurland 2008) directly combines two probabilities $|a_d|^2$ and $|a_d^e|^2$. This direct combination ignores the theoretical fact that two probabilities are under different basis, i.e. $|q\rangle$ and $|q^e\rangle$, respectively.

In this thesis, we propose to fuse $|a_d|^2$ and $|a_d^e|^2$ on the same basis. First, to connect different basis $|q\rangle$ and $|q^e\rangle$, let $|q^e\rangle = a_{q^e}\,|q\rangle + b_{q^e}\,|\neg q\rangle$, where $|a_{q^e}|^2 + |b_{q^e}|^2 = 1$. Assuming the amplitudes in Eq. 6.3 and Eq. 6.4 have been estimated, a_{q^e} can be estimated by solving the equation $|\varphi_d\rangle = |\varphi_d^e\rangle$ (see Eq. 6.3 and 6.4). If we consider the collapse of $|\varphi_d\rangle$ to $|q\rangle$ after the first-round retrieval, the equation $|q\rangle = a_d^f\,|q^e\rangle + b_d^f\,|\neg q^e\rangle$ needs to be solved too, using the estimation of a_{q^e}. The a_d^f here denotes the fused amplitude on the basis $|q^e\rangle$. The process of solving the above equations is omitted due to the space limit. The solution is that $a_d^f = a_d a_d^e + b_d b_d^e$. The amplitudes b_d and b_d^e correspond to the irrelevance basis and leads to unstable performance in our experiments. Therefore, we can drop the term $b_d b_d^e$ in a_d^f. Nevertheless, we will investigate its effect in more detail in the future. Then, we have

$$a_d^f = a_d a_d^e \tag{6.5}$$

Model	**Fused Score for each** d
combMNZ	$(\delta_q(d) + \delta_{q^e}(d)) \cdot (\delta_q(d)\|a_d\|^2 + \delta_{q^e}(d)\|a_d^e\|^2)$
interpolation	$\lambda\delta_q(d)\|a_d\|^2 + (1-\lambda)\delta_{q^e}(d)\|a_d^e\|^2 \quad (0 \leq \lambda \leq 1)$
QFM1	$(\delta_q(d)\|a_d\|^2) \cdot (\delta_{q^e}(d)\|a_d^e\|^2)$
QFM2	$(\delta_q(d)\|a_d\|^2) \cdot (\delta_{q^e}(d)\|a_d^e\|^2)^{1/\eta} \qquad (\eta > 0)$

Table 6.1: Summary of Fusion Models

Let $|a_d^f|^2 = |a_d|^2 \cdot |a_d^e|^2$ denote the fused relevance probability, which considers both $|a_d|^2$ (see Eq. 6.3) and $|a_d^e|^2$ (see Eq. 6.4), on the same basis $|q^e\rangle$. For each document d, $|a_d|^2$ and $|a_d^e|^2$ can be estimated as the normalized relevance scores by a retrieval model for the original query q and the expanded query q^e, respectively. Before describing the QM-inspired fusion model, it is also necessary (Zighelnic & Kurland 2008) to define two functions $\delta_q(d)$ and $\delta_{q^e}(d)$, the value of which is 1 if d is in the result list of the corresponding query, and 0 otherwise.

Then, based on Eq. 6.5, we propose two QM-inspired Fusion Models (namely QFM1 and QMF2), as formulated in Tab. 6.1. Two existing fusion models in (Zighelnic & Kurland 2008), namely combMNZ and interpolation, are re-formulated in Tab. 6.1 for comparison. The combMNZ and interpolation are additive (i.e. adding up two scores $|a_d|^2$ and $|a_d^e|^2$), while the QM-based models are multiplicative. combMNZ and QFM1 parameter-free. In QMF2, the smaller parameter η can make scores of different documents retrieved for q^e more separated from each other, leading to more distinctive scores. In the interpolation model, the smaller λ, the more biased the fused score is to the second-round score (i.e. $|a_d^e|^2$) for the expanded query q^e. For the initial experiments about the proposed QM-inspired fusion method, please refer to (Zhang, Song, Zhao & Hou 2011).

6.3 Summary

In this chapter, we propose a basic bias-variance formulation in terms of the user satisfaction in the personalization task and provide three research questions for the further explorations. On the other hand, we propose to look at the relevance estimation (including document relevance and query relevance) from a novel theoretical perspective inspired by the photon polarization in QM, and accordingly we have developed a novel fusion approach to query expansion. In the future, we will investigate the above research directions (i.e., bias-variance in personalization, analogy of photo polarization in relevance feedback) in depth and aim to integrate them into a unified uncertainty principle of IR.

Chapter 7

Conclusion and Future Work

In this thesis, we aim to tackle the challenges in approximating the true relevance model, which represents the underlying information need. Novel frameworks, theories and methods have been developed to improve the retrieval effectiveness and/or stability of query model estimation in the context of relevance feedback. This chapter summarizes our main contributions and points out the future directions.

7.1 Contributions

We have presented a bias-variance framework, which provides a series of formulations to analyze the retrieval performance and the estimation quality of an estimation query model, with respect to the true query model. The true query model approximates the true relevance model and gives the upper-bound performance among all the query model we investigated in this thesis . It has been demonstrated that based on the bias-variance tradeoff, the retrieval effectiveness-stability tradeoff as well as the estimation effectiveness-stability tradeoff, can be studied in a principled way. It turns out that some factors, e.g., query model complexity, query model combination, document weight smoothness and irrelevant documents removal, can affect the bias and variance of the query model estimation. We then investigate the latter two factors about the document weight and irrelevant documents by exploring new theoretical aspects and proposing new generic methods. Specifically, we proposed to study the rank-independent risk of document relevance estimation, associated to the document weight smoothing. We also went beyond the document-level of removing irrelevant documents, by proposing a distribution separation method to separate the true relevance distribution from the mixture distribution, by removing the irrelevant distribution. In the following, we will give more detailed descriptions of the above main contributions.

7.1.1 The Bias-Variance Analysis Framework

In Chapter 3, we have proposed a bias-variance analysis framework, which provides a new theoretical perspective to address the challenges about the retrieval effectiveness-stability tradeoff in the query model estimation. Specifically, in Section 3.1.2, we proposed the bias-variance formulation about the retrieval performance. The performance bias and variance are related to the retrieval effectiveness and stability, respectively. Therefore, the retrieval effectiveness-stability tradeoff can be naturally studied based on the general principles of bias-variance tradeoff.

