The Hong Kong Polytechnic University
Department of Computing

A Study of Document-Context Models in Information Retrieval

WU Ho Chung

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

August 2010

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____(Signed)

WU Ho Chung (Name of student)

Abstract

In this thesis we study new retrieval models which simulate the "local" relevance decision-making for every term location in a document, these local relevance decisions are then combined as the “document-wide” relevance decision for the document. Local relevance decision for a term *t* occurred at the *k*-th location in a document is made by considering the document-context which is the window of terms centred at the term *t* at the *k*-th location. Therefore, different relevance scores (preferences) are obtained for the same term *t* at different locations in a document depending on its document-contexts. This differs from traditional models which term *t* receives the same score disregard of its locations in a document.

Particularly, a hybrid document-context model is studied which is the combination of various existing effective models and techniques. It estimates the relevance decision preference of document-contexts as the log-odds and uses smoothing techniques as found in language models to solve the problem of zero probabilities. It combines the estimated preferences of document-contexts using different types of aggregation operators that comply with the relevance decision principles. The model is evaluated using retrospective experiments with full relevance information to reveal the potential of the model. The model obtained a mean average precision of 60% - 80% in retrospective experiments using different TREC ad hoc English collections and the NTCIR-5 ad hoc Chinese collection. The experiments showed that the operators that are consistent with aggregate relevance principle were effective in combining the estimated preferences of document-contexts. Besides retrospective experiments, we also use top 20 documents from the initial ranked list to perform relevance feedback experiments with a probabilistic document-context model and the results are promising.

We also showed that when the size of the document-contexts is shrunk to unity, the document-context model is simplified to a basic ranking formula

that directly corresponds to the TF-IDF term weights. Thus TF-IDF term weights can be interpreted as making relevance decisions. This helps to establish a unifying perspective about information retrieval as relevance decision-making and to develop advance TF-IDF-related term weights for future elaborate retrieval models. Empirically, we show that, using four TREC ad hoc retrieval data collections, the IDF of a term *t* is related to the probability of randomly picking a non-relevant usage of the term *t*.

Lastly, we apply the notion of document-context to develop a new relevance feedback algorithm. Instead of letting user to judge the documents from the top in the ranked document list, we split the ranked document list into multiple lists of document-contexts. Therefore, the judgement of relevance of the documents is not done sequentially. This is called active feedback and we show that in the experiments with various TREC data collections, our new relevance feedback algorithm using document-contexts obtained better results than the conventional relevance feedback algorithm and this is done more reliably than a maximal marginal relevance (MMR) method which does not use document-contexts. The experimental results suggest that using document-contexts can improve retrieval effectiveness.

Publications Arising from the Thesis

WU, H.C., LUK, R.W.P., WONG, K.F., KWOK, K.L., AND LI, W.J. 2005. A retrospective study of probabilistic context-based retrieval. In *Proceedings of ACM SIGIR 05*, pp. 663-664.

WU, H.C., LUK, R.W.P., WONG, K.F., AND KWOK, K.L. 2006. Probabilistic document-context based relevance feedback with limited relevance judgment. In *Proceedings of ACM CIKM 06*, pp. 854-855.

WU, H.C., LUK, R.W.P., WONG, K.F., AND KWOK, K.L. 2007. A retrospective study of a hybrid document-context based retrieval model. *Information Processing & Management*, 43, 5, 1308-1331.

WU, H.C., LUK, R.W.P., WONG, K.F., AND KWOK, K.L. 2008. Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, 26, 3, 13.

WU, H.C., DANG, E.K.F., LUK, R.W.P., NGAI, G., LI, Y., ALLAN, J., KWOK, K.L., AND WONG, K.F. 2008. Evaluating a Novel Kind of Retrieval Models Based on Relevance Decision Making in a Relevance Feedback Environment. In *Proceedings of TREC 08*.

DANG, E.K.F., WU, H.C., LUK, R.W.P., AND WONG, K.F. 2009. Building a framework for the probability ranking principle by a family of expected weighted rank. *Transactions on Information Systems (TOIS)*, 27, 4, 20.

Acknowledgements

First and foremost, I would like to thank my supervisor, Dr. Robert W.P. Luk, for his encouragement, support, and guidance throughout my study in these years. This thesis would not have been possible without his careful supervision. I have benefited greatly from his insightful advices and I am motivated by his enthusiasm in doing research. My immense gratitude goes to him.

I would like to thank Prof. K.F. Wong, Prof. K.L. Kwok and Dr. W.J. Li for their support by providing helpful comments to my research paper manuscripts.

I would also like to thank my fellow research group members, W.S Wong, Y.H. Li, Andy D.Y. Wang and Edward K.F. Dang.

Last, I must thank my parents and my beloved, LAM Ka Mei, for their care, love, and understanding about me.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In Information Retrieval (IR), the ultimate goal is to find relevant information effectively and efficiently. Examples of earlier IR tasks include finding library records and scientific publications. The target users of IR systems were limited to professionals such as scientists and journalists. However, the situation changed with the invention of the World Wide Web in the 1990s. Since then the size of the Web grows exponentially, and begins the era of electronic information. Nowadays, the number of web pages on the Web is in terms of billions and they are readily reached by most of the people in the world. With the vast amount and variety of information, the problem of finding relevant information becomes essential to people's everyday lives.

Usually, when using an IR system, user's information need is transformed to a query which consists of one or more keywords and then entered to the system. The IR system then matches those input keywords with the contents of documents in the indexed collection. Matched documents are ranked using certain methods and finally a list of documents as an output is presented to the user. Lexical problems may arrive during the process which could cause difficulties in finding relevant documents for the user. First, the transformation from user's information need to the query terms may be inaccurate. That is, incorrect keywords are used to represent the information need. Second, the problem of polysemy causes ambiguity when matching query terms with document terms. A polysemy is a word or phrase with multiple meanings. For example, the keyword "bank" exists in a query may refer to a "commercial bank" or a "river bank". The existence of polysemy in natural language may cause non-relevant documents to be retrieved. Third, the term mismatch problem in which the same concept is referred to by different words. As a result, relevant documents that do not contain query terms are not retrieved.

During the course of IR research in the last decades, many retrieval models have been developed and investigated. Generally speaking, a retrieval model defines:

(1) the representation of documents,

(2) the representation of queries, and

(3) the ranking function.

For example, the vector space models [Salton et al., 1975; Wong et al., 1985] use vectors of features (e.g., index terms) for representing documents and queries. The ranking function is calculating the deviation of angles (i.e., the cosine similarity measure) between each of the document vectors and the query vector. A document vector with a smaller angle of deviation is considered more similar to the query and therefore the document would be ranked higher. Many of the retrieval models are based on a variety of mathematical frameworks [Dominich, 2000]. These models provide a system point of view of how to retrieve documents that are sufficiently relevant such that they satisfy a user's information need.

A retrieval model can also be thought of as simulating the human user when making relevance decisions in the retrieval process [Bollmann and Wong, 1987]. In this case, the ranking of the relevance of the documents to the user's information need is in terms of preferences [Yao and Wong, 1991].

In this thesis, we investigate retrieval models that use "document-contexts" to simulate a human user making "local" relevance decisions. A document-context is essentially a concordance or a keyword in context (KWIC) [Kupiec et al., 1995]. Figure 1.1 shows some example document contexts containing a query term in the title query, "Hubble Telescope Achievements". By using "document-contexts", we try to deal with the problem caused by polysemy which mentioned above. The meaning of a single term could be ambiguous while a term with context should have definite meaning. For example, in Figure 1.1, we see several contexts for the term "Hubble" which is a query term. In some contexts the term "Hubble"

refers to the person Edwin P. Hubble while it refers to the Hubble telescope in other contexts. It is intuitive that keyword in context (KWIC) is important for users to make local relevance judgments, as a result we do not perform user research on this.

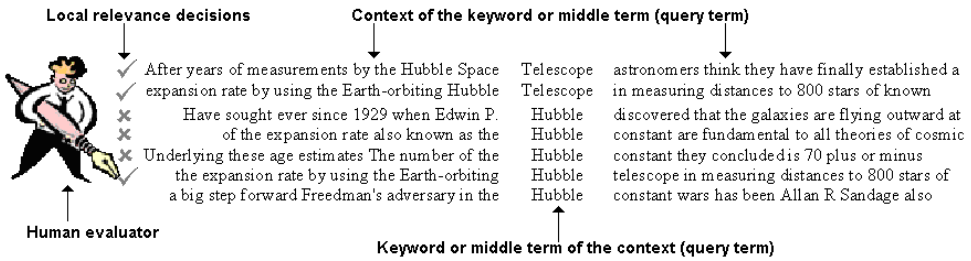

Figure 1.1: Example contexts extracted from a relevant document (NYT19990525.0358) in the TREC-2005 robust-track data collection. The query is "Hubble Telescope Achievements" with query id = 303. Contexts with (✓) are judged relevant after human examination of the contexts.

1.1 Research Problems and Motivations

An important element in retrieval models is how to weight the terms in a document. That is, the terms in a document are weighted using different factors. The score of a document is then the combination of the individual term weights. The term-weighting function of a document usually depends on three main factors [Salton and Buckley, 1988]: (1) the term frequency (TF) of the term, (2) the inverse document frequency (IDF) of the term, and (3) the document length. Using the 3 main factors, a well-know and common term weighting method is the TF-IDF. In general, the TF-IDF weight of a term *t* in document *d* is the same no matter where term *t* occurs in *d*. In our study, we believe that the locations of a term in a document play an important role in determining the relevance of the document to a query. Since different locations of a term in a document reveal different contexts which would individually affect the relevance of the document.

By considering the locations of terms in a document, we develop document-context based retrieval models which simulates human user when making relevance decisions. The document-context based retrieval models provide a relevance score for every location in a document (i.e., the local relevance

decisions). The relevance scores of each of the locations are then combined to form the document score (i.e., the document-wide relevance decision). We investigated different methods of combining the local relevance decisions by following the relevance decision principles [Kong et al., 2004], namely the *Disjunctive Relevance Decision* (DRD) principle, the *Aggregate Relevance Decision* (ARD) principle and the *Conjunctive Relevance Decision* (CRD) principle.

While developing new retrieval models, we are motivated to justify theoretically and empirically that the document-context based retrieval models are compatible with the common TF-IDF term weighting models. Specifically, by making the Minimal Context assumption which shrinks the context size to unity such that the local relevance decisions no longer depend on locations, we show that the TF-IDF term weight models are actually a special case in our proposed document-context based retrieval models. The significance of this justification is that potentially there is a unifying perspective about information retrieval (IR) as relevance decision-making.

We are also motivated by enhancing retrieval effectiveness using the document-context based retrieval models. We perform relevance feedback experiments and compare our models with the stat-of-the-art retrieval model. Lastly, by using the notion of document-contexts, we are motivated to investigate new relevance feedback algorithm such that user's satisfaction during the relevance feedback can be enhanced by judging less non-relevant documents.

1.2 Contributions and their Significance

In this section we briefly state the main contributions of our work and their significance.

1. Interpreting TF-IDF as making relevance decisions

We show that theoretically and empirically, TF-IDF term weights can be the outcome of modeling relevance decision-making. The significance of this justification is that potentially there is a unifying perspective about information retrieval (IR) as relevance decision-making. Many past retrieval models are already related to relevance decision-making; for example, the binary independence retrieval (BIR) model [Robertson and Sparck Jones, 1976], the logistic regression model [Cooper et al., 1992], the vector space model [Salton et al., 1975], the Boolean model [Wong et al., 1986], and the extended Boolean model [Salton et al., 1983]. However, it is not known whether TF-IDF term weights are related to relevance decision-making because they were originally not conceived in this way. Instead, the term frequency factor was originally thought to be indicative of document topic [Luhn, 1958], and the inverse document frequency (IDF) is reasoned [Sparck Jones, 1972] on the basis of Zipf law.

2. By using two language models to model relevance and irrelevance independently, the Binary Independence Language Model is developed and it shows improvement in retrieval effectiveness using various TREC data collections

We have investigated the probabilistic document-context based retrieval model. The model uses the log-odds ratio that combines two relevance decision component models which are designed to mimic human relevance decision-making. They simulate what a human evaluator does and make local relevance decisions at each document location. These local relevance decisions of a document are combined to produce the final document-wide relevance decision for the document. Retrospective experiments with our models have produced mean average precisions between 70% and 80% using various reference TREC ad hoc retrieval test collections. For relevance feedback using the top 20 ranked, judged documents, our model using fixed parameter values performs statistically significantly better than support vector machines and the highly effective, modified Markov random

field model with a 90% confidence interval across different TREC collections. These results show that the proposed theory and its retrieval model are promising.

3. The split-list approach to relevance feedback is proposed which is new and the experiment results show that the new approach results in enhancement in user's satisfaction during relevance feedback

We have proposed a new algorithm for relevance feedback in information retrieval which uses document-contexts by splitting the retrieval list into sub-lists according to the query term patterns exist in the top ranked documents. Query term patterns include single query term, a pair of query terms occur in a phrase and in proximity. By considering the document-contexts of the query patterns, more relevance documents can be found during relevance feedback which can enhance user's satisfaction.

1.3 Outline

The rest of this thesis is organized as follows

Chapter 2 Literature review: In this chapter we describe information retrieval models in the literature which are related to our work. In particular the binary independence retrieval (BIR) model [Robertson and Sparck Jones, 1976]. Other models which also use the concept of document-contexts are also reviewed.

Chapter 3 A retrospective study of a hybrid document-context based retrieval model: This chapter describes our novel retrieval model that is based on contexts of query terms in documents (i.e., document-contexts). The model explicitly takes into account of the document-contexts instead of implicitly using the document-contexts to find query expansion terms. The model is a hybrid of various existing effective models and techniques. We tested the model retrospectively (i.e., with the presence of relevance

information) to show its potential and have a better understanding of the model.

Chapter 4 Interpreting TF-IDF Term weights as making relevance decisions: In this chapter we investigate a probabilistic non-relevance decision model. By assuming the Minimal Context assumption, it forms a basis to interpret the TF-IDF term weights as making relevance decisions.