In order to directly investigate the estimation quality of an estimated query model with respect to the true query model, we also formulated the estimation bias and variance of an estimated query model. The estimation bias represents the expected estimation error over all queries, while the estimation variance is the variance of estimation error across different individual queries. The sum of squared bias and variance can yield the total (squared) estimation error which can directly indicate the overall estimation quality. It turns out that the estimation bias and variance directly are directly related to how closely an estimated query model can approach the true one.

We systematically analyzed the bias and variance of several estimated query models based on four factors, i.e., query model complexity, query model combination, document weight smoothness and irrelevant documents removal. We also addressed different bias-variance trends on different kinds of bias-variance and different metrics. Based on our analysis, we then proposed a set of hypotheses with respect to those factors on bias-variance tradeoff and on reducing both bias and variance simultaneously. A series of experiments based on TREC datasets have been conducted and generally supported the hypotheses. Next, we summarize the evaluation results of the bias-variance trends for different query models.

Evaluation results showed that the performance bias of the original query model was higher than that of the expanded query model, while the performance variance of the original one was lower than the expanded one. This indicated that the expanded query model was more effective, but less stable, than the original one. We explained this phenomenon by the effect of the *query model complexity* on the bias and variance. According to the general intuitions of bias-variance tradeoff, the more complex method can have lower bias but higher variance. The expanded query model is more complex than the original query model, due to the fact that it has more parameters (e.g., the number of expanded query terms) and has more assumptions (e.g., top-ranked documents can be relevant). On the other hand, the tradeoff was not obvious for the KL-divergence-based estimation bias and variance, due to the sparsity of the original query model and the scale problem of KL-divergence. We did observe the tradeoff when we use other metrics (e.g., JS-divergence and Cosine similarity) to formulation the estimation bias and variance.

The *combination* between the original query model and the expanded query model can

reduce the performance bias and variance simultaneously, subject to a proper combination coefficient. Our analysis suggested that a small coefficient (e.g., 0.1) could adjust the probabilities of original terms in the expanded query model while preventing the combined query model being dominated by the original query terms [1]. The evaluation results also supported such analysis. With respect to the estimation bias-variance, we also observed that a small coefficient can reduce the bias and variance simultaneously when we use the KL-divergence as the metric. On the other hand, the bias-variance tradeoff still occurred obviously on the whole range (i.e., [0,1]) of combination coefficients. In this range, the combined query model is moving from the expanded query model to the original query model. Recall that the tradeoff occurred obviously between the original query model and the expanded one. Therefore, the bias-variance tradeoff would still occur obviously for the combined query model with its coefficient value in the range [0,1].

In addition to the query model complexity and the query model combination, the *document weight smoothness* is another important factor which affects the bias and variance in query model estimation. It can be observed that document weight is one difference between the expanded query model by RM (in Eq. 3.41) and the true query model (in Eq. 3.44). In the true query model, the document weights are uniform for all the relevant documents, while in RM, the document weights are the normalized query likelihood scores which are not smooth. Smoothing document weights can improve the smoothness of document weights of those relevant documents in the feedback document set. Evaluation results showed that a moderate smoothing reduced the performance bias, while increased the performance variance of expanded query model by RM. The increased performance variance is due to the variance (or fluctuation) of the quality of the feedback documents across all queries. For example, some queries may have many relevant feedback documents, but others only have a few. Experimental results showed that the larger performance variance of the initial ranking by the original query model, the larger performance variance can caused by the smoothing method. On the other hand, it is more likely that the document weight smoothing can reduce the estimation bias and variance simultaneously, as indicated by our analysis and also supported by the evaluation. It is because that the estimation bias and variance are directly related to the true query model, which are generated from relevant documents. Therefore, to smooth the document weights can improve the smoothness of relevant document, thereby improving the estimation quality with respect to the true query model.

To remove the *irrelevant documents* in the feedback document set is certainly a very important factor to make any estimated query model approach to the true query model. It can reduce the performance bias and variance simultaneously, as well as reduce the estimation bias and variance simultaneously. It is because that one often need to obtain

[1] Since the original terms are sparse in the original query model, a large coefficient can easily make the combined query model be dominated by the original query terms.

more data (e.g., relevance judgements) to remove the irrelevant documents. The larger the available data we have, the more likely that the bias and variance can be reduced simultaneously. This is also consistent with the principles of bias-variance tradeoff. Once all the irrelevant documents are removed in the feedback document set, smoothing the document weights of the remaining relevant documents can further reduce the bias and variance simultaneously and approximate the true query model better. This has been observed in our experiments.

In addition to the above results about the bias and variance, we also consider the sum of bias and variance as a metric to indicate the robustness of the retrieval performance and the estimation quality. Recall that we argued that the robustness of a query model should take into account both the effectiveness and stability. Therefore, the sum of bias and variance can naturally serve as a criteria for the robustness. The bias-variance decomposition can be considered as an effectiveness-stability decomposition of the robustness in the query model estimation. Evaluation results showed that the aforementioned factors, i.e., the combination method, the document weight smoothing and the irrelevant documents removal can contribute in varying degrees to the robustness of the query expansion, where the irrelevant documents removal is the most important factor.

7.1.2 Document Weight Smoothing and Allocation Methods

In addition to the bias-variance analysis, we also proposed to investigate the rank-independent risk associated to the document weights, which are often computed by the normalized document relevance scores . In the literature, the estimation quality of document relevance is closely dependent on the ranking performance of the corresponding retrieval method. This is reflected by the facts that if a good ranking performance has been obtained, one would often neglect whether the document relevance estimation is precise or not.

However, we proposed a research question: can the optimal or even ideal ranking always guarantee that the estimation is precise. It turned out that the answer is no and part of the estimation risk should be independent of the rank. It also imposes practical risks in relevance feedback, where different estimates of relevance in the first-round retrieval will make a difference even when two corresponding ranks are identical.

We clarified that the *rank-dependent risk* refers to the relevance estimation risk that can influence the rank, while the *rank-independent risk* does not. In practice the ideal rank is usually unavailable, both types of risks may exist in the estimated relevance probabilities. Therefore, we first singled out the effect of the rank-independent risk associated to different estimated relevance estimators when the resultant ranks are identical.

Specifically, we showed that even though two language modeling approaches were rank-equivalent, their estimated relevance distributions were different. Motivated by such difference (see Eq. 4.8), a risk management method was proposed to manage the rank-independent risk. An entropy-bias explanation is provided to support the rationality of

the proposed risk management method. This risk management method is actually the document weight smoothing method used in Chapter 3. For a given retrieval model, the rank-independent risk management method (i.e., the document weight smoothing) can be regarded as the micro-level adjustment, as opposed to the re-ranking approaches (tackling the rank-dependent risk). Evaluation results on several TREC collections demonstrated the effectiveness of the proposed method against the state-of-the-art re-ranking methods that are used to tackle the rank-dependent risk in pseudo-relevance feedback.

Based on the document weight smoothing method, we also propose two weight allocation methods, which can tackle the rank-dependent risk by re-ranking the feedback documents. We constructed a systematic bias-variance evaluation, which showed that the weight allocation methods are able to further improve the weight smoothing method, evidenced by the result that the weight allocation methods can further reduce the performance bias-variance and estimation bias-variance over the weight smoothing method. The nonlinear weight allocation methods are more likely to achieve the lowest bias and/or variance values.

7.1.3 Distribution Separation Method (DSM) and Outlier Detection

Removing irrelevant documents has been demonstrated as a more important factor than smoothing document weights, to reduce the bias and variance simultaneously (see Chapter 3). In Chapter 5, we then went beyond the document level, i.e., directly removing irrelevant documents, by proposing a distribution-level method, namely Distribution Separation Method (DSM). DSM can derive an optimal approximation of the true relevance distribution from a mixture distribution (corresponding to pseudo relevance feedback documents), by removing a seed irrelevant distribution.

DSM is more general than the document-level removing. First, it can treat the distribution generation method (e.g., RM) as a black box and automatically obtain the desirable distribution from the mixture distribution, by automatically identify the combination coefficient of the desirable distribution. Second, DSM can be applicable in many scenarios where each document is not available. For example, we have already discarded old relevant or irrelevant documents, but still keep updating the relevant or irrelevant query term distribution. Third, the inputs of DSM are simply distributions, which makes it be potentially applied to other fields dealing with distributions.

DSM is also feasible since only a seed irrelevant documents/distribution is needed. Moreover, we presented solid theoretical justifications, proofs, and explanations of its solutions. Experimental evaluation also demonstrated the soundness of the proposed lemmas and propositions. DSM consistently and significantly outperforms RM in terms of both retrieval effectiveness and stability as indicated by the performance bias and variance, respectively. In addition, DSM (without the refinement step) has a very similar performance as RM when the seed irrelevant documents have been directly removed in the feedback

documents. This demonstrates that DSM can derive a less irrelevant term distribution, which supported our analysis in Section 5.1.2. Moreover, DSM (with refinement) can also outperform RM++, which is RM after the seed irrelevant documents have been directly removed in the feedback documents. Our evaluation based on performance bias-variance shows that DSM are often the most robust methods among the tested method in terms of the retrieval robustness, i.e., combined effect of the retrieval effectiveness and robustness.

In addition to the explicit feedback, we also develop automatic approaches to seed irrelevant distribution. We present the outlier document detection (denoted as OutlierD) to detect seed irrelevant documents. We also propose an outlier term detection (denoted as OutlierDT) to extract the irrelevant terms from feedback documents. Experiments have shown that DSM using OutlierD can improve the retrieval effectiveness of RM, and the improvements are significant when using OutlierDT to generate irrelevant distribution for DSM. Moreover, DSM using OutlierDT can be more robust than RM (with seed irrelevant documents removed), on some collections, e.g., AP8889 and WT10G. These observations suggest that the outlier methods can help DSM become more practical and automatic. DSM using automatic outlier detection can have comparable performance with DSM using explicit feedback (when $r_n = 0.1$). We will also keep investigating this line of research in the future work.

7.2 Future Works

In the future, we will endeavor to overcome the limitations of the works we have done and make the proposed theories and models more general. At first, we will systematically study the *model complexity* of IR models. Currently, the model complexity we are concerned about is only related to the query language model. In our opinion, the expanded query model is more complex than the original query model since the former has more query terms or more parameters. In the future, we would define a more general definition of model complexity and study the model complexity in a broader areas. For instance, we can investigate what kinds of ranking functions are more complex. A start point can be from the relation between the risk issue (Wang & Collins-Thompson 2011, Wang & Zhu 2009) of retrieval models and the model complexity of retrieval models. Intuitively, a model which is more complex can be more risky, resulting in a bigger performance variance and a smaller performance bias. We certainly need to construct a systematic evaluation to observe the relation between the model complexity and risk business.

The model complexity of retrieval models (e.g., the document ranking functions and query expansion methods, etc.) is expected to be a fundamental theory of IR. In the literature, there have been quite a lot of retrieval models proposed. However, few attention has been paid to the model complexity, which can be an intrinsic property in each category of retrieval models. To make clear this issue is not only of theoretical importance, but also

can guide the practice of formal retrieval models. For instance, we can qualitatively predict the retrieval effectiveness and/or retrieval stability before a retrieval model is applied to a task, since the model complexity is closely related to the bias and variance.

Guided by the understanding of model complexity, we will also explore self-adaptive retrieval models. We can set a parameter to control/adjust the model complexity for different queries. For example, in the combination between the original query model and the expanded query model, the combination coefficient can be used to adjust the model complexity. For different queries, different coefficients can be adopted. In the document weight smoothing, we will also try to develop adaptive methods to adaptively adjust the smoothing parameter for different queries. Moreover, the refinement parameter in DSM can also be used a parameter which is associated to the model complexity. It would be very useful that the refinement parameter for each query can be adaptively tuned automatically. Indeed, the adaptive methods can be different in different tasks. Overall, this direction is very interesting and we believe that it can help improve the retrieval effectiveness and/or stability.

The outlier detection has been demonstrated as an effective approach to seed irrelevant distribution for DSM. We will keep investigating this direction of research in our future work. In the theoretical point of view, we can connect the parameters (e.g., k nearest neighbors the outlier document detection and the term window size in the outlier term detection) to the model complexity and bias-variance analysis [2]. On the other hand, practically, we will keep improving the outlier techniques to further improve the retrieval effectiveness and/or stability.