Chapter 5 Probabilistic document-context based retrieval model: By no longer assuming the Minimal Context assumption, in this chapter, we develop a binary independence language model and experiment it in using relevance feedback experiments.

Chapter 6 A split-list approach to relevance feedback in information retrieval: We describe a new algorithm for relevance feedback which applied the document-contexts. The objectives are to (a) find more relevance documents and (b) find documents with higher diversity, hence enhancing user's satisfaction during relevance feedback.

Chapter 7 Conclusion and future work: This chapter summarize the thesis and describe some items for possible future work.

Chapter 2

Literature Review

In this chapter we describe information retrieval models in the literature which are related to our work. The main focus of our work is to investigate the use of document-context in information retrieval. A term *t* having multiple meanings is called a polyseme. With the existence of polysemes, a non-relevant document may be retrieved even that it contains the same term as in the query. We believe that using the context of the term *t* in the document can alleviate the problem caused by polysemes, since the meaning of the term *t* can be clarified by its neighbouring terms (i.e., the document-context of the term *t*). Therefore, the location of a term in a document plays an important role in determining the meaning of the term. As a result, we incorporate positional information of terms in the retrieval model which traditional models do not. In Chapter 3 (A Retrospective Study of a Hybrid Document-context Based Retrieval Model), we develop a hybrid document-context model based on the well-known Binary Independence Retrieval (BIR) model [Robertson and Sparck Jones, 1976]. In Chapter 4 (Interpreting TF-IDF term weights as making relevance decision), we show that when we shrink the context size to unity, our document-context model can be interpreted as using TF-IDF term weights which are similar to the term weights used in the empirically successful BM25 model by Robertson and Walker [1994]. The BM25 model is an approximation to the 2-Poisson model [Harter 1974, 1975a, 1975b; Bookstein and Swanson, 1974; Robertson et al., 1980] while the 2-Poisson model is related to the BIR model. We briefly describe these traditional probabilistic retrieval models in section 2.1.

There are other works in the literature which use document-context similar to ours. Some of these works include the integration of collocation statistics into probabilistic retrieval model by Vechtomova and Robertson [2000], modeling term dependence using Markov random field by Metzler and

Croft [2005], the use of term context models for information retrieval by Pickens and Macfarlane [2006] and the use of lexical cohesion between query terms by Vechtomova et al. [2006]. These related works are described in section 2.2.

When considering document-contexts, we divide a document into smaller pieces using a sliding window. A window of terms with a centre term *t* is said to be the context of the term *t*. This is similar to passage-based retrieval which considers passages instead of the whole document. In section 2.3, we briefly review passage-based retrieval.

Most of our experiments are done in a relevance feedback environment, that is, some or full relevance information is available to the retrieval model. Therefore, we briefly review works on relevance feedback in information retrieval in section 2.4.

2.1 The Traditional Probabilistic Retrieval Models

The probabilistic approach to retrieval was first presented by Maron and Kuhns [1960] in 1960. The idea of using probability theory in information retrieval has generated the development of a variety of probabilistic retrieval models which differ by the estimation of probabilities in the ranking functions. For examples, the Binary Independence Retrieval (BIR) model by Robertson and Sparck Jones [1976], the logistic regression model by Cooper et al. [1992, 1993], the TF-IDF term weights in the BM25 model by Robertson and Walker [1994] which is based on the 2-Poisson model [Harter 1974, 1975a, 1975b; Bookstein and Swanson, 1974; Robertson, Van Rijsbergen and Porter, 1981], the language model by Ponte and Croft [1998], Zhai and Lafferty [2004] and Lavrenko and Croft [2001,2003], and more recently the divergence models by Amati and Van Rijsbergen [2002]. These models either minimize the (Bayesian) risks (e.g., the BIR model and language model [Zhai and Lafferty, 2006]), or they accept the Probabilistic Ranking Principle (PRP) [Robertson, 1977] as the best way to rank

documents or maximize the information gain [Amati and Van Rijsbergen, 2002] or optimize the cross-entropy [Lavrenko and Croft, 2003].

The Probability Ranking Principle (PRP) [Robertson, 1977] states that the greatest retrieval effectiveness is achieved when documents are ranked in the decreasing order of probabilities of relevance to the query, while the probabilities are estimated on all data available to the retrieval system. In Chapter 3 (A Retrospective Study of a Hybrid Document-context Based Retrieval Model), we show that our document-context model follows the PRP by experimenting the model with different amount of relevance information presence to the model.

2.1.1 The Binary Independence Retrieval (BIR) Model

In the BIR model [Robertson and Sparck Jones, 1976], the basic question to ask for each document and each query is:

What is the probability that this document is relevant to this query?

Denote a binary relevance variable $R \in \{0, 1\}$which models the relevance of documents, R equals to 1 means relevant while R equals to 0 means non-relevant. For a term t belongs to the vocabulary V (i.e., $t \in V$), the BIR model considers only the presence or absence of the term t in a query q and a document d. Hence, a query q is represented by a set of binary term occurrence variables $q_t \in \{0, 1\}$where q_t equals to 1 if q contains t and q_t equals to 0 if q does not contain t (i.e., $q \in \{0, 1\}^{|V|}$). Similarly, a document d is represented by a set of term occurrence variables $d_t \in \{0, 1\}$ where d_t equals to 1 if d contains t and d_t equals to 0 if d does not contain t (i.e., $d \in \{0, 1\}^{|V|}$). To rank the documents, the BIR model tries to estimate the probability of relevance of a given document d:

$$P(R = 1 \mid d, q)$$

Using Bayes' Theorem,

$$P(R=1|d,q)=\frac{P(d|R=1,q)P(R=1|q)}{P(d|q)} \tag{2.1}$$

In order to avoid the estimation of $P(d \mid q)$, the odds is used while preserving the ranking order of documents ($\overset{rank}{=}$ is a binary relation called rank equivalence [Lafferty and Zhai, 2001] that preserves the ranking of both sides of the relation by some monotonic transformation):

$$\begin{aligned} P(R=1|d,q) &\overset{rank}{=} \frac{P(R=1|d,q)}{P(R=0|d,q)} \\ &= \frac{P(d|R=1,q)P(R=1|q)}{P(d|R=0,q)P(R=0|q)} \\ &\overset{rank}{=} \frac{P(d|R=1,q)}{P(d|R=0,q)} \end{aligned} \tag{2.2}$$

By asserting the independence assumption that the occurrence of terms in a document is conditionally independent given a relevance class, $P(d \mid R, \mathrm{q})$ can be expanded by the multiplication of the conditional probabilities of individual term occurrence variables d_t:

$$\begin{aligned} P(R=1|d,q) &\overset{rank}{=} \prod_{t\in V}\frac{P(d_t|R=1,q)}{P(d_t|R=0,q)} \\ &= \prod_{t\in V:d_t=1}\frac{P(d_t=1|R=1,q)}{P(d_t=1|R=0,q)}\times\prod_{t\in V:d_t=0}\frac{P(d_t=0|R=1,q)}{P(d_t=0|R=0,q)} \end{aligned} \tag{2.3}$$

For simplicity, define p_t to be the probability that d contains t given d is relevant and u_t to be the probability that d contains t given d is non-relevant:

$$p_t = P(d_t=1|R=1,q) \tag{2.4}$$

$$u_t = P(d_t=1|R=0,q) \tag{2.5}$$

Further assume that terms that do not occur in the query q are equally likely to occur in relevant and non-relevant documents, (i.e., $p_t = u_t$ if $q_t = 0$). The ranking function becomes:

$$
\begin{aligned}
P(R=1\mid d,q) &\overset{rank}{=} \prod_{t\in V: d_t=q_t=1} \frac{p_t}{u_t} \times \prod_{t\in V: d_t=0, q_t=1} \frac{1-p_t}{1-u_t} \\
&= \prod_{t\in V: d_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} \times \prod_{t\in V: q_t=1} \frac{1-p_t}{1-u_t} \\
&\overset{rank}{=} \prod_{t\in V: d_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} \\
&\overset{rank}{=} \sum_{t\in V: d_t=q_t=1} \log \frac{p_t(1-u_t)}{u_t(1-p_t)} \\
&= \sum_{t\in V: d_t=q_t=1} w_t
\end{aligned} \tag{2.6}
$$

where

$$
w_t = \log \frac{p_t(1-u_t)}{u_t(1-p_t)} \tag{2.7}
$$

and the remaining concern is to estimate p_t and u_t.

With some or full relevance information, p_t and u_t can be estimated by counting the number of documents containing term t in the sets of relevant and non-relevant documents:

$$
p_t = \frac{r_t}{R} \tag{2.8}
$$

$$
u_t = \frac{df_t - r_t}{D - R} \tag{2.9}
$$

where r_t is the number of relevant documents containing the term t, R is the number of relevant documents for a given query q, df_t is the number of documents containing the term t and D is the total number of documents in the collection. Putting Equations 2.7, 2.8 and 2.9 together:

$$
w_t = \log \frac{\left(\frac{r_t}{R-r_t}\right)}{\left(\frac{df_t - r_t}{D - R - df_t + r_t}\right)} \tag{2.10}
$$

In order to avoid zero values which would cause undefined result in the calculation, 0.5 is added to each of the quantities in Equation 2.10 for smoothing:

$$w_t = \log \frac{\left(\frac{r_t + 0.5}{R - r_t + 0.5} \right)}{\left(\frac{df_t - r_t + 0.5}{D - R - df_t + r_t + 0.5} \right)} \tag{2.11}$$

In the presence of relevance information, the above quantity is called the w_4 weight in the literature [Robertson and Sparck Jones, 1976]. Obviously, the w_4 weight measures the importance of a term in a document without considering the position of the term in the document. In [Wu et al., 2005], a document-context model which incorporates the positional information of the terms for measuring importance of terms is compared with the w_4 weight in a retrospective experiment, that is, full relevance information is presence to the retrieval models. The results show that using document-context in calculating the term weights can improve retrieval effectiveness.

In practice, relevance information is difficult to obtain. Without any relevance information, it is assumed that for a given query q, the number of relevant documents is very small when compared to the total number of documents in the whole collection. In other words, a very large percentage of documents in the collection are non-relevant. This assumption is valid for large collections. As a result, r_t and R are set to zero (i.e., $r_t = R = 0$) in Equation 2.11 and w_t becomes:

$$w_t = \log \frac{D - df_t + 0.5}{df_t + 0.5} \tag{2.12}$$

The above quantity is the well-known Inverse Document Frequency (IDF) which measures the importance of a term in a collection without any relevance information. When there are a lot of documents containing the term t, df_t is large and hence w_t is small which means that the term t is of

less importance. Stop words such as prepositions are examples of these less important terms. These terms occur in almost every document in the collection so they do not have much discriminative power when they appear in a query. On the other hand, if only a limited number of documents containing the term t, df_t is small and hence w_t is large which means the term t is of high importance to identify the documents containing term t. The IDF weight has been used in many information retrieval systems since its introduction.

In Chapter 3 (A Retrospective Study of a Hybrid Document-context Based Retrieval Model), we develop a hybrid document-context model based on the BIR model. Instead of considering the probability of relevance of a document, $P(R=1 \mid d, q)$, we try to calculate the probability of relevance of a context in a document, $P(R=1 \mid c(d, k), q)$ where $c(d, k)$ is the context at the k-th position in the document d. As a result, a term weight which depends on the document-context is proposed.

2.1.2 The TF-IDF Term Weight

In [Luhn, 1958], term frequency was introduced as an indicator for the significance of a term t in a document. Intuitively, the higher the term frequency of t in d, the more the importance of t in d. In the BIR model, documents are represented by the set of term occurrence variables which only reveals the presence or absence of terms (i.e., $d_t \in \{0,1\}$), information such as term frequencies is lost in the BIR modeling. In order to overcome this problem, the 2-Poisson model was developed [Harter 1974, 1975a, 1975b; Bookstein and Swanson, 1974; Robertson, Van Rijsbergen and Porter, 1981]. In the 2-Poisson model, queries and documents are represented by the set of occurrence variables which are natural numbers (i.e., $d_t \in \mathbb{N}$) showing the term frequencies of each term occurring in a query or document, (i.e., $q \in \mathbb{N}^{|V|}$ and $d \in \mathbb{N}^{|V|}$). This differs from the BIR model which uses binary occurrence variables.

Given a query q and a document d, define $p_{t,f}$ to be the probability that the term t occurs f times in the document d given that d is relevant to q, and $u_{t,f}$ to be the probability that the term t occurs f times in the document d given that d is non-relevant to q:

$$p_{t,f} = P(d_t = f \mid R = 1, q) \tag{2.13}$$

$$u_{t,f} = P(d_t = f \mid R = 0, q) \tag{2.14}$$

Using $p_{t,f}$ and $u_{t,f}$, the ranking formula of the 2-Poisson model is derived analogously to Equation 2.6:

$$\begin{aligned} P(R=1 \mid d, q) &\overset{rank}{=} \prod_{t \in V: d_t > 0, q_t > 0} \frac{p_{t,f}}{u_{t,f}} \times \prod_{t \in V: d_t = 0, q_t > 0} \frac{p_{t,0}}{u_{t,0}} \\ &= \prod_{t \in V: d_t > 0, q_t > 0} \frac{p_{t,f} \times u_{t,0}}{u_{t,f} \times p_{t,0}} \times \prod_{t \in V: q_t > 0} \frac{p_{t,0}}{u_{t,0}} \\ &\overset{rank}{=} \prod_{t \in V: d_t > 0, q_t > 0} \frac{p_{t,f} \times u_{t,0}}{u_{t,f} \times p_{t,0}} \\ &= \prod_{t \in V: d_t > 0, q_t > 0} w_{t,2-Poisson} \end{aligned} \tag{2.15}$$

and

$$w_{t,2-Poisson} = \frac{p_{t,f} \times u_{t,0}}{u_{t,f} \times p_{t,0}} \tag{2.16}$$

To estimate $p_{t,f}$ and $u_{t,f}$, the 2-Poisson model uses a mixture of two Poisson distributions, one from an elite source and the other one from a non-elite source. Eliteness is a hidden variable shows whether a term t is "about" a document d and it is difficult to define in practice. Therefore, the 2-Poisson model faces a problem of difficult parameter estimation.