Moreover, we will continue to explore the bias-variance analysis in other IR areas/tasks, e.g., the personalization task. Recall that we presented an initial exploration and provided the basic formulation in Chapter 7. In the future, we are going to investigate 1) When will the personalization be helpful? 2) When there will be a bias-variance tradeoff of the user satisfaction using personalization techniques? 3) How to design personalization techniques to balance such a tradeoff to optimize the user satisfaction? These three questions are essential in the user-centered IR research.

Furthermore, it is interesting to keep studying the analogy of photon polarization in IR, as we discussed in Chapter 7. In the current implementation of the Quantum-inspired fusion approach, the probabilities are actually estimated by the language modeling approaches. In the future, we would make the Quantum fusion approach have more fundamental difference from the traditional approaches. For example, we can try to integrate the quantum interference and tensor product into the current formulation.

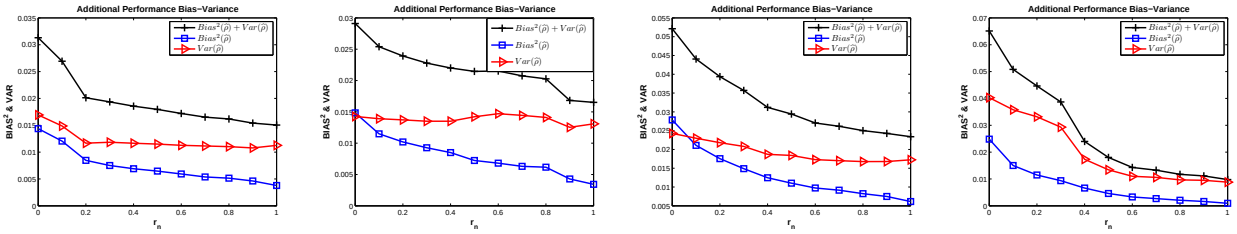
Our ultimate aim is to integrate all above research works into an Uncertainty principle of IR. The bias-variance tradeoff idea was initially inspired by one quantum theory – the

[2]Certainly, the k value in $k-nn$ algorithm is studied for bias-variance tradeoff in statistical machine learning.

Heisenberg's Uncertainty Principle. Bias-variance tradeoff can be considered as a kind of the uncertainty principle (Grenander 1952, Geman et al. 1992). We will explore more phenomenon of uncertainty principle of IR. For instance, the relevance estimation can be varied for different users, in different contexts, and during different times. We hope that the sound formulation of uncertainty principle can shed lights on the IR community and the science.

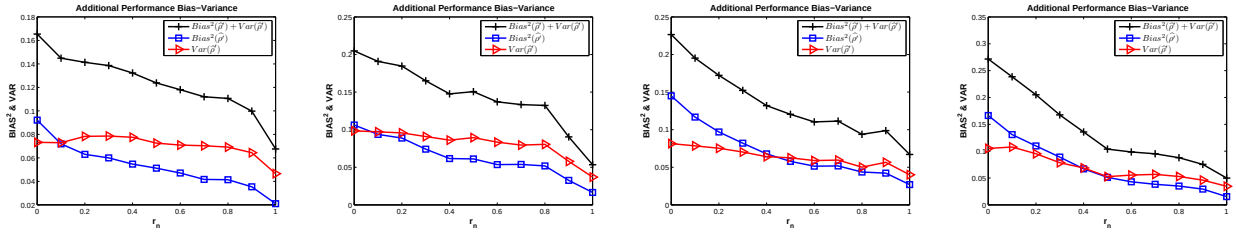
Chapter 8

Appendix



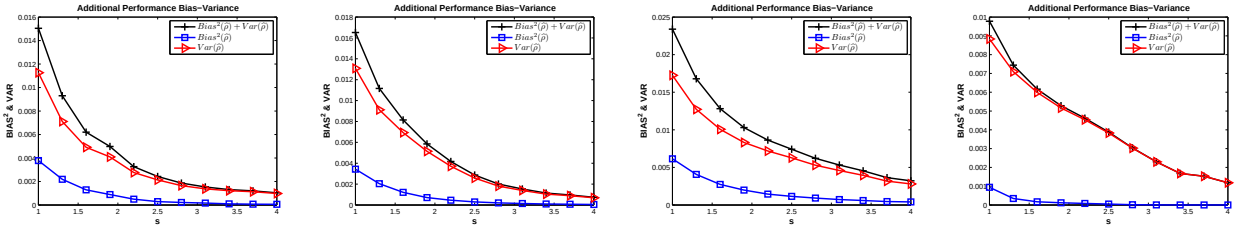
(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.1: Additional Performance bias-variance (based on $\widehat{\rho}$) of the expanded query model with non-relevant data. The x-axis shows non-relevance percentage r_n from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.



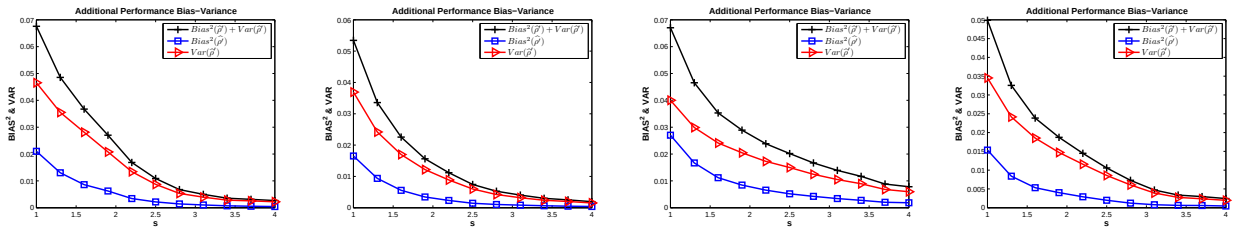
(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.2: Additional Performance bias-variance (based on $\widehat{\rho}'$) of the expanded query model with non-relevant data. The x-axis shows non-relevance percentage r_n from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.