In 1994, Robertson and Walker presented some simple and effective approximations to the 2-Poisson model [Robertson and Walker, 1994] in the form of TF-IDF which is the multiplication of the term frequency (TF)

factor and the inverse document frequency (IDF) factor. This is called the BM25 model in the literature and the ranking formula is:

$$P(R=1\mid d,q) \overset{rank}{=} \sum_{t\in d\cap q} tf_{t,q} \frac{tf_{t,d}\times(k_1+1)}{tf_{t,d}+k_1\times\left(1-b+b\times\frac{|d|}{\Delta}\right)} \log\frac{D-df_t+0.5}{df_t+0.5} \quad (2.17)$$

where $tf_{t,q}$ is the occurrence frequency of the term t in query q, $tf_{t,d}$ is the occurrence frequency of the term t in document d, $|d|$ is the length of the document d, Δ is the average document length in the collection, D is the number of documents in the collection, df_t is the number of documents containing term t, finally, k_1 and b are model parameters.

While the BM25 model is simple to implement, it continues to achieve state-of-the-art retrieval effectiveness. Besides the BM25 model, there are other retrieval models which combine the variants of the TF and IDF components. Salton and Buckley [1988] identified three main components for effective retrieval, they are (1) the term frequency (TF) factor, (2) the inverse document frequency (IDF) factor, and (3) the document length factor. The document length factor is used to normalize the term frequency factor in most cases, this is to avoid the bias to long documents since longer documents tend to contain more query terms. Note that the three components can take different forms in different retrieval models, while the BM25 model being one of them. In Chapter 4 (Interpreting TF-IDF term weights as making relevance decision), we show that when we shrink the context size to unity, our document-context model can be interpreted as using TF-IDF term weights which is similar to the various retrieval models in the literature including the BM25 model and the variants presented in [Salton and Buckley, 1988].

2.2 Models with Document-Context

The probabilistic retrieval models described in the previous section are built with a strong assumption that the attributes (terms) describing the

documents are independent to each other. Generally, a document is treated as bag-of-terms which means the terms exist independently. In reality, this kind of bag-of-terms modelling is obviously over simplified since terms are inter-related with each other. As a result, there are attempts to overcome the limitation of the traditional bag-of-terms models by modelling the relationship among terms in a document. Some of the work in the literature use document-context similar to ours. In this section, we briefly describe these works.

In 2000, Vechtomova and Robertson [2000] presented a method of combining corpus-derived data on word co-occurrences using collocations with the traditional probabilistic model of information retrieval. Significant collocates are selected using a window-based technique around the interested (node) terms. Given a query q, for every query term $t \in q$, top collocates for the term t are selected using collection statistics, which are mutual information (MI) and Z statistic specifically. After extracting the top collocates for every query term, the query is expanded using the extracted collocates.

Standard mutual information score between a pair of terms measures the mutual dependence of the two terms. If two terms always co-occur with each other, they have a high mutual information score. On the other hand, if two terms co-occur mainly due to chance, their mutual information score will be close to zero. Below shows the formula for calculating standard mutual information score of two terms x and y:

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{2.18}$$

where $P(x, y)$ is the probability that the terms x and y occur together, $P(x)$ and $P(y)$ are the probabilities that the terms x and y occur individually. In order to allow for terms co-occur within a distance (i.e., within a window of terms), a modified MI score was used in [Vechtomova and Robertson, 2000]:

$$I_w(x,y) = \log_2 \frac{P_w(x,y)}{P(x)P(y)} = \log_2 \frac{\left(\frac{f(x,y)}{L \times w_x}\right)}{\left(\frac{f(x)f(y)}{L^2}\right)} \qquad (2.19)$$

where $P_w(x, y)$ is the probability of occurrence of y in the windows around x, $f(x, y)$ is the frequency of co-occurrence of x and y, $f(x)$ and $f(y)$ are the individual frequencies of occurrence of x and y respectively, L is the collection length which is the sum of document lengths for all documents in the collection and w_x is the average window size for x. The window size used was 201 with 100 terms for each side of the centre term. Besides MI, Z statistic was also used to measure the degree of confidence that the pair of terms x and y are associated:

$$Z(x,y) = \frac{f(x,y) - \frac{w_x f(x) f(y)}{L}}{\sqrt{\frac{w_x f(x) f(y)}{L}}} \qquad (2.20)$$

where $f(x, y)$, $f(x)$, $f(y)$, w_x and L are defined the same as those in Equation 2.19 when computing MI.

The notion of context in our proposed model is very similar to the window-based technique used by Vechtomova and Robertson However, unlike [Vechtomova and Robertson, 2000], our model does not extract collocates for query expansion. The statistics may not be reliable when the frequency of occurrence of the term is low. Therefore, undesired terms which will degrade retrieval effectiveness may be extracted using the statistics. Instead, we consider all possible occurrences of terms in the context and use log-odds to rank them. Moreover, we derive our model using the notion of document-context at the very beginning (see Chapter 3 (A Retrospective Study of a Hybrid Document-context Based Retrieval Model)).

In 2005, Metzler and Croft [2005] developed a novel retrieval model based on Markov random field (MRF). Their model assigns different weights to

different types of query term occurrence patterns in the documents. Specifically, the MRF approach models the joint distribution $P_\Lambda(q, d)$ over queries and documents by three classes of lexical features, they are (1) individual terms, (2) contiguous phrases, and (3) terms with proximity. It is the third class (terms with proximity) which document-context is employed. This class of lexical features models a pair of query terms occur in a document within a distance. The MRF model considers the mixture of the three classes of lexical features:

$$P_\Lambda(d,q) \propto \lambda_T f_T + \lambda_O f_O + \lambda_U f_U \qquad (2.21)$$

where f_T is the feature function for individual terms, f_O is the feature function for contiguous phrases, f_U is the feature function for proximity, and λ_T, λ_O and λ_U are the corresponding weights such that $\lambda_T + \lambda_O + \lambda_U = 1$. Note that when $\lambda_T = 1$ and $\lambda_O = \lambda_U = 0$, the MRF model is equivalent to the query likelihood model [Ponte and Croft, 1998] which ranks the documents according to the probability of generating the query q by the language model of document d (i.e., $P(q| d)$).

In the MRF model the scores of different types of query term occurrence patterns are combined together as the document score for ranking. Although Metzler and Croft [2005] used document- contexts, their model did not use the non-query terms in the contexts for document ranking. Since only query terms are considered, the proximity is within a small distance (less than 10 terms) when compared to the context sizes used in our document-context model (51-101 terms). Also, their model was not motivated by modeling or simulating relevance decision making while we do that in Chapter 4 (Interpreting TF-IDF term weights as making relevance decision).

The MRF model [Metzler and Croft, 2005] and its modified version [Lease, 2008] have shown to be very effective in text retrieval at TREC [Metzler et al., 2005; Lease, 2008]. In Chapter 5 (Probabilistic Document-Context Based Retrieval Model), we compare results from our document-context

model with those from the MRF model as in [Lease, 2008] in a relevance feedback environment.

In 2006, Pickens and MacFlane [2006] proposed the term context model for information retrieval. For a query q, instead of just looking at the occurrence of the query terms $t \in q$ in a document, the term context model considers a set of non-independent supporting terms which was said to be the "context" of a query term. That is relationships among context terms are also considered. The model makes use of the maximum-entropy framework to compute the relationships among terms. They have shown that related terms found using the term context model are different from those found using co-occurrence statistics.

Other work has also been done on exploiting word co-occurrence statistics using windowing techniques. For examples, Lv and Zhai [2009] proposed the positional language model (PLM) which defines a language model for each position of a document. The PLM is estimated based on propagated counts of words within a document through a proximity-based density function, which captures both proximity heuristics and achieves an effect of "soft" passage retrieval. De Kretser and Moffat [1999] introduced the shape, height and spread factors to measure the influence of query terms at difference locations. Vechtomova et al. [2006] empirically investigated whether the degree of lexical cohesion between the contexts of query terms' occurrences in a document is related to its relevance to the query. By contrast, we concentrate on individual contexts of the query terms in a document to test whether a particular context is relevant. Xu and Croft [2000] implicitly assumed that query terms and expansion terms are related within some context windows using the local context analysis (LCA) in the local collection (i.e., the top ranked documents). However, we do not perform query expansion but utilize the term distributions in relevant and irrelevant documents. In [Lund and Burgess, 1996; Burgess and Lund, 1997; Burgess et al., 1998], the researchers developed the Hyperspace Analogue to Language (HAL) model to automatically construct the dependencies of a

term with other terms using their co-occurrences [Bruza and Song, 2003] inside a context in a sufficiently large corpus. For a given term, a vector is created which elements are the probabilities of the term co-occurring with other terms. Song and Bruza [2003] proposed an information inference mechanism in information retrieval for making inferences via computations of information flow in a high dimensional context-sensitive vector space constructed using the HAL model. Gao et al. [2004] extended the language modeling approach by incorporating dependencies between terms in the model using term co-occurrence statistics which showed that using the co-occurrence information in language modeling benefits the retrieval effectiveness. Bai et al. [2005] proposed the context-dependent query expansion technique in language modeling approach using extended inference model with information flow. All the studies provided evidence that exploiting the term co-occurrence information is crucial for increasing retrieval effectiveness. In our model, by considering the document-contexts of the query terms in documents, it simulates relevance-decision making for the document-contexts. This extends the usage of term co-occurrence information to match the conceptual meaning of the query terms and document terms.

Our document-context model simulates “local” relevance-decision making for every term location in a document. We believe that term locations play an important role in determining relevance of documents to queries. There are works which do not explicitly take into account term locations in a document even though term locations have been acknowledged as an important component in determining relevance. For instance, passage retrieval [Kaszkiel et al., 1999; Liu and Croft, 2002] implicitly assumes that the influence of the query term is limited within a passage and local context analysis [Xu and Croft, 2000] implicitly assumes that query terms and expansion terms are related within some context windows. Language models [Ponte and Croft, 1998] used locations to define location frequencies of term occurrences [Roelleke and Wang, 2006], but they have not used locations in a more elaborate manner than frequency counting. The question-and-answering (QA) tasks explicitly requested the retrieved results

to include term locations but many retrieval models for QA tasks are extensions of existing retrieval models without explicit consideration of term locations in the model. Instead of adding term locations in the retrieval model as a post-processing module, we develop our probabilistic retrieval model with term locations at the beginning. The local relevance at a certain location is thought to depend on the document-context at that location.

2.3 Passage-based Retrieval Models

In passage-based retrieval, e.g., [Salton et al., 1993; Callan, 1994; Kaszkiel et al., 1999; Liu and Croft, 2002], a document is divided into passages and each of the passages is evaluated individually by the retrieval model. Xi et al. [2001] investigated a window-based passage retrieval technique but they did not require the centre term of a window to be a query term. The passage-based retrieval implicitly agrees that the query term is related to other terms in limited distances but not in large distances. Our model shares the same intuition. However, passage-based retrieval divides a document exhaustively without regard to query term locations (e.g., number of terms, passage tags). By contrast, our model divides a document based on the occurrences of query terms in the document (i.e., terms around query term).

A main concern in passage-based retrieval is how to combine the passage scores to from the final document score, since the final ranked list produced from a retrieval system usually consists of documents rather than passages. Some models use the maximum passage score to be the document score while others may average the passage scores. Our document-context model also has the same concern as we divide a document into document-contexts. Kong et al. [2004] showed that different relevance decision principles (namely the Disjunctive Relevance Decision (DRD) principle, the Aggregate Relevance Decision (ARD) principle and the Conjunctive Relevance Decision (CRD) principle) can be applied to passage-based retrieval in different scenarios when simulating the human user in making relevance decisions:

(a) the Disjunctive Relevance Decision (DRD) principle which states that a document is relevant to a topic if a particular document part is relevant to the topic;

(b) the Aggregate Relevance Decision (ARD) principle which states that a document is more relevant to a topic if more occurrences of the concepts related to the topic are found in the document; and

(c) the Conjunctive Relevance Decision (CRD) principle which states that a document is relevant to a topic if all the document parts are relevant to the topic.

In Chapter 3 (A Retrospective Study of a Hybrid Document-context Based Retrieval Model), we examine different methods of combining the context scores in our document-context model following the relevance decision principles.

2.4 Relevance Feedback in Information Retrieval

Relevance feedback (RF) can be used to enhance retrieval effectiveness [Rocchio, 1971; Salton and Buckley, 1990; Harman, 1992]. In this thesis, most experiments are performed in a relevance feedback environment. That is some or full relevance information is presence to the retrieval model. In a standard relevance feedback procedure, an initial ranked list of documents is produced by the retrieval model using no relevance information, then the user scan through the ranked list from the top ranked document and provides feedback for relevance of documents to the retrieval model. After a certain number of top ranked documents are judged by the user, a second ranked list of documents is produced using the relevance information obtained from the user.

In some cases, the top ranked documents are very similar to each other or even identical. Judging the relevance of the nearly identical documents would waste user efforts and provide no additional useful information to the retrieval model. Therefore, the set of documents used for relevance feedback

may not be the top ranked ones. This is called active feedback [Shen and Zhai, 2005] in which the retrieval system actively chooses suitable documents for the user to judge for relevance. In Chapter 6 (A Split-List Approach for Relevance Feedback in Information Retrieval), we propose a new active feedback algorithm which uses document-contexts and compare it with another active feedback algorithm, namely the maximal marginal relevance (MMR) algorithm [Carbonell and Goldstein, 1998].

Relevance feedback serves two main purposes, they are (1) query expansion and (2) parameters estimation. Rocchio [1971] was the first to formulate query expansion using relevance feedback, it was implemented in the Vector Space Model (VSM) which queries and documents are modelled as $|V|$-dimensional vectors where $|V|$ is the size of the vocabulary V. In Rocchio's formulation, all terms from the judged relevant and non-relevant documents are added to the original query using:

$$\vec{q}_{rf} = \vec{q} + \frac{1}{|R|}\sum_{\vec{d}\in d_{REL,q}} \frac{\vec{d}}{|\vec{d}|_1} - \frac{1}{|I|}\sum_{\vec{d}\in d_{IRL,q}} \frac{\vec{d}}{|\vec{d}|_1} \qquad (2.22)$$

where $\vec{q}$ is the original query vector, $\vec{q}_{rf}$ is the expanded query vector, $d_{REL,q}$ is the set of judged documents relevant to q, $d_{IRL,q}$ is the set of judged documents non-relevant to q and $|\vec{d}|_1$ is the city-block length of the document vector $\vec{d}$. Later, Harman [1992] showed that retrieval effectiveness can be enhanced by selecting terms for expansion rather than using all the terms in the judged documents.