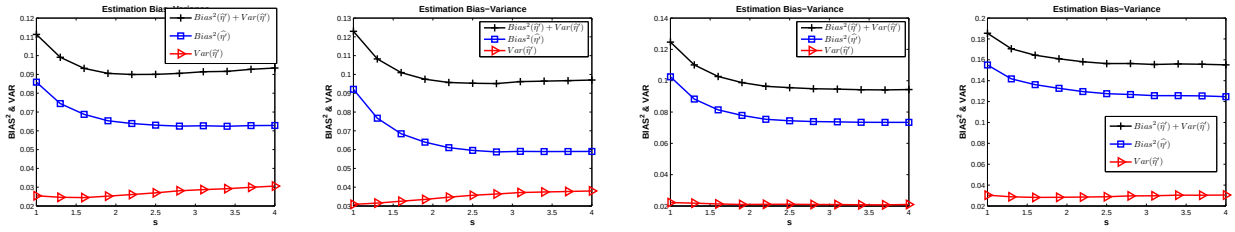
(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.3: Additional Performance bias-variance (based on $\widehat{\rho}$) of the expanded query models on relevant documents with smoothed document wight. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.



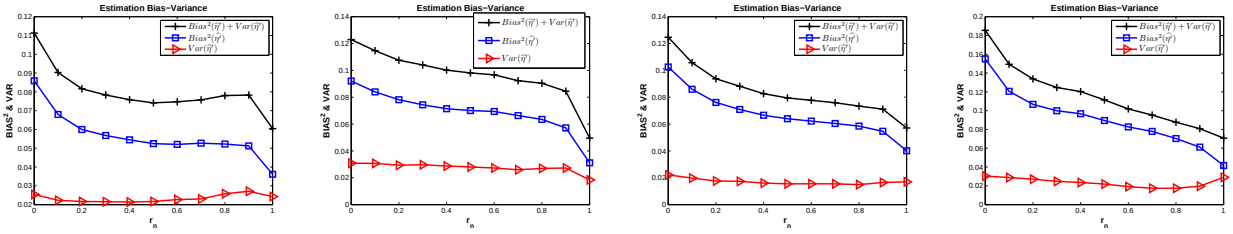
(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.4: Additional Performance bias-variance (based on $\widehat{\rho}'$) of the expanded query models on relevant documents with smoothed document wight. The x-axis shows smoothing parameter s from [1,4] with increment 0.3, and the y-axis represents the bias-variance results.



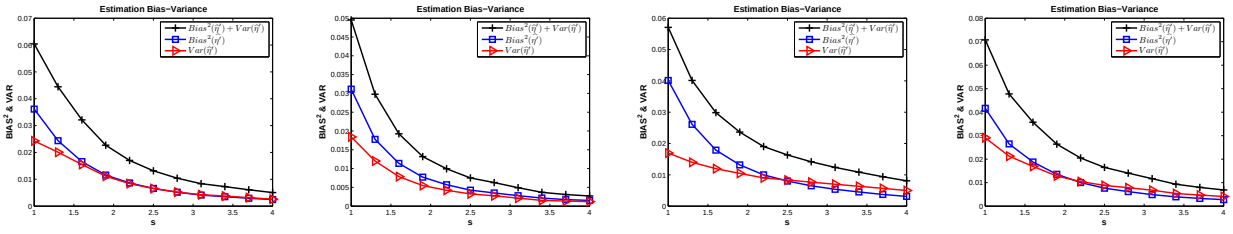
(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.5: Estimation bias-variance (based on JS-divergence) of the smoothed query model. The x-axis shows smoothing parameter s from [1, 4] with increment 0.3, and the y-axis represents the bias-variance results.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.6: Estimation bias-variance (based on JS-divergence) of the expanded query model with non-relevant data available. The x-axis shows non-relevance percentage r_n from [0,1] with increment 0.1, and the y-axis represents the bias-variance results.



(a) WSJ8792 (b) AP8889 (c) ROBUST2004 (d) WT10G

Figure 8.7: Estimation bias-variance (based on JS-divergence) of the expanded query model on relevant documents with smoothed document wight. The x-axis shows smoothing parameter s from [1, 4] with increment 0.3, and the y-axis represents the bias-variance results.

Chapter 9

Published Papers

- Peng Zhang, Yuexian Hou and Dawei Song. Approximating True Relevance Distribution from a Mixture Model Based on Irrelevance Data. In: *Proceedings of The 32nd Annual ACM SIGIR Conference (***SIGIR 2009***) (acceptance rate=16.5%)* . , pp. 107-114, 19-23 July 2009, Boston, USA

- Peng Zhang, Ulises Cervino Beresi, Dawei Song and Yuexian Hou, A Probabilistic Automaton for the Dynamic Relevance Judgement of Users, *in Proceedings of The 33nd ACM* **SIGIR** *Workshop on Simulation of Interaction*. pp. 17-18, 19-23 July 2010, Geneva, Switzerland

- Peng Zhang, Dawei Song, Jun Wang, Xiaozhao Zhao, Yuexian Hou, On Modeling Rank-Independent Risk in Estimating Probability of Relevance. Accepted by *the 7th Asia Information Retrieval Societies Conference (***AIRS 2011***)* (acceptance rate=24%), 18-20 December 2010, Dubai, United Arab Emirates. .

- Peng Zhang, Dawei Song, Xiaozhao Zhao and Yuexian Hou (2011). Investigating Query-Drift Problem from a Novel Perspective of Photon Polarization. *The 3rd International Conference on the Theory of Information Retrieval (***ICTIR 2011***)*, LNCS, pp. 332-336. 12-14 September 2011, Bertinoro, Italy.

- Xiaozhao Zhao, Peng Zhang, Dawei Song, Yuexian Hou (2011). A Novel Re-Ranking Approach Inspired by Quantum Measurement. **Best Poster Award** *in The 33rd European Conference on Information Retrieval (***ECIR'2011***)*, LNCS 6661, pp. 721-724. 19-21 April 2011, Dublin.

- Peng Zhang, Dawei Song, Xiaozhao Zhao, Yuexian Hou, A Study of Document Weight Smoothness in Pseudo Relevance Feedback. In *the 6th Asia Information Retrieval Societies Conference (***AIRS 2010***)* (acceptance rate=22%), LNCS 6458, pp. 521-538. 1-3 December 2010, Taipei.

- Peng Zhang, Wenjie Li, Yuexian Hou, Dawei Song. (2011) Developing Position Structure based Framework for Chinese Entity Relation Extraction. *ACM Transactions on Asian Language Information Processing (***TALIP***)*, Volume 10 Issue 3, September 2011.

- Tianxu Yan, K. Tamsin Maxwell, Dawei Song, Yuexian Hou, and Peng Zhang. Event-based Hyperspace Analogue to Language for Query Expansion. Accepted by: *the 48th Annual Meeting of the Association for Computational Linguistics (***ACL 2010***)*, pp. 120-125. 11-16 July 2010, Uppsala, Sweden.

- Yuexian Hou, Peng Zhang, Tingxu Yan, Wenjie Li and Dawei Song. Beyond Redundancies: A Metric Invariant Method for Unsupervised Feature Selection. *IEEE Transaction on Knowledge and Data Mining (***TKDE***)*, 22(3), pp. 348-364, 2010.

- Yuexian Hou, Peng Zhang, Xingxing Xu, Xiaowei Zhang and Wenjie Li. Nonlinear Dimensionality Reduction by Locally Linear Inlaying. *IEEE Transaction on Neural Networks (***TNN***)*. 20(2), pp. 300-315, 2009

- Leszek Kaliciak, Jun Wang, Dawei Song, Peng Zhang and Yuexian Hou (2011). Contextual Image Annotation and Quantum Theory Inspired Measurement for Integration of Textual and Visual Features. *The Fifth International Symposium on Quantum Interaction (QI'2011)*, Aberdeen, UK

- Peng Zhang, Dawei Song, Yuexian Hou, Jun Wang, Peter Bruza, Massimo Melucci and John McCall Automata Modeling for Cognitive Interference in Users' Relevance Judgment, accepted by **AAAI**-*Fall 2010 Symposium on Quantum Informatics for Cognitive, Social, and Semantic Processes*, pp. 125-133. Washington DC, November 11-13, 2010

- D. Song, M. Lalmas, C.J. van Rijsbergen, I. Frommholz, B. Piwowarski, J. Wang, P. Zhang , G. Zuccon, P.D. Bruza, S. Arafat, L. Azzopardi, E. Di Buccio, A. Huertas-Rosero, Y. Hou, M. Melucci, S. Rueger. How quantum theory is developing the field of information retrieval, *AAAI-Fall (QI) Symposium on Quantum Informatics for Cognitive, Social, and Semantic Processes*, Washington, DC, November 2010.

Bibliography

Abdul-Jaleel, N., Allan, J., Croft, W. B., Diaz, F., Larkey, L., Li, X., Metzler, D., Smucker, M. D., Strohman, T., Turtle, H. & Wade, C. (2004). Umass at trec 2004: Novelty and hard, *TREC '04*.

Agarwal, D., Gabrilovich, E., Hall, R., Josifovski, V. & Khanna, R. (2009). Translating relevance scores to probabilities for contextual advertising, *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pp. 1899–1902.

Amati, G., Carpineto, C., Romano, G. & Bordoni, F. U. (2004). Query difficulty, robustness and selective application of query expansion, *eds, European Conf. on IR Research*, Springer, pp. 127–137.

Angiulli, F. & Pizzuti, C. (2002). Fast outlier detection in high dimensional spaces, *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, PKDD '02, Springer-Verlag, London, UK, UK, pp. 15–26.

Arampatzis, A., Robertson, S. & Kamps, J. (2009). Score distributions in information retrieval, *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pp. 139–151.

Bai, J., Song, D., Bruza, P., Nie, J.-Y. & Cao, G. (2005). Query expansion using term relationships in language models for information retrieval, *CIKM*, pp. 688–695.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Brain, D. & Webb., G. (1999). On the effect of data set size on bias and variance in classification learning, *Proceedings of the Fourth Australian Knowledge Acquisition Workshop, University of New South Wales*, pp. 117–128.

Bruza, P. & Song, D. (2003). A comparison of various approaches for using probabilistic dependencies in language modeling, *SIGIR*, pp. 419–420.

Buckley, C. (2004). Why current ir engines fail, *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pp. 584–585.

Buckley, C. & Salton, G. (1995). Optimization of Relevance Feedback Weights, *SIGIR*, pp. 351–357.

Burgess, C., Livesay, K. & Lund, K. (1998). Explorations in context space: words, sentences, discourse, *Discourse Processes* **25**: 211–257.

Cao, G., Nie, J.-Y., Gao, J. & Robertson, S. (2008). Selecting good expansion terms for pseudo-relevance feedback, *SIGIR*, pp. 243–250.

Carlton, A. G. (1969). On the bias of information estimates, *Psychological Bulletin* **71**: 108–09.

Carpineto, C., de Mori, R., Romano, G. & Bigi, B. (2001). An information-theoretic approach to automatic query expansion, *ACM Trans. Inf. Syst.* **19**(1): 1–27.

Carpineto, C., Romano, G. & Giannini, V. (2002). Improving retrieval feedback with multiple term-ranking function combination, *ACM Trans. Inf. Syst.* **20**(3): 259–290.

Collins-Thompson, K. (2009a). Accounting for stability of retrieval algorithms using risk-reward curves, *In Proceedings of the SIGIR 2009 Workshop on the Future of IR Evaluation*, pp. 27–28.

Collins-Thompson, K. (2009b). Reducing the risk of query expansion via robust constrained optimization, *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, ACM, New York, NY, USA, pp. 837–846.

Collins-Thompson, K. B. (2008). *Robust Model Estimation Methods for Information Retrieval*, PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA.

Collins-Thompson, K. & Callan, J. (2005). Query expansion using random walk models, *Proceedings of the 14th ACM international conference on Information and knowledge management*, CIKM '05, pp. 704–711.

Collins-Thompson, K. & Callan, J. (2007). Estimation and use of uncertainty in pseudo-relevance feedback, *SIGIR '07: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 303–310.

Crouch, C. J., Crouch, D. B., Chen, Q. & Holtz, S. J. (2002). Improving the retrieval effectiveness of very short queries, *Inf. Process. Manage.* **38**(1): 1–36.

Diaz, F. (2005). Regularizing ad hoc retrieval scores, *CIKM '05: Proceedings of the 14th ACM Conference on Information and Knowledge Management*, pp. 672–679.

Diaz, F. (2007). Regularizing query-based retrieval scores, *Inf. Retr.* **10**(6): 531–562.