Besides query expansion, model parameters such as b and k_1 of the BM25 model in Equation 2.17 can be estimated more accurately for better performance with the help of relevance information.

2.4.1 Evaluation in Relevance Feedback

The mean average precision (MAP) is the most commonly used measure to evaluate the performance of a retrieval model. It is the arithmetic mean of the average precision (AP) across the set of all tested queries *Q*:

$$MAP(Q) = \frac{1}{|Q|}\sum_{q \in Q} AP(n,q) \tag{2.23}$$

where *n* is the number of documents returned by the retrieval model for a particular query *q*. In TREC, usually, top 1000 documents are considered (i.e., *n* = 1000). The average precision *AP*(*n*, *q*) is the average of the precisions at the point of each relevant documents in the ranked list:

$$AP(n,q) = \frac{1}{R}\sum_{r=1}^{n} Prec(r,q) \times Rel(r,q) \tag{2.24}$$

where *R* is the number of relevant documents for *q* and,

$$Prec(r,q) = \frac{\text{number of relevant documents ranked} \le r}{r} \tag{2.25}$$

$$Rel(r,q) = \begin{cases} 1 & \text{if the document ranked at } r \text{ is relevant to } q \\ 0 & \text{if the document ranked at } r \text{ is non- relevant to } q \end{cases} \tag{2.26}$$

In relevance feedback environment, a question to ask is whether we should include the judged documents in the final ranked list for computing the MAP. In 1971, Chang et al. [1971] studied various evaluation methods of relevance feedback including modified rank freezing, residual collection and test and control group.

Rank freezing refers to assigning the ranks of judged documents in the final ranked list according to the order of the documents being judged. For example, if the top 20 documents in the initial ranked list are judged by the user, the ranks of those 20 documents are frozen in the final ranked list such that they are the same as those in the initial ranked list. Therefore, the

retrieval model can only utilize the relevance feedback information by changing the ranks of documents from rank number 21 onwards. That is $Prec(r, q)$ (Equation 2.25) and $Rel(r, q)$ (Equation 2.26) will be the same in the initial and final ranked list for $r \leq 20$. In modified freezing [Chang et al., 1971], instead of freezing all the ranks of judged documents (relevant and non-relevant), the ranks of judged non-relevant documents ranked below the last relevant one are not frozen. In [Chang et al., 1971], it showed that modified freezing may be better for individual query comparisons.

The residual collection [Ide, 1969] evaluation attempts to measure the effectiveness of relevance feedback by the number of *newly* retrieved relevant documents. The judged documents during relevance feedback are discarded from the final ranked list. Note that when evaluating the final ranked list using residual collection, the number of relevant documents R (Equation 2.24) should be changed accordingly by eliminating the judged relevant documents during relevance feedback. In Chapter 5 (Probabilistic Document-Context Based Retrieval Model), we use residual collection to evaluate our document-context model for relevance feedback.

In the test and control method [Chang et al., 1971], a given document collection is split into two halves randomly. One half is used for performing the initial retrieval and outputting the initial ranked list for feedback (i.e., the test group). The other half is used for the final retrieval and performing evaluation (i.e., the control group).

2.4.2 Relevance Feedback in Practice

In practice, relevance information is difficult to obtain because users are unwilling to make relevance judgements. Therefore, methods are introduced to perform relevance feedback without direct user involvement including pseudo-relevance feedback (PRF) [Croft and Harper, 1979; Buckley, 1995] and implicit feedback [Joachimes et al., 2005]. In pseudo-relevance feedback, top ranked documents are assumed to be relevant and contain

useful terms for query expansion. The terms in the top ranked documents are weighted using a certain metric (e.g., TF-IDF term weights) and terms with highest weights are selected to combine with the original query. The selected terms are called PRF terms. Usually, a mixture parameter is used to control the weight of original query terms and the PRF terms in order to avoid query drift which can decrease retrieval effectiveness.

In web search, implicit feedback [Joachimes et al., 2005] can be done by analyzing the clickthrough data of users instead of directly asking them for giving feedback. When a user clicks on a document (web page) link, it is interpreted as an endorsement to the document relating to the query. Therefore there is a higher chance of the document being relevant. On the other hand, if a user bypasses a document link, there is a higher chance of the document being non-relevant. Since web search engines keep a huge amount of clickthrough data from all users, implicit feedback may be feasible in web search environment.

Chapter 3

A Retrospective Study of a Hybrid Document-Context Based Retrieval Model

This chapter describes our novel retrieval model that is based on contexts of query terms in documents (i.e., document contexts). Our model is novel because it explicitly takes into account of the document contexts instead of implicitly using the document contexts to find query expansion terms. Our model is based on simulating a user making relevance decisions, and it is a hybrid of various existing effective models and techniques. It estimates the relevance decision preference of a document context as the log-odds and uses smoothing techniques as found in language models to solve the problem of zero probabilities. It combines these estimated preferences of document contexts using different types of aggregation operators that comply with different relevance decision principles (e.g., aggregate relevance principle). Our model is evaluated using retrospective experiments (i.e., with full relevance information), because such experiments can (a) reveal the potential of our model, (b) isolate the problems of the model from those of the parameter estimation, (c) provide information about the major factors affecting the retrieval effectiveness of the model, and (d) show that whether the model obeys the probability ranking principle. Our model is promising as its mean average precision is 60%- 80% in our retrospective experiments using different TREC ad hoc English collections and the NTCIR-5 ad hoc Chinese collection. Our experiments showed that (a) the operators that are consistent with aggregate relevance principle were effective in combining the estimated preferences, and (b) that estimating probabilities using the contexts in the relevant documents can produce better retrieval effectiveness than using the entire relevant documents.

3.1 Introduction

Various retrieval models have been developed and investigated over the past several decades based on a variety of mathematical frameworks [Dominich, 2000]. For example, Salton et al. [1975] and Wong et al. [1985] worked on retrieval models based on vector spaces. The Binary Independence Retrieval (BIR) model [Robertson and Sparck-Jones, 1976], the logistic regression model [Cooper et al. 1992], the 2-Poisson model [Harter, 1975] and its later practical approximation [Robertson and Walker, 1994] and the language modelling approach [Ponte and Croft, 1998; Lafferty and Zhai, 2001; Lavrenko and Croft, 2001] are based on the probability theory. The fuzzy retrieval model [e.g., Miyamoto, 1990] and the extended Boolean model [Salton et al., 1983] are based on the fuzzy set theory [Zadeh, 1965]. These models provide a system point of view of how to retrieve documents that are sufficiently relevant that they satisfy a user's information need. On the other hand, an information retrieval system can be thought of as simulating the human user when making relevance decisions in the retrieval process [Bollmann and Wong, 1987]. In this case, the ranking of the relevance of the documents to the user's information need is in terms of preferences [Yao and Wong, 1991].

In this work, we simulate human relevance decision making in the development of a novel retrieval model that explicitly models a human relevance decision at each location in a document. The relevance decision at the specified location in the document is based on the context at that location so that the relevance decision preference (or context score) at the specified location is estimated using the context at that location. Although using contexts in documents to explore term co-occurrence relationships for query expansion is not new, to the best of our knowledge, it is new to model the contexts/windows features explicitly in the retrieval model by incorporating the locations of terms inside a document for re-weighting the query terms. By re-weighting the query terms using the contexts of the

query terms in documents, the model assigns context dependent term weights which are aggregated together as the final document similarity score.

A document-context is essentially a concordance or a keyword in context (KWIC) [Kupiec et al., 1995]. Figure 1.1 shows some example document contexts containing a query term in the title query, “Hubble Telescope Achievements”. The contexts were extracted from a raw (un-processed) document. During retrieval, unlike Figure 1.1, all the terms are stemmed and the stop words are removed. From Figure 1.1, it should be noted that even for a relevant document, not all contexts in the document are relevant.

Our model uses current successful retrieval models and techniques to estimate the relevance decision preferences (or context scores) of document contexts containing a query term in the center. The relevance decision preferences are defined as the log-odds estimated using smoothing techniques and they are combined using aggregation operators. More specifically, we used the technique of smoothing [Chen and Goodman, 1996; Zhai and Lafferty, 2004] to solve the problem of zero probabilities [Ponte and Croft, 1998] in estimating the term distributions in relevant documents similar to the language models [Ponte and Croft, 1998; Lafferty and Zhai, 2001; Lavrenko and Croft, 2001]. We calculated the probability of the relevance of a particular document context similar to that of the BIR model [Robertson and Sparck-Jones, 1976]. In order to calculate the document score for ranking, the document-context log-odds are combined using different evidence aggregation operators based on the extended Boolean model [Salton et al., 1983] and some fuzzy (aggregation) operators [Dombi, 1982; Yager, 1988; Paice, 1984]. Therefore, our proposed retrieval model is a *hybrid* of various past successful retrieval models and techniques.

In predictive experiments, a major source of difficulty in developing novel retrieval models is in determining whether the effectiveness performance is limited by the underlying model or by the poor parameter estimation techniques used. Instead of predictive experiments, we propose to evaluate our novel retrieval model based on retrospective experiments that are

performed using relevance information (e.g., the TREC relevance judgments), similar to the retrospective experiments in [Robertson and Sparck-Jones, 1976; Sparck-Jones et al., 2000; Hiemstra and Robertson, 2001]. The purpose of the retrospective experiments is to:

(a) evaluate the potential of the underlying novel retrieval model by observing the best effectiveness that can be attained by the model;

(b) reveal the (near) optimal performance of the model and provide a yardstick for future (predictive) experiments. In the probability ranking principle [Robertson, 1977], full relevance information can enable the model obtain optimal performance [Hiemstra and Robertson, 2001];

(c) focus on gathering crucial factors (e.g., the size of the context) affecting the performance of the model when using the context of query terms in a document. We gather statistical data on these factors for analysing and designing the model to operate in predictive experiments;

(d) show whether the model obeys the probability ranking principle [Robertson, 1977]; and

(e) examine the relevance decision principles in [Kong et al., 2004] and determine which is the most suitable in simulating the human user when making relevance decisions.

The problem of estimating parameters with limited or no relevance information is left for future work since it is not known whether the proposed model is worth further investigation. When considering the terms in relevant documents, we discard those terms with document frequency equals to one. This avoids finding identifiers (e.g., document id) that uniquely identify relevant documents, thereby guaranteeing to obtain high precision when the relevance information is present. By contrast, we would like to utilize the term distributions in relevant and irrelevant documents for retrieval.

We emphasize that our document-context based model is a *descriptive* model in this chapter even though it could become a normative model. A

descriptive model describes how the decision is made while a normative model specifies how the (optimal) decision should be made. Our document-context based model is descriptive in this chapter because it does not feedback any effectiveness performance information (e.g., MAP) to the system for performance optimization (e.g., query optimization [Buckley and Harman, 2003] or model parameter optimization). For instance, our retrieval model directly estimates the probabilities without any effectiveness feedback about the estimation being good or not for document ranking. Also, the retrieval process of our model is a one-pass re-ranking process using the proposed ranking formula (discussed in details in Section 3.2) that describes how the relevance decision is made.

One may argue that if we know the relevance information, then the retrieval effectiveness performance must be good and it is pointless to do the experiments. However, as mentioned above, we are not finding identifiers of relevant documents (terms with document frequency equals to one are ignored). The descriptive model does not optimize the query or the model parameters using effectiveness performance results from previous runs. Moreover, the retrieval performance is not guaranteed to be good even when we know the relevance information. (e.g., in [Hiemstra and Robertson, 2001], the performances of the retrospective experiments are similar to those in the predictive experiments [Robertson and Walk, 1999]). This is because the terms in the relevant documents may also appear in the irrelevant documents. By using the relevance information, we are not manipulating or restricting the term distributions/occurrences in the documents but using existing probabilistic methods to estimate the term distributions/occurrences in the documents. Furthermore, we tested our model with different document collections (TREC-2, TREC-6, TREC-7, TREC-2005 and NTCIR-5) to show that the model is reliable. Finally, doing the retrospective experiments also provides us with an important clue about the potential of the retrieval model because an applicable model should perform well in the presence of relevance information. The use of relevance information can reveal the (near) optimal performance and the estimation of

the relevance information is possible using various techniques such as pseudo relevance feedback.

The index structure used in our experiments does not contain the positional information of terms in a document, as a result we have to access the disk to read the document and extract the contexts of query terms. This will greatly increase the time needed for the experiments. The problem can be eased by including the positional information of terms in the index structure similar to the index used in the Indri retrieval system [Strohman et al., 2004]. However, including the positional information of term in the index structure will increase the storage requirement of the index. In this thesis, we do not examine the time-efficiency of our retrieval model or retrieval system because:

(a) it is already very challenging to design and develop a highly effective retrieval model;

(b) once the effective retrieval model is developed, then we have enough information to design and develop (novel) index structures to support such an effective model;

(c) the time-efficiency problem may reduce its significance in time as computers are continually becoming more and more powerful.

We leave how to make our retrieval model more time-efficient to our future investigation.

The rest of the chapter is organized as follows. Section 3.2 presents the details of our hybrid document-context based retrieval model. Section 3.3 shows the results of the model-oriented experiments which test the model extensively using one data collection, TREC-6. Section 3.4 shows the results of the scope-oriented experiments which test the model across different data collections and with another language. Section 3.5 concludes the chapter.

3.2 Document-Context Based Retrieval Model

In this section, we introduce our document-context based retrieval model that ranks documents on the basis of the contexts of the query terms in documents (i.e., document contexts). A document-context is uniquely identified by the location where the query term occurs in the document. Therefore, assigning different term weights to the same query term in different contexts can be thought of as assigning different term weights to the same query term in different locations in the document. Hence, we can explicitly incorporate the (query) term locations in a document in our retrieval model as reflecting the relevance of the corresponding contexts to the query. We believe that the term distributions of the contexts are similar for query terms having the same meaning, while the term distributions of the contexts are different when the same query term refers to different meanings in different contexts in documents. By incorporating the document-context information for weighting the query terms, we are trying to solve the problem of polysemy (i.e., a term with multiple meanings) in natural language because the meaning of terms without contexts can be ambiguous while terms with contexts should have definite meanings.

Given that each context has a score reflecting its relevance to a particular topic, some methods or bases are needed to combine the scores in a principle manner. Kong et al. [2004] showed that different relevance decision principles (namely the *Disjunctive Relevance Decision* (DRD) principle, the *Aggregate Relevance Decision* (ARD) principle and the *Conjunctive Relevance Decision* (CRD) principle) can be applied to passage-based retrieval in different scenarios when simulating the human user in making relevance decisions. In this chapter, we extend their work by applying the relevance decision principles to guide the selection of aggregation operators to combine the context scores of query terms in a document (e.g., Figure 1.1), instead of passages.

Our model allows the probability of making relevance decision at each location in a document to be different. Our initial study [Wu et al., 2005] showed that the model could achieve high retrieval effectiveness (i.e., about 36% mean average precision in TREC-6 using retrospective experiments). In here, we further improve our model in order to investigate whether our retrieval model can be more effective than before.

The rest of this section is divided into four parts. First, we define the document context. Second, we develop the context score that reflects the relevance preference of the context to a given topic. Third, we discuss different techniques to solve the zero probability problems. Finally, we describe various context score combination methods that are consistent with different relevance decision principles [Kong et al., 2004]. For convenience, the symbols used in the rest of this chapter are shown in Table 3.1.

Table 3.1: Symbols used in this chapter and their descriptions.

Symbol	Description
t	A particular term
D	The collection of documents
$card(D)$	The cardinality of D (i.e., the number of documents in D)
$\|D\|$	The collection length (i.e., the sum of all document lengths in D)
d_i	The i-th document where $i \in [1, card(D)]$
$\|d_i\|$	The length (total number of terms) of d_i
$d_i[k]$	The k-th term in the i-th document where $k \in [1, \|d_i\|]$
q	A particular query
$\|q\|$	The length (total number of terms) of q
$q[j]$	The j-th term in the query q where $j \in [1, \|q\|]$
$c(d_i, k)$	The context of the k-th term in the i-th document with size $2n+1$
$c(d_i, k)[l]$	The l-th term in $c(d_i, k)$ where $l \in [1, 2n+1]$
R	The binary random variable of relevance ($R = 1$ means relevant, $R = 0$ means irrelevant)
$M(R=1, q)$	The relevance model for q
$M(R=0, q)$	The irrelevance model for q
$M(D, q)$	The collection model for q
δ_a	The parameter in additive smoothing
δ_{jm}	The parameter in Jelinek-Mercer smoothing
δ_d	The parameter in absolute discounting
$weight(d_i, k)$	The context score at the k-th location in the i-th document
$w(d_i, k)$	The normalized context score at the k-th location in the i-th document
$sim(d_i, q)$	The similarity score of d_i for q

3.2.1 Context Definition

The context $c(d_i, k)$ of a term $d_i[k]$ appears at the k-th location in the i-th document with size $2n+1$ is defined by the set of terms surrounding and including the term inside the document (we use $c(d_i, k)$ instead of $c(d_i, k, n)$ in this chapter because we want to distinguish the context size parameter n from d_i and k which are input variables, as a result, n is specified implicitly):

$$c(d_i, k) \equiv \{d_i[k-n] , \ldots, d_i[k-1], d_i[k], d_i[k+1], \ldots, d_i[k+n]\}.$$

We are interested in the contexts that with a query term at the center position (i.e., $d_i[k] \in q$). Other contexts that have non-query terms at the center are considered irrelevant. This is equivalent to making the following assumption in our model:

Query-Centric Assumption: For a particular query q and a document d_i relevant to q, the relevant information for q locates only in the contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$. (i.e., the relevant information locates around query terms.)

The query-centric assumption states that if one can find relevant information in a document, then the relevant information must locate around the query terms inside the document. Note that the query-centric assumption does not require *all* contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$ to be relevant, and it only requires that the relevant information locates in the contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$. The query-centric assumption may be invalid because some of the relevant documents found in the TREC and NTCIR collections for some queries do not contain any of the query terms. In order to show that the query-centric assumption is not entirely unrealistic, we have to know the number of relevant documents which do not contain any query terms. Table 3.2 shows that the average proportions of relevant documents without any query terms per topic using title queries across different data collections (including the Chinese collections in NTCIR-5)

are less than 13%. Although Table 3.2 does not directly validating the query-centric assumption, the purpose of Table 3.2 is to show that the query-centric assumption is not entirely unrealistic.

Table 3.2: Statistics on No. of relevant documents without query terms.

	TREC-2	TREC-6	TREC-7	TREC-2005	NTCIR-5	
					Relax	Rigid
Topics (Title query)	101-150	301-350	351-400	50 past hard topics	001-050	
No. of relevant documents	11,645	4,611	4,674	6,561	3,052	1,885
No. of relevant documents without any stemmed query terms	644	354	634	385	202	63
Average % of relevant document without any stemmed query terms per topic	5.6%	9.7%	12.9%	7.0%	6.2%	4.7%

The context size (i.e., defined as $2n+1$) is determined empirically. It should not be too large or too small to include irrelevant information or exclude relevant information respectively. This issue is addressed in our model-oriented experiments (see Section 3.3.1).

3.2.2 Context Score

Each interested context $c(d_i, k)$ with size $2n+1$ has a query term $q[j]$ where $j \in [1, |q|]$ appears in the center of the context. The context $c(d_i, k)$ contains the set of terms $\{d_i[k+p]\}$ where the term $d_i[k+p]$ occurs at the p-th location relative to $d_i[k]$ in the i-th document, and $p \in [-n, n]$. In other words, a context $c(d_i, k)$ is the set of terms surrounding and including $d_i[k]$.

In the BIR model [Robertson and Sparck-Jones, 1976], the basic question to ask for each document and each query is:

What is the probability that this document is relevant to this query?

Although there are actually implicit assumptions behind the above basic question, we try to extend it and use it as the starting point to develop our

document-context based model. Upon defining the notion of context, we can now ask another similar question for each context in each document and each query:

What is the probability that this context in this document is relevant to this query?

By query, we actually mean the topic or the user information need in the above question. However, since such questions were framed in this way before and these are well entrenched in the literature [Sparck-Jones et al., 2000], we followed the existing formulation. For a particular query *q*, we define a binary random variable *R* for the outcome of the relevance decision. $R = 1$ means relevant to *q* and $R = 0$ means irrelevant to *q*. That is, our model is currently designed for binary relevance (although it can be extended to graded relevance later). For each context, there are two possible outcomes (events):

(a) The context is relevant, i.e., $R = 1$.

(b) The context is irrelevant, i.e., $R = 0$.

Similar to the BIR model [Robertson and Sparck-Jones, 1976], given a particular context $c(d_i, k)$, we want to calculate its probability of relevance and irrelevance by the log-odds:

$$P(R=1 \mid c(d_i,k),q) \overset{rank}{=} \log \frac{P(R=1 \mid c(d_i,k),q)}{P(R=0 \mid c(d_i,k),q)} \tag{3.1}$$

where $\overset{rank}{=}$ is the binary operator of rank equivalence as defined in [Lafferty and Zhai, 2003]. The log-odds reflects the relevance decision preference of the concerned context. Using Bayse' rule, we have:

$$\log \frac{P(R=1 \mid c(d_i,k),q)}{P(R=0 \mid c(d_i,k),q)} = \log \frac{P(c(d_i,k) \mid R=1,q)}{P(c(d_i,k) \mid R=0,q)} + \log \frac{P(R=1 \mid q)}{P(R=0 \mid q)} \tag{3.2}$$

The second term of Equation 3.2 is a constant and will be eliminated by the linear normalization [Lee, 1997] when combining the context scores of a

document (see Section 3.2.4), it can be ignored for ranking the contexts as follows:

$$\log \frac{P(R=1 \mid c(d_i,k),q)}{P(R=0 \mid c(d_i,k),q)} \overset{rank}{=} \log\left(\frac{P(c(d_i,k) \mid R=1,q)}{P(c(d_i,k) \mid R=0,q)}\right) \tag{3.3}$$

In order to calculate the above probabilities by the individual document terms found in the contexts, our model makes the same assumption as proposed in Cooper [1995]:

Linked-Dependence Assumption [Cooper, 1995]: The degree of statistical dependence between the terms in the relevant set is associated with their degree of statistical dependence in the irrelevant set.

The linked-dependence assumption (a) simplifies the mathematical calculations and (b) avoids the problem of data inconsistency pointed out by Cooper [1995] when assuming conditional independence of terms in relevant set and irrelevant set individually. Using the linked-dependence assumption, we have:

$$\begin{aligned}\frac{P(c(d_i,k),q \mid R=1)}{P(c(d_i,k),q \mid R=0)} &= \frac{\prod_{l=1}^{2n+1} P(c(d_i,k)[l] \mid R=1,q)}{\prod_{l=1}^{2n+1} P(c(d_i,k)[l] \mid R=0,q)} \\ &= \frac{\prod_{p=-n}^{n} P(d_i[k+p]=t \mid R=1,q)}{\prod_{p=-n}^{n} P(d_i[k+p]=t \mid R=0,q)}\end{aligned} \tag{3.4}$$

The question now is to calculate the probabilities:

$$P(t \mid R=1,q) \tag{3.5}$$
$$P(t \mid R=0,q) \tag{3.6}$$

where $t \in c(d_i, k)$.

In order to calculate Probabilities 3.5 and 3.6, we have to obtain the term distributions (i.e., the probability of seeing a particular term in a set of terms) in the relevant set and the irrelevant set. For a particular query q, let $M(R=1, q)$ be the relevance model defining the term distribution in the relevant set, $M(R=0, q)$ be the irrelevance model defining the term distribution in the irrelevant set and $M(D, q)$ be the collection model defining the term distribution in the collection. In general, Probabilities 3.5 and 3.6 are equal to:

$$P(t \mid R=1,q) = P_{M(R=1,q)}(t) = \frac{f(t, M(R=1,q))}{\sum_{w \in M(R=1,q)} f(w, M(R=1,q))} \quad (3.7)$$

$$P(t \mid R=0,q) = P_{M(R=0,q)}(t) = \frac{f(t, M(R=0,q))}{\sum_{w \in M(R=0,q)} f(w, M(R=0,q))} \quad (3.8)$$

where $f(t, M(R=1, q))$ is the raw frequency count of t in $M(R=1, q)$ and similarly for $f(t, M(R=0, q))$. So Equations 3.7 and 3.8 are the relative frequency estimates of $P_{M(R=1, q)}(t)$ and $P_{M(R=0, q)}(t)$ respectively.

Next, we use the collection model to substitute the irrelevance model because almost all of the documents are irrelevant for a query in a sufficiently large collection:

Collection-Irrelevance Assumption: For a sufficiently large collection and a query q, the irrelevance model $M(R=0, q)$ and the collection model $M(D, q)$ are similar to each other.

Hence,

$$P_{M(R=0,q)}(t) \approx P_{M(D,q)}(t) = \frac{f(t, M(D,q))}{\sum_{w \in M(D,q)} f(w, M(D,q))} \quad (3.9)$$

where $f(t, M(D, q))$ is the raw frequency count of t in the collection model $M(D, q)$. The validity of the collection-irrelevance assumption will be addressed in the model-oriented experiments (see Section 3.3.5).

The remaining concern goes to calculating the formula shown in Equation 3.7. There are various methods to calculate Equation 3.7 depending on the training method used where training here refers to estimating the probability distribution of the terms in the relevant set (i.e., the relevance model $M(R=1, q)$) using the training data. Training includes defining what terms should be included in the relevance model $M(R=1, q)$, and how much should each of the terms weight (i.e., what is the probability of seeing a term in the model). In this chapter, we explored two training methods (namely *document-training* and *context-training*) based on different assumptions and depending on what are the terms that should be included in the relevance model $M(R=1, q)$.

In *document-training*, we use the whole document (i.e., all terms inside the document) for training the model, based on the following assumption.

Document-Training Assumption: For a particular query q, the entire relevant document d_i is considered relevant so that the terms $d_i[k]$ for $k \in [1, |d_i|]$ are included in the relevance model $M(R=1, q)$.

The document-training assumption contradicts with our query-centric assumption for the relevant documents. However, the query-centric assumption is used in the ranking process while the document-training assumption is used in this particular training method. We provide this method to show that inconsistent assumptions in training and retrieval using the proposed model may degrade the retrieval effectiveness and the document-training method was used in our previous study [Wu et al., 2005].

Based on the document-training assumption, $f(t, M(R=1, q))$ is:

$$f(t, M(R=1, q)) = \sum_{d \in d_{REL,q}} f(t, d) \tag{3.10}$$

which is the occurrence frequency of the term t in the set of relevant documents ($d_{REL,q}$), $f(t, d)$ is the occurrence frequency of t in d. Hence, $P_{M(R=1, q)}(t)$ is the relative frequency estimate of the probability of seeing t in the relevance model $M(R=1, q)$.

In *context-training*, we use the contexts $\{c(d_i, k)\}$ inside a document d_i for $k \in [1, |d_i|]$ where $d_i[k] \in q$ (i.e., only the terms inside the contexts are included in the relevance model $M(R=1, q)$) for training the model, based on the following assumption.

Context-Training Assumption: For a particular query q, only the contexts in the relevant documents are relevant so that for a document d_i relevant to q, the terms in the contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$ are included in the relevance model $M(R=1, q)$.

The context-training assumption is consistent with the query-centric assumption but these two assumptions are different. The query-centric assumption does not assume that all contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$ in the relevant document d_i to be relevant. By contrast, the context-training assumption assumes that all contexts $\{c(d_i, k)\}$ for $k \in [1, |d_i|]$ where $d_i[k] \in q$ in the relevant document d_i are relevant.