Diaz, F. (2008). Improving relevance feedback in language modeling with score regularization, *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 807–808.

Dillon, J. V. & Collins-Thompson, K. (2010). A unified optimization framework for robust pseudo-relevance feedback algorithms, *Proceeding of the 19th ACM conference on Information and knowledge management*, pp. 1069–1078.

Duda, R. O., Hart, P. E. & Stork, D. G. (2001). *Pattern Classification (2nd Edition)*, 2 edn, Wiley-Interscience.

Dumais, S., Joachims, T., Bharat, K. & Weigend, A. (2003). SIGIR 2003 workshop report: implicit measures of user interests and preferences, *SIGIR Forum* **37**(2): 50–54.

Gao, J., Nie, J.-Y., Wu, G. & Cao, G. (2004). Dependence language model for information retrieval, *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pp. 170–177.

Geman, S., Bienenstock, E. & Doursat, R. (1992). Neural networks and the bias/variance dilemma, *Neural Comput.* **4**: 1–58.

Gey, F. C. (1994). Inferring probability of relevance using the method of logistic regression, *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '94, pp. 222–231.

Ghahramani, Z., , Ghahramani, Z. & chul Kim, H. (2003). Bayesian classifier combination.

Gleason, A. M. (1957). Measures on the closed subspaces of a hilbert space, *Journal of Mathematics and Mechanics* **6**: 885–893.

Grenander, U. (1952). On empirical spectral analysis of stochastic processes, *Arkiv for Matematik* **1**: 503–531.

Harman, D. & Buckley, C. (2004). The nrrc reliable information access (ria) workshop, *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pp. 528–529.

He, B. & Ounis, I. (2009). Finding good feedback documents, *CIKM '09: Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 2011–2014.

Henzinger, M. R., Motwani, R. & Silverstein, C. (2002). Challenges in Web Search Engines, *SIGIR Forum* **36**(2): 11–22.

Hou, Y., He, L., Zhao, X. & Song, D. (2011). Pure high-order word dependence mining via information geometry, *ICTIR*, pp. 64–76.

Hou, Y., Yan, T., Zhang, P., Song, D. & Li, W. (2010). On tsallis entropy bias and generalized maximum entropy models, *arXiv-CoRR* **abs/1004.1061**.

Hyvärinen, A. & Oja, E. (n.d.). Independent component analysis: algorithms and applications, *Neural Netw.* **13**(4-5): 411–430.

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

Kozorovitzky, A. K. & Kurland, O. (2011). Cluster-based fusion of retrieved lists, *SIGIR*, pp. 893–902.

Kurland, O. & Lee, L. (2004). Corpus structure, language models, and ad hoc information retrieval, *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 194–201.

Lafferty, J. D. & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval, *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 111–119.

Lafferty, J. D. & Zhai, C. (2003). Probabilistic relevance models based on document and query generation, *Language Modeling and Information Retrieval*, Kluwer Academic Publishers, pp. 1–10.

Lang, H., Metzler, D., Wang, B. & Li, J.-T. (2010). Improved latent concept expansion using hierarchical markov random fields, *CIKM*, pp. 249–258.

Lavrenko, V. (2004). *A Generative Theory of Relevance*, PhD thesis, University of Massachusetts.

Lavrenko, V. & Croft, W. B. (2001). Relevance-based language models, *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 120–127.

Lebanon, G. (2010). Bias, variance, and mse of estimators.

Li, X. (2008). A new robust relevance model in the language model framework, *Inf. Process. Manage.* **44**(3): 991–1007.

Lipka, N. & Stein, B. (2011). Robust models in information retrieval, *8th International Workshop on Text-based Information Retrieval (TIR)*.

Liu, X. & Croft, W. B. (2004). Cluster-based retrieval using language models, *SIGIR '04: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM, New York, NY, USA, pp. 186–193.

Lv, Y. & Zhai, C. (2009). Adaptive relevance feedback in information retrieval, *CIKM '09: Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 255–264.

Lv, Y., Zhai, C. & Chen, W. (2011). A boosting approach to improving pseudo-relevance feedback, *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pp. 165–174.

Manning, C. D., Raghavan, P. & Schtze, H. (2008). *Introduction to Information Retrieval*, Cambridge University Press, New York, NY, USA.

Maron, M. E. & Kuhns, J. L. (1960). On relevance, probabilistic indexing and information retrieval, *J. ACM* **7**: 216–244.

Mei, Q., Zhang, D. & Zhai, C. (2008). A general optimization framework for smoothing language models on graph structures, *SIGIR '08: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 611–618.

Meister, L., Kurland, O. & Kalmanovich, I. G. (2011). Re-ranking search results using an additional retrieved list, *Inf. Retr.* **14**(4): 413–437.

Metzler, D. & Croft, W. B. (2005). A markov random field model for term dependencies, *SIGIR*, pp. 472–479.

Metzler, D. & Croft, W. B. (2007). Latent concept expansion using markov random fields, *SIGIR*, pp. 311–318.

Miller, D. R. H., Leek, T. & Schwartz, R. M. (1999). A hidden markov model information retrieval system, *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 214–221.

Miller, G. (1955). Note on the bias of information estimates, *Information theory in psychology II-B* pp. 95–100.

Mitra, M., Singhal, A. & Buckley, C. (1998). Improving automatic query expansion, *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pp. 206–214.

Mizzaro, S. (1996). Relevance: The whole (hi)story, *Journal of the American Society for Information Science* **48**: 810–832.

Ogilvie, P. & Callan, J. (2002). Experiments using the lemur toolkit, *TREC '02: Proceedings of the ACM 11th Text Retrieval Conference*, pp. 103–108.

Paninski, L. (2003). Estimation of entropy and mutual information, *Neural Comput.* **15**(6): 1191–1253.

Perlich, C., Provost, F. J. & Simonoff, J. S. (2003). Tree induction vs. logistic regression: A learning-curve analysis, *Journal of Machine Learning Research* **4**: 211–255.

Pickens, J. & MacFarlane, A. (2006). Term context models for information retrieval, *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pp. 559–566.

Ponte, J. M. & Croft, W. B. (1998). A language modeling approach to information retrieval, *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 275–281.

Ramaswamy, S., Rastogi, R. & Shim, K. (2000). Efficient algorithms for mining outliers from large data sets, *SIGMOD Rec.* **29**(2): 427–438.