Based on the context-training assumption, $f(t, M(R=1, q))$ is:

$$f(t, M(R=1, q)) = \sum_{d \in d_{REL,q}} \sum_{k\,:\,d[k] \in q} f(t, c(d, k)) \tag{3.11}$$

which is the occurrence frequency of the term t in the contexts $\{c(d,k)\}$ of relevant documents ($d_{REL,q}$) where $d \in d_{REL,q}$ and $d[k] \in q$, $f(t, c(d, k))$ is the occurrence frequency of t in $c(d, k)$. Hence, $P_{M(R=1, q)}(t)$ is the relative frequency estimate of the probability of seeing t in the relevance model

$M(R=1, q)$. We believe that the context-training assumption is more realistic than the document-training assumption because most of the time only a part of the document contains relevant information (e.g., Figure 1.1), especially for long documents.

3.2.3 Estimation Issue

The probability of seeing a term t in the relevance model $M(R=1, q)$ may equal to zero, if $f(t, M(R=1, q)) = 0$. This is the problem of zero probability in estimating $P_{M(R=1, q)}(t)$ similar to the language modelling approach [Ponte and Croft, 1998]. As the term t does not appear in $M(R=1, q)$ (whether a term is included in $M(R=1, q)$ depends on the training method, if t does not appear during training, it will become an unseen term during retrieval), it will be assigned a zero probability. Note that the problem of zero probability does not occur when estimating $P_{M(R=0, q)}(t)$ as we are using the collection model $M(D, q)$ to substitute the irrelevance model $M(R=0, q)$ (Equation 3.9). The collection model $M(D, q)$ contains all the terms in the collection, so unseen terms do not exist. The zero probability will set Equation 3.4 to zero and it can cause anomalies in ranking.

Smoothing [Chen and Goodman, 1996; Zhai and Lafferty, 2004] of the term distribution is a solution to the zero probability problem. The basic idea of smoothing is to adjust the term distribution so that zero probability will not assign to unseen terms. In this section, we describe different commonly used interpolation-based smoothing techniques [see Zhai and Lafferty, 2004] (namely additive smoothing, Jelinek-Mercer smoothing and absolute discounting) which we will apply them for investigating the effect of smoothing to our model.

Additive smoothing [Lidstone, 1920; Johnson, 1932; Jeffreys, 1948] adds a constant δ_a to all terms which make unseen terms to have uniform, non-zero probabilities:

$$P_{M(R=1,q)}(t) = \frac{f(t, M(R=1,q)) + \delta_a}{\sum_{w \in M(R=1,q)} f(w, M(R=1,q)) + \delta_a \left| M(R=1,q) \right|} \qquad (3.12)$$

where $|M(R=1, q)|$ is the number of unique terms in the relevance model and $\delta_a \in [0, 1]$. Laplace smoothing is a special case of additive smoothing (i.e., $\delta_a=1$). Additive smoothing is relatively simpler than the other two smoothing techniques because it does not require the information from the collection model.

The Jelinek-Mercer smoothing [Jelinek and Mercer, 1980; Zhai and Lafferty, 2004] is the linear interpolation of the relevance model $M(R=1, q)$ and the collection mode $M(D, q)$:

$$P_{M(R=1,q)}(t) = \delta_{jm} \frac{f(t, M(R=1,q))}{\sum_{w \in M(R=1,q)} f(w, M(R=1,q))} + (1 - \delta_{jm}) \frac{f(t, M(D,q))}{\sum_{w \in M(C,q)} f(w, M(D,q))} \qquad (3.13)$$

where $\delta_{jm} \in [0, 1]$ is the mixture control parameter of the interpolation. For an unseen term t, $f(t, M(R=1, q)) = 0$, the probability $P_{M(R=1, q)}(t)$ will become $(1-\delta_{jm}) \times P_{M(D, q)}(t)$ which depends on both δ_{jm} and the collection model $M(D, q)$.

In absolute discounting [Ney et al., 1994; Zhai and Lafferty, 2004], the raw counts of the seen terms are decreased by a constant:

$$P_{M(R=1,q)}(t) =$$

$$\frac{\max(f(t, M(R=1,q)) - \delta_d, 0)}{\sum_{w \in M(R=1,q)} f(w, M(R=1,q))} + \delta_d \left(\frac{\left| M(R=1,q) \right|}{\sum_{w \in M(R=1,q)} f(w, M(R=1,q))} \right) \left(\frac{f(t, M(D,q))}{\sum_{w \in M(D,q)} f(w, M(D,q))} \right) \qquad (3.14)$$

where $|M(R=1, q)|$ is the number of unique terms in the relevance model and $\delta_d \in [0, 1]$ is a constant.

3.2.4 Combining Context Scores

The score of a context (i.e., context score) $c(d_i, k)$ is calculated using Equation 3.3 and it is the weight of the query term $d_i[k] \in q$ (i.e., $weight(d_i, k)$) at location k in the document d_i:

$$weight(d_i, k) = \log\left(\frac{P(c(d_i, k) \mid R = 1, q)}{P(c(d_i, k) \mid R = 0, q)}\right) \tag{3.15}$$

For combining context scores, we need $weight(d_i, k)$ to be between zero and one. So we normalize $weight(d_i, k)$ by the linear normalization [Lee, 1997] across documents:

$$w(d_i, k) = \frac{weight(d_i, k) - \min(weight)}{\max(weight) - \min(weight)} \tag{3.16}$$

where min(*weight*) is the minimum context score obtained among all the retrieved documents and max(*weight*) is the maximum context score obtained among all the retrieved document. Using the linear normalization, the second term of Equation 3.2 can be eliminated.

A document may contain more than one contexts (i.e., when the query terms occur more then once in the document). Hence, we need to aggregate the context scores for obtaining the document score for ranking. This is similar to combining passage scores in passage-based retrieval [Callan, 1994; Kaszkiel and Zobel, 1997]. Previously, passage scores are combined using arithmetic mean, as well as taking the maximum [Callan, 1994].

Kong et al. [2004] proposed three principles regarding how to make the relevance decision for a document about a particular topic by combining relevance of document parts, as follows:

(a) the *Disjunctive Relevance Decision* (DRD) principle which states that a document is relevant to a topic if a particular document part is relevant to the topic;

(b) the *Aggregate Relevance Decision* (ARD) principle which states that a document is more relevant to a topic if more occurrences of the concepts related to the topic are found in the document; and

(c) the *Conjunctive Relevance Decision* (CRD) principle which states that a document is relevant to a topic if all the document parts are relevant to the topic.

According to Harman [2004], the TREC ad hoc evaluation is recall-oriented and if any part of the document is relevant, the TREC evaluator considers that the entire document is relevant for ad hoc retrieval. Therefore, the DRD principle seems to be consistent with the TREC evaluation policy for ad hoc retrieval.

3.2.4.1 Extended Boolean Operators

We used the extended Boolean conjunction and disjunction [Fox et al., 1992] (i.e., the p Norm) to test different methods (i.e., AND and OR) for combining the context scores in a document d_i. The combined score is the document score, $sim(d_i, q)$ (Table 3.3). The parameter p in the extended Boolean conjunction or disjunction is the soft/hard decision parameter and $p \geq 1$. The parameter m in Table 3.3 is the total number of occurrences of the query terms in d_i (i.e., the number of interested contexts found in d_i).

Table 3.3: Formula of extended Boolean conjunction and disjunction.

Extended Boolean conjunction (AND)	Extended Boolean disjunction (OR)
$sim(d_i, q) = 1 - \sqrt[p]{\frac{1}{m} \sum_{k:d_i[k] \in q} (1 - w(d_i, k))^p}$	$sim(d_i, q) = \sqrt[p]{\frac{1}{m} \sum_{k:d_i[k] \in q} w(d_i, k)^p}$

For the extended Boolean operators, when $p = 1$, the extended Boolean conjunction and the disjunction are the same which is the arithmetic mean of the context scores in a document. When $p = \infty$, the extended Boolean conjunction (AND) returns the minimum context score in the document while the extended Boolean disjunction (OR) returns the maximum context score in the document. Note that the extended Boolean disjunction is the

same as the generalized mean function [Dyckhoff and Pedrycz, 1984] which complies with the ARD principle [Kong et al., 2004].

3.2.4.2 Dombi Operators

Besides the extended Boolean operators, we also used the fuzzy operators for combining the context scores in a document. In the framework of fuzzy set theory [Zadeh, 1965], the fuzzy conjunction operator complies with the CRD principle while the fuzzy disjunction operator complies with the DRD principle [Kong et al., 2004]. We used the Dombi's [Dombi, 1982] fuzzy operators to experiment the two principles (Table 3.4) where p is again the soft/hard decision parameter and $p \geq 1$. For the Dombi's operators, when $p = \infty$, similar to the extended Boolean operators, the Dombi's conjunction (AND) returns the minimum context score in the document while the Dombi's disjunction (OR) returns the maximum context score in the document [Dombi, 1982].

Table 3.4: Formula of Dombi's conjunction and disjunction.

Dombi's conjunction (AND)	Dombi's disjunction(OR)
$sim(d_i, q) = \frac{1}{1 + \sqrt[p]{\sum_{k:d_i[k] \in q} \left(\frac{1}{w(d_i, k)} - 1 \right)^p}}$	$sim(d_i, q) = \frac{1}{1 + \sqrt[-p]{\sum_{k:d_i[k] \in q} \left(\frac{1}{w(d_i, k)} - 1 \right)^{-p}}}$

3.2.4.3 Ordered Weighted Averaging (OWA) Operators

Apart from the extended Boolean operators and Dombi's operators, there are also other aggregation operators in multi-criteria decision making such as the ordered weighted averaging (OWA) operators proposed by [Yager, 1988]. OWA operators have been used in applications of decision making, expert system, neural networks and etc. An OWA operator with dimension m is a mapping F: $\mathbf{R}^m \rightarrow \mathbf{R}$ that has an associated weighting vector $Y = (y_1, \ldots, y_m)^T$ having the properties (a) $y_1 + \cdots + y_m = 1$ and (b) $0 \leq y_j \leq 1$ for $j = 1, \ldots, m$, such that:

$$F(a_1,\ldots,a_m) = \sum_{j=1}^{m} y_j b_j \qquad (3.17)$$

where b_j is the j-th largest element of the collection of the aggregated objects $\{a_1, \ldots, a_m\}$.

For our proposed model, let $(a_1, ..., a_m)$ be the vector of the m context scores of d_i and $(b_1, ..., b_m)$ be the ordered (*descending*) vector of the m context scores of d_i. Then,

$$sim(d_i, q) = F(a_1,\ldots,a_m) = \sum_{j=1}^{m} y_j b_j \qquad (3.18)$$

is the aggregated score (i.e., document score) of d_i for the query q.

An important issue of the theory of OWA operators is to determine the weighting elements y_j, $j = 1, \ldots, m$ of the weighting vector Y. There are some special cases of defining Y, for examples:

(a) Taking the maximum: $y_1 = 1, y_2 = \cdots = y_m = 0$.

(b) Taking the minimum: $y_1 = \cdots = y_{m-1} = 0, y_m = 1$.

(c) Taking the arithmetic mean: $y_1 = \cdots = y_m = 1 / m$.

From the above examples, it should be clear that different ways of defining the weighting vector Y yield different OWA operators. We investigated two previous retrieval models (namely the MMM model [Waller and Kraft, 1979] and the Paice model [Paice, 1984]) that can be considered to be using OWA operators.

The MMM model considers only the minimum and maximum context scores in a document using the coefficients $Cand \in [0, 1]$ and $Cor \in [0, 1]$ for AND and OR operations respectively:

The MMM model (AND):

$$sim(d_i, q) = Cand \times \min_{k:d_i[k] \in q} \left(w(d_i, k)\right) + (1 - Cand) \times \max_{k:d_i[k] \in q} \left(w(d_i, k)\right) \quad (3.19)$$

The MMM model (OR):

$$sim(d_i, q) = Cor \times \max_{k:d_i[k] \in q} \left(w(d_i, k)\right) + (1 - Cor) \times \min_{k:d_i[k] \in q} \left(w(d_i, k)\right) \quad (3.20)$$

When applying the MMM model in our aggregation of context scores, we only need one equation because *Cor* = 1 – *Cand*. Therefore, we use a single parameter $\alpha \in [0, 1]$ in the weighting vector *Y* such that $y_1 = \alpha$, $y_2 = \cdots = y_{m-1} = 0$, $y_m = 1 - \alpha$:

$$Y = (\alpha, 0, \ldots, 0, 1 - \alpha)^T \quad (3.21)$$

and use Equation 3.18 for ranking the documents.

The Paice model [Paice, 1984] uses a normalized geometric series with a parameter $r \in [0, 1]$ for weighting the criteria. Assume that we have *m* criteria for making decisions (e.g. the dimension of the OWA operator is *m*) and let *S* be the geometric sum:

$$S = 1 + r + r^2 + \cdots + r^{m-1} \quad (3.22)$$

The weighting vector *Y* of the Paice model for AND and OR operators are the normalized geometric series in Table 3.5. The ranking formula is using Equation 3.18 like the MMM model with the corresponding weighting vector *Y*. For the Paice model, the weighting vectors for AND and OR operators are the reverse of each other.

Table 3.5: The weighting vector *W* for the Paice model AND and OR operator.

The Paice model AND	The Paice model OR
$Y = \left(\frac{r^{m-1}}{S}, \frac{r^{m-2}}{S}, \cdots, \frac{r}{S}, \frac{1}{S}\right)^T$	$Y = \left(\frac{1}{S}, \frac{r}{S}, \cdots, \frac{r^{m-2}}{S}, \frac{r^{m-1}}{S}\right)^T$

Both the MMM model and the OR operator of the Paice model comply with the ARD principle. In Table 3.6, we group the aggregation operators described above using the three principles in [Kong et al., 2004] by their consistency with the axioms of the different relevance decision principles. The grouping in Table 3.6 is not exclusive because any operators which comply with the CRD/DRD principle also comply with the ARD principle. However, the opposite may or may not be true.

Table 3.6: Grouping of the aggregation operators using the three principles.

CRD principle	ARD principle	DRD principle
Dombi's AND	Extended Boolean OR	Dombi's OR
	The MMM model	
	The Paice model OR	

3.3 Model-Oriented Experiments

In this section, we present the results of the model-oriented experiments which extensively investigate the factors affecting the effectiveness of the model using the TREC-6 ad-hoc collection. This collection contains 556,077 English documents. We use the TREC-6 title (short) queries 301-350 in the experiments. Title queries are used because they have few (one to four) query terms which are similar to the lengths of web queries. All the terms in the documents and queries are stemmed using the Porter stemming algorithm [Porter, 1980]. Stop words are removed in both the documents and queries. Terms with document frequency equals to one are also removed in the documents. This is because we do not want document-identifying terms such as document ids to be included in the training and retrieval process. For statistical inference, we performed various non-parametric (Wilcoxon) statistical significance tests. Non-parametric tests are used because we do not know the underlying distributions of the mean average precision (MAP) performances of the retrieval systems. We also report the precision of the top 10 documents (i.e., P@10) and the R-precision in the experiments. The top 10 document precision is a precision-oriented measure

which complements the recall-oriented measures like MAP. R-precision is provided for reference.

3.3.1 Document-Training V.S. Context-Training

In this section, we compare the performances of using document-training and context-training with different context sizes (defined as $2n+1$). The objectives of the experiments in this section are to determine:

(a) whether document-training or context-training is better; and

(b) the most suitable n empirically (i.e., the context size).

The smoothing technique used in all the experiments in this section is the Laplace smoothing (i.e., setting $\delta_a = 1$ in Equation 3.12) for simplicity and the document score is the maximum context score found in the document (i.e., setting $p = \infty$ in the extended Boolean OR in Table 3.3) similar to some passage-based retrieval [Callan, 1994].

First, we performed a predictive retrieval using the BM25 term weight of the 2-Poisson model [Robertson and Walker, 1994] using the standard parameter setting [Walker et al., 1997] (i.e., k_1=1.2 and b=0.75) with passage-based retrieval and pseudo relevance feedback (PRF). The result in Table 3.7 is our baseline performance. The top 3000 retrieved documents using the 2-Poisson model are re-ranked by our model for later evaluations. Next, we experimented with the document-training (Doc-T) and context-training (Con-T) training methods and compare them with different context sizes $2n+1$ (Table 3.8).

Table 3.7: Our predictive baseline performance using the BM25 of 2-Poisson Model with passage-based retrieval and pseudo relevance feedback (PRF).

	P@10	MAP	R-Precision
TREC-6	.4540	.2791	.3051

In Table 3.8, the differences between the MAP of context-training over document-training are statistically significant with 99.9% confidence interval (C.I.) for all tested n. This is expected based on our earlier argument

that the context-training assumption is more realistic than the document-training assumption. Thus, context-training has a higher MAP performance. The results also suggest that using contexts is a viable method in information retrieval. It should be noted that the highest MAP of document-training is obtained when n = 35 (i.e., context size of 71) and the highest MAP of context-training is obtained when n = 50 (i.e., context size of 101). This suggests that document-training favours smaller contexts while context-training favours larger contexts. This may due to the fact that document-training has more terms to match than context-training. Based on the results of the experiments in this section (Table 3.8), we find out that, for TREC-6 ad-hoc collection, context-training is preferred over document-training and n should be set to 50 (i.e., context size of 101) to obtain a balance of good effectiveness and efficiency, as increasing n also increases the processing time. Note that the purpose of Table 3.8 is to show an example that context-training is preferred because it is compatible with the query-centric assumption.

Table 3.8: Comparison between Document-Training (Doc-T) and Context-Training (Con-T) with different context sizes 2n+1 in TREC-6.

n	P@10		MAP		R-Precision	
	Doc-T	Con-T	Doc-T	Con-T	Doc-T	Con-T
5	.3700	.4240	.2372	.2843*	.2724	.3181
15	.3600	.4520	.2687	.3429*	.3135	.3794
25	.3460	.4720	.2682	.3726*	.3174	.4218
35	.3540	.5160	**.2711**	.3881*	.3197	.4433
50	.3600	.5600	.2572	**.3913***	.3228	.4559
75	.3560	.5720	.2308	.3678*	.2983	.4320
100	.3520	.5800	.2120	.3469*	.2812	.4094
150	.3640	.5760	.1738	.2977*	.2511	.3580

(*) – indicates that the difference in MAP between Con-T and Doc-T is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% confidence interval (C. I.).

3.3.2 Smoothing

The experiments in this section aim to discover the effects of smoothing to our proposed model. We tested the model with different smoothing techniques (namely additive smoothing (A), Jelinek-Mercer smoothing (JM) and absolute discounting (D)) in estimating $P_{M(R=1,\,q)}(t)$ to avoid the problem of zero probabilities (Section 3.2.3).

From the results of previous experiments (Section 3.3.1), we are using context-training and context size of 101 (i.e., n = 50) in all the experimental runs in this section (Table 3.9). The document score is the maximum context score found in the document [Callan, 1994]. In Table 3.9, δ can be the parameter δ_a, δ_{jm} and δ_d of additive smoothing (Equation 3.12), Jelinek-Mercer smoothing (Equation 3.13) and absolute discounting (Equation 3.14), respectively, depending on the columns in the table.

Table 3.9: Results of using additive smoothing (A), Jelinek-Mercer smoothing (JM) and absolute discounting (D) with different values of δ (δ_a, δ_{jm} and δ_d).

δ	P@10			MAP			R-Precision		
	A	JM	D	A	JM	D	A	JM	D
0.1	.866	.888	.898	**.651**	.705	**.707**	.651	.694	.697
0.3	.794	.884	.898	.588	.702	.706	.595	.692	.697
0.5	.726	.886	.896	.515	.702	.703	.540	.694	.696
0.7	.656	.890	.894	.461	.704	.700	499	.695	.690
0.9	.594	.896	.892	.412	**.706**	.695	.469	.696	.680

For additive smoothing (A), we can see that from Table 3.9 when the value of δ_a increases, the performance decreases. The best performance is obtained when δ_a (Equation 3.12) is equal to 0.1. We believe that the reason is due to the important effect of the presence or absence of the terms in the relevance model $M(R{=}1, q)$, while setting a larger value of δ_a would lower this effect (i.e., unseen terms are given a value of higher probabilities than they should be).

For Jelinek-Mercer smoothing (JM) and absolute discounting (D), we can see that from Table 3.9 the performances are quite stable over the ranges of δ_{jm} (Equation 3.13) and δ_d (Equation 3.14) from 0.1 to 0.9. From Table 3.9, the best performances (MAP) for each of the smoothing techniques are highlighted (δ_a = 0.1, δ_{jm} = 0.9 and δ_d = 0.1 in additive smoothing, Jelinek-Mercer smoothing and absolute discounting respectively). The best performances for each of the smoothing techniques are similar to each other. The best MAP obtained among all the runs in this section is 0.7078 which is absolute discounting with δ_d = 0.1 (Table 3.9). We believe that the effect of smoothing of using *different* smoothing techniques is not very different

when the optimal values of the parameters δ_a, δ_{jm} and δ_d are determined. In the subsequent experiments, absolute discounting with $\delta_d = 0.1$ is used. We do not evaluate with parameter values which less than 0.1 or greater than 0.9 because the purpose of Table 3.9 is for comparison but not finding the optimal parameter values. In practice, the optimal parameter values can be determined using cross-fold validation. Note that the objective of the experiments in this section is not to find the optimal parameter values but to reveal the performance of the document-context based model in TREC-6. As a result, we do not perform sensitivity studies on the parameters.

3.3.3 Context Scores Aggregation

The experiments in this section try to discover the best aggregation operator discussed in Section 3.2.4 for combining the context scores in a document, and find out which of the 3 decision principles (i.e., the CRD, the ARD and the DRD principles) [Kong et al., 2004] performs better. From the results of the previous experiments, we used context-training as the training method, the context size of 101 (i.e., $n = 50$) and absolute discounting with $\delta_d = 0.1$ (Equation 3.14). First, we test the extended Boolean operators (Table 3.3) and the Dombi's fuzzy operators (Table 3.4). The results are shown in Tables 3.10 and 3.11.

Table 3.10: Results of using the extended Boolean operators with different values of p.

p	P@10		MAP		R-Precision	
	AND	OR	AND	OR	AND	OR
1	.8940	.8940	**.7105**	.7105	.7072	.7072
5	.8800	.8980	.6854	.7139*	.6842	.7107
10	.8660	.8980	.6605	.7157*	.6584	.7110
20	.8600	.8980	.6432	**.7163***	.6407	.7123
40	.8520	.9000	.6342	.7141*	.6293	.7054
∞	.8500	.8980	.6235	.7078*	.6199	.6972

(*) – indicates that the difference in MAP between extended Boolean AND and OR operators is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

From Table 3.10, when $p = 1$, the results of extended Boolean AND is the same as that of extended Boolean OR as the formulae for the two operators when $p = 1$ are the same. As p increases, the performance difference

between extended Boolean AND and extended Boolean OR becomes apparent. From $p = 5$ onwards, the differences are statistically significant with 99.9% C. I. using the Wilcoxon matched-pairs signed-ranks test. When $p = \infty$, the extended Boolean AND is the same as using minimum context score as the document score and the extended Boolean OR is the same as using maximum context score as the document score. In general, the results in Table 3.10 suggest that extended Boolean OR is better than extended Boolean AND when combining the context scores.

Table 3.11: Results of using the Dombi's operators with different values of p.

p	P@10		MAP		R-Precision	
	AND	OR	AND	OR	AND	OR
1	.5940	.5840	.3349	.3937	.3474	.4305
5	.8360	.8940	.6172	.6951*	.6144	.6839
10	.8520	.8980	.6244	.7061*	.6206	.6967
20	.8500	.8980	**.6245**	.7077*	.6200	.6977
40	.8500	.8860	.6239	.7055*	.6195	.6972
∞	.8500	.8980	.6235	**.7078***	.6199	.6972

(*) – indicates that the difference in MAP between Dombi's AND and OR operators is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

From Table 3.11, we can see that Dombi's OR operator performs better than Dombi's AND operator in all cases of p when aggregating the context scores. This suggests that Disjunctive Relevance Decision (DRD) principle is preferred over Conjunctive Relevance Decision (CRD) principle. The reason is that in the TREC relevance judgements, if a part of a document is judged relevant, then the whole document is judged relevant. This in fact favours the DRD principle. The results also confirm with that in [Kong et al., 2004] in which DRD principle is preferred over CRD principle.

Next, we test the Ordered Weighted Averaging (OWA) operators (Equation 3.18). Table 3.12 shows the results of using the MMM model (Equation 3.21) with different values of α. When α increases from 0.1 to 0.9, the MMM model behaves from AND operator to OR operator. We can see that the OR operator is better than the AND operator using the MMM model as the performance (MAP) increases while the value of α goes from 0.1 to 0.9. We obtain the best MAP with α= 0.7 (highlighted in Table 3.12).

Table 3.12: Results of using the MMM model with different values of α.

α	P@10	MAP	R-Precision
0.1	0.8560	0.6400	0.6364
0.3	0.8680	0.6728	0.6702
0.5	0.8860	0.6965	0.6959
0.7	0.8940	0.7104	0.7079
0.9	0.9000	**0.7112**	0.7007

Table 3.13 shows the results of using the Paice model operators (Table 3.5) with different values of *r*. From Table 3.13, the performance difference in the AND and OR operators of the Paice model is larger for small *r* (i.e., $r \leq 0.5$) but not for large *r* (i.e., $r > 0.5$). Both the performance of the Paice model AND and OR operators increases with *r* from 0.1 to 0.9. The best MAP (highlighted) for both operators are obtained when $r = 0.9$.

Table 3.13: Results of using the Paice model with different values of *r*.

r	P@10		MAP		R-Precision	
	AND	OR	AND	OR	AND	OR
0.1	.8520	.8980	.6287	.7083*	.6250	.6980
0.3	.8560	.8980	.6416	.7089*	.6382	.6985
0.5	.8620	.9000	.6585	.7090*	.6565	.7003
0.7	.8700	.8980	.6800	.7091	.6797	.6993
0.9	.8800	.8960	**.7015**	**.7103**	.6998	.7057

(*) – indicates that the difference in MAP between the Paice AND and OR operators is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

From Tables 3.10 – 3.13, the performance of OR operators is better than that of corresponding AND operators. While OR operators behave like DRD principle and AND operators behave like CRD principle, all the results suggest that DRD principle is preferred because on average the MAP performance is higher. In Table 3.6, we grouped different operators using the relevance decision principles [Kong et al., 2004], and we compare the results between each of the groups. Table 3.14 shows the best result obtained using the aggregation operators, which grouped the contexts scores according to the relevance decision principles in Table 3.6. Using the best result in each of the operators, we pair-wise compare them with statistical tests in Table 3.15.

Table 3.14: Best results obtained from different aggregation operators which are grouped by the relevance principles in Table 3.6.

	Operator	P@10	MAP	R-Precision
CRD	Dombi's AND	0.8500	0.6245	0.6200
ARD	Extended Boolean OR	0.8980	0.7163	0.7123
	The MMM model	0.9000	0.7112	0.7007
	The Paice model OR	0.8960	0.7103	0.7057
DRD	Dombi's OR	0.8980	0.7078	0.6972

Table 3.15: Comparisons between the relevance decision principles using best MAP performance of the aggregation operators.

		CRD	ARD			DRD
	p value	Dombi's AND	Extended Boolean OR	The MMM model	The Paice model OR	Dombi's OR
CRD	Dombi's AND	-	< .0010*	< .0010*	< .0010*	< .0010*
ARD	Extended Boolean OR	-	-	< .0010*	.0128	< .0010*
	The MMM model	-	-	-	.8876	< .0010*
	The Paice model OR	-	-	-	-	.0641
DRD	Dombi's OR	-	-	-	-	-

(*) – indicates that the difference in the best MAP performance between the 2 operators is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

In Table 3.15, the smaller the *p* value obtained in the statistical test, the larger the confidence interval in MAP performance between the two operators. We can see that the MAP performance of ARD principle is statistically significantly different (i.e., higher, see Table 3.14) from that using the operator based on the CRD principle. The MAP performance of DRD principle is also statistically significantly different (i.e., higher, see Table 3.14) from that using the operator based on the CRD principle. This suggests that the ARD principle is robust.