Rieffel, E. G. & Polak, W. (2000). An introduction to quantum computing for non-physicists, *ACM Comput.Surveys.* **32**: 300–335.

Robertson, S. (2005). On event spaces and probabilistic models in information retrieval, *Inf. Retr.* **8**(2): 319–329.

Robertson, S. E. (1977). The probability ranking principle in IR, pp. 294–304.

Robertson, S. E. & Spärck Jones, K. (1976). Relevance weighting of search terms, *Journal of the American Society for Information Science* **27**(3): 129–146.

Robertson, S. E. & Zaragoza, H. (2009). The probabilistic relevance framework: Bm25 and beyond, *Foundations and Trends in Information Retrieval* **3**(4): 333–389.

Rocchio, J. J. (1971). Relevance feedback in information retrieval, *The SMART Retrieval System - Experiments in Automatic Document Processing*, Prentice Hall, pp. 313–323.

Rodgers, J. L. & Nicewander, A. W. (1988). Thirteen ways to look at the correlation coefficient, *The American Statistician* **42**: 59–66.

Salton, G., Wong, A. & Yang, C. S. (1975). A vector space model for automatic indexing, *Commun. ACM* **18**(11): 613–620.

Song, D., Huang, Q., Bruza, P. & Lau, R. (2012). An aspect query language model based on query decomposition and high-order contextual term associations, *Computational Intelligence* **28**(1): 1–23.

Song, D., Huang, Q., Rüger, S. M. & Bruza, P. (2008). Facilitating query decomposition in query language modeling by association rule mining using multiple sliding windows, *ECIR*, pp. 334–345.

Song, D., Lalmas, M. & van Rijsbergen et al., C. J. (2010). How quantum theory is developing the field of information retrieval., *QI*, pp. 105–108.

Sparck Jones, K., Robertson, S., Hiemstra, D. & Zaragoza, H. (2003). Language modelling and relevance, *Language Modeling and Information Retrieval*, Kluwer Academic Publishers, pp. 57–31.

Spärck Jones, K., Walker, S. & Robertson, S. E. (2000). A probabilistic model of information retrieval: development and comparative experiments, *Inf. Process. Manage.* **36**: 779–808.

Tao, T. & Zhai, C. (2006). Regularized estimation of mixture models for robust pseudo-relevance feedback, *SIGIR*, pp. 162–169.

Teevan, J., Dumais, S. T. & Horvitz, E. (2010). Potential for personalization, *ACM Trans. Comput.-Hum. Interact.* **17**(1).

Tombros, A. & van Rijsbergen, C. J. (2004). Query-sensitive similarity measures for information retrieval, *Knowl. Inf. Syst.* **6**(5).

Valentini, G., Dietterich, T. G. & Cristianini, N. (2004). Bias-variance analysis of support vector machines for the development of svm-based ensemble methods, *Journal of Machine Learning Research* **5**: 725–775.

van Rijsbergen, C. J. (1979). *Information Retrieval*, Butterworths.

van Rijsbergen, C. J. (2004). *The Geometry of Information Retrieval*, Cambridge University Press, New York, NY, USA.

Wang, J. (2009). Mean-variance analysis: A new document ranking theory in information retrieval, *ECIR*, pp. 4–16.

Wang, J. & Collins-Thompson, K. (2011). Statistical information retrieval modelling: from the probability ranking principle to recent advances in diversity, portfolio theory, and beyond, *CIKM*, pp. 2603–2604.

Wang, J. & Zhu, J. (2009). Portfolio theory of information retrieval, *SIGIR '09: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 115–122.

Wang, X., Fang, H. & Zhai, C. (2008). A study of methods for negative relevance feedback, *SIGIR*, pp. 219–226.

Wei, X. & Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval, *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 178–185.

White, R., Ruthven, I. & Jose, J. (2005). A Study of Factors Affecting the Utility of Implicit Relevance Feedback, *SIGIR*, pp. 35–42.

Yan, T., Maxwell, T., Song, D., Hou, Y. & Zhang, P. (2010). Event-based hyperspace analogue to language for query expansion, *ACL (Short Papers)*, pp. 120–125.

Zhai, C. (2007). A brief review of information retrieval models,, *Technical report, Dept. of Computer Science, UIUC*.

Zhai, C. & Lafferty, J. D. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval, *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 334–342.

Zhai, C. & Lafferty, J. D. (2006). A risk minimization framework for information retrieval, *Inf. Process. Manage.* **42**(1): 31–55.

Zhang, P., Beresi, U. C., Song, D. & Hou, Y. (2010). A probabilistic automaton for the dynamic relevance judgement of users, *in Proceedings of The 33nd ACM SIGIR Workshop on Simulation of Interaction (SIGIR-SimInt)*, pp. 17–18.

Zhang, P., Hou, Y. & Song, D. (2009). Approximating true relevance distribution from a mixture model based on irrelevance data, *SIGIR '09: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 107–114.

Zhang, P., Song, D., Hou, Y., Wang, J. & Bruza, P. (2010). Automata modeling for cognitive interference in users' relevance judgment, *AAAI-Fall 2010 Symposium on Quantum Informatics for Cognitive, Social, and Semantic Processes (AAAI-QI)*.

Zhang, P., Song, D., Wang, J., Zhao, X. & Hou, Y. (2011). On modeling rank-independent risk in estimating probability of relevance, *AIRS*, pp. 13–24.

Zhang, P., Song, D., Zhao, X. & Hou, Y. (2010). A study of document weight smoothness in pseudo relevance feedback, *AIRS*, pp. 527–538.

Zhang, P., Song, D., Zhao, X. & Hou, Y. (2011). Investigating query-drift problem from a novel perspective of photon polarization, *ICTIR*, pp. 332–336.

Zhu, J., Wang, J., Cox, I. J. & Taylor, M. J. (2009). Risky business: modeling and exploiting uncertainty in information retrieval, *SIGIR '09: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 99–106.

Zighelnic, L. & Kurland, O. (2008). Query-drift prevention for robust query expansion, *SIGIR*, pp. 825–826.

Zucchini, W., Berzel, A. & Nenadic, O. (2005). Applied smoothing techniques, *Lecture notes, Institute for Statistics and Econometrics, University of Gottingen.*