In Table 3.14, the MAP performance using the DRD principle is similar to the ARD principle. However, in Table 3.15, there is a statistical significant difference between the MAP using operators based on the ARD principle and the MAP using the operators based on the DRD principle. This difference might be due to the difference between the DRD and ARD principles where the DRD assert the additional boundary condition. That is, if any single document part is relevant, then the entire document is considered relevant. In practice, this occurs only for the top ranked context

affecting only one document because all the other contexts are assigned normalized scores that are less than one by the linear normalization [Lee, 1997]. Since there is statistical significant difference, it seemed that the top ranked context may have a noticeable impact on the effectiveness because subsequent recall and precision values at different locations in the retrieval list are affected by it. When the top ranked context is assigned a value of one by the linear transformation [Lee, 1997], the document ranking score is one by the boundary condition of the DRD principle, thereby loosing the differentiability of relevance scores in the context of that document. However, for the AP principle, the document that contains the top ranked context does not necessarily have the highest document relevance score of one, because the other context scores in the document may affect the final document relevance score. Also, in practice, many relevant documents require multiple contexts to make relevance judgments and the likelihood of making a relevance judgment on the basis of a single context in a relevant document is not high. These mitigating factors suggest why the operators based on the ARD principle appeared to be performing slightly better than the operators based on the DRD principle, even though the DRD principle is consistent with the TREC ad hoc evaluation policy.

3.3.4 Probability Ranking Principle

The probability ranking principle [Robertson, 1977] states that for a particular query q, if a retrieval model ranks the documents in the collection in the order of *decreasing* probability of relevance to q, then the best overall effectiveness of the model will be achieved with respect to the data available to the model. The probability ranking principle assumes that the relevance of a document d_i to a query q is independent to all other documents $\{d_j : j \in [1, card(D)] \text{ and } j \neq i\}$ in the collection ($card(D)$ is the number of documents in the collection D). If a retrieval model obeys the probability ranking principle, with more and more relevance information for q available to the model, we expect that the model will never decrease the performance for q. Because if the model's performance is degraded when it has more

relevance information, this means that the best overall effectiveness is not achieved for having more relevance information and this violates the probability ranking principle. In this section, we provide the experimental results to show that our proposed model obeys the probability ranking principle for TREC-6 data.

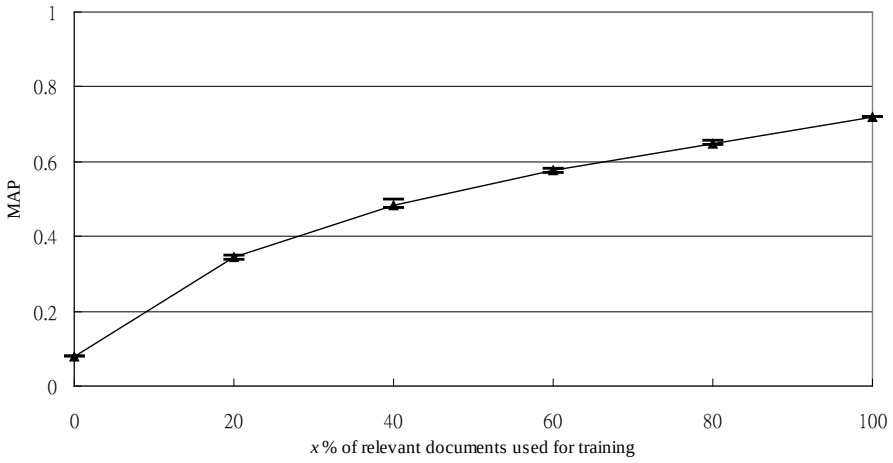

Figure 3.1: Performance of our model using different % of relevant documents for training. The bars show the maximum and minimum MAP of the five retrievals.

Using context-training with context size of 101 (i.e., n = 50), additive smoothing with δ_d = 0.1 (Equation 3.14) and the extended Boolean OR operator with p = 20 (Table 3.3) for combining the context scores in a document, we tested the model using different percentage of relevant documents to train the model. More specifically, we randomly sample x% of the relevant documents (where x = 0, 20, 40, 60, 80 and 100) for a query q and use the randomly sampled relevant documents for training the model (i.e., for constructing $M(R{=}1, q)$ using context-training). Then, we perform a retrieval for q using the trained model. The procedure is repeated five times for every query of TREC-6 (i.e., for every query, we perform five retrievals using five sets of randomly sampled relevant documents for training). When x = 0 or x = 100, the performances of the five retrievals are the same. After the experiments, we discovered that for all 50 queries of TREC-6, the performance (MAP) increases *monotonically* when x goes from 0 to 100. In Figure 3.1, we show the average MAP for all 50 queries in the five retrievals while the maximum and minimum MAP of the five retrievals at

each level of x are also shown. The results suggest that our model obeys the probability ranking principle for the TREC-6 data set. The best performance is obtained when 100% of the relevant documents are used for training the model.

3.3.5 Validation of the Collection-Irrelevance Assumption

When calculating the context score, we used the collection model $M(D, q)$ to substitute/estimate the irrelevance model $M(R=0, q)$ by the collection-irrelevance assumption (Equation 3.9 in Section 3.2.2). As the assumption states that the collection model and the irrelevance model are similar to each other in a sufficiently large collection, it is expected that the MAP performance of using the two models are similar as well. In this section, we validate the collection-irrelevance assumption. When using the irrelevance model, smoothing should be applied to the model similar to that in the relevance model. From the results of smoothing of the relevance model (Table 3.9), we are using absolute discounting with $\delta_d = 0.1$ (Equation 3.14). Table 3.16 shows the performance difference when using the collection model and the irrelevance model.

Table 3.16: Difference in results of using the collection model (col) and the irrelevance model (irrel).

P@10		MAP		R-Precision	
col	irrel	col	irrel	col	irrel
.8980	.9100	.7163	.7472*	.7123	.7423

(*) – indicates that the difference in MAP between the two results is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

From Table 3.16, although the MAP performance when using the irrelevance model is statistically significantly higher than that when using the collection model, the difference is only about 3%. This shows that, in doing retrospective experiments, using the irrelevance model can improve the MAP performance for most of the queries (to obtain the statistical significant difference) but the improvement is not large for each query. This confirms to our claim that the irrelevance model and the collection model are similar to each other. In order to reveal the optimal results, we are using

the irrelevance model instead of the collection model in the subsequent experiments in this chapter.

3.4. Scope-Oriented Experiments

In this last set of experiments, we test the reliability of the proposed model by experimenting it with different data collections (the ad-hoc retrieval of TREC-2, TREC-6, TREC-7 and the robust-track retrieval of TREC-2005) and another language (Chinese NTCIR-5). Similar to the experiments in Section 3.3, title (short) queries in each of collections are used as they are commonly found in web search. The performance of TREC-6 has been evaluated in the previous section (Section 3.3) and the TREC-7 data collection is a subset of the TREC-6 data collection.

Table 3.17: Statistics of the collections used in the experiments.

	TREC-2	TREC-6	TREC-7	TREC-2005	NTCIR-5
Language	English	English	English	English	Chinese
Topics	101-150	301-350	351-400	50 past hard topics	001-050
No. of documents	714,858	556,077	528,155	1,033,461	901,447
No. of relevant documents	11,645	4,611	4,674	6,561	3,052 (Relax) 1,885 (Rigid)
Storage (GB)	3.9	3.3	3.0	5.3	3.5

Table 3.17 shows some collection statistics of the data collections for the experiments reported in this section. Based on the results of the experiments in Section 3.3, we use context-training with context size $2n+1 = 101$ (i.e., $n = 50$), absolute discounting with $\delta_d = 0.1$ (Equation 3.14) and the extended Boolean OR operator with $p = 20$ (Table 3.3) for combining the context scores in a document for *all* the data collections tested in the scope-oriented experiments.

3.4.1 Different English Data Collections

Table 3.18 shows the results of the predictive baseline experiments using BM25 term weight of the 2-Poisson model [Robertson and Walker, 1994] with passage-based retrieval and pseudo relevance feedback (PRF) and our

retrospective experiments. The purpose of this comparison is to show that we have used state-of-the-art retrieval models (based on our implementation) and to show that our novel retrieval model is worth further investigation. For the latter, we provide the results of the statistical tests in Table 3.18 for completeness. We caution that comparing the results of retrospective experiments dryly with the predictive baseline experiments is unfair, as the former has relevance information while the latter does not.

Table 3.18: Our predictive baseline performance (predictive) using BM25 2-Poisson Model with passage-based retrieval and pseudo relevance feedback and our retrospective performance (retro) in different TREC data collections.

	P@10		MAP		R-Precision	
	predictive	retro	predictive	retro	predictive	retro
TREC-2	.5020	.9860	.2534	.7197*	.3096	.7010
TREC-6	.4540	.9100	.2791	.7472*	.3051	.7423
TREC-7	.4360	.9460	.2295	.7150*	.2662	.7065
TREC-2005	.4900	.9580	.2730	.7744*	.3147	.7613

(*) – indicates that the MAP difference between the retrospective and predictive experiments is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

In Table 3.18, the predictive performance of TREC-7 is not as good as the others. This is probably due to many of the relevant documents in TREC-7 do not contain any of the title query terms (i.e., about 12.9% of the relevant documents per TREC-7 topic do not contain any of the query terms in Table 3.2).

In order to test the robustness of our model in different TREC data collections, we test the results of the experiments using the Wilcoxon two sample test (results shown in Table 3.19). Wilcoxon matched-pairs signed-ranks test is not used here because different queries are used in different data collections and we cannot pair the retrieval effectiveness performance on the basis of the same topics. The Wilcoxon two sample test compare the MAP of two sets of topics in two collections and used the pooled variance that is estimated by summing the standard errors of MAPs of each set of topics. The null hypothesis is that the MAPs of two sets of topics in two different collections are the same.

Table 3.19: Comparisons between English data collections using Wilcoxon two sample test.

p value	TREC-2	TREC-6	TREC-7	TREC-2005
TREC-2	-	.2995	.5884	.0617
TREC-6	-	-	.6003	.7123
TREC-7	-	-	-	.4100
TREC-2005	-	-	-	-

From Table 3.19, the smaller the *p* value, the larger the confidence interval of the MAP performance of the two collections. We *cannot* conclude that the results in different TREC data collections are statistically significantly different with 94% confidence interval (C. I.) as all the *p* values are not smaller than 0.06 (i.e., the null hypothesis is not rejected at 94% C. I.). The results suggest that our model performs similarly over different TREC English data collections which show that the proposed model is not unreliable.

3.4.2 Different Language

We also test the proposed model using the Chinese collection, NTCIR-5 [Kishida et al., 2005], for showing that the model can operate with more than one language. Table 3.20 shows the results of the predictive baseline experiments which were obtained using BM25 of the 2-Poisson model based on bigram indexing with pseudo relevance feedback [Luk and Kwok, 2002]. As the NTCIR-5 has two sets of relevance judgments, relax and rigid, we have two results for the same run where Relax-E means evaluated using the relax relevance judgments and Rigid-E means evaluated using the rigid relevance judgments.

Table 3.20: Our predictive baseline performance using passage-based 2-Poisson Model with pseudo relevance feedback (PRF) based on bigram indexing.

	P@10	MAP	R-Precision
Relax-E	.5460	.3750	.3694
Rigid-E	.4340	.3398	.3443

In the retrospective experiments of NTCIR-5 using our model, we used bigram indexing to collect terms in contexts so as to be consistent with our Chinese indexing strategy for the initial retrieval. The NTCIR-5 data set has two sets of relevance judgements, Relax and Rigid, we can train the model

using context-training based on one set of relevance judgements and then evaluate the result using the same or another set of relevance judgments. This yields four combinations of training and evaluation results as shown in Table 3.21 (Relax-T means training using the relax judgments, Relax-E means evaluation using the relax judgments, Rigid-T means training using the rigid judgments and Rigid-E means evaluation using the rigid judgments).

Table 3.21: Our retrospective performance using the proposed model in NTCIR-5.

	P@10		MAP		R-Precision	
	Relax-T	Rigid-T	Relax-T	Rigid-T	Relax-T	Rigid-T
Relax-E	.8940	.8680	.8834	.7185	.8807	.6674
Rigid-E	.5980	.8600	.6490	.8897	.6414	.8888

Table 3.22: Comparisons between using relax judgments for training (Relax-T) and evaluation (Relax-E) and using rigid judgments for training (Rigid-T) and evaluation (Rigid-E).

p value	Relax-T, Relax-E	Rigid-T, Relax-E	Relax-T, Rigid-E	Rigid-T, Rigid-E
Relax-T, Relax-E	-	< .0010*	< .0010*	.2418
Rigid-T, Relax-E	-	-	.0051	< .0010*
Relax-T, Rigid-E	-	-	-	< .0010*
Rigid-T, Rigid-E	-	-	-	-

(*) – indicates that the difference in MAP between the two experimental runs is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I.

From Table 3.21, we see that the performance is higher when using the same relevance judgments for training and evaluation (i.e., "Relax-T, Relax-E" and "Rigid-T, Rigid-E"). When using relax judgments for training and rigid judgments for evaluation (i.e., "Relax-T, Rigid-E"), the relevance model $M(R=1, q)$ may contain noise for the rigid judgments. When using rigid judgments for training and relax judgments for evaluation (i.e., "Rigid-T, Relax-E"), the relevance model $M(R=1, q)$ may contain insufficient information for the relax judgments. Therefore, the results in Table 3.21 are not unexpected. In Table 3.22, we can see that the difference of using the same judgment for training and for evaluation compared with using different judgments for training and for evaluation is statistically significantly different.

In Table 3.23, we compare our retrospective results of the English TREC data collections (Table 3.18) and our retrospective results of the Chinese

NTCIR-5 collections (Table 3.21) using the Wilcoxon two sample test (as the queries used in different collections are different).

Table 3.23: Cross-language comparisons in different data collections using Wilcoxon two sample test.

p value	TREC-2	TREC-6	TREC-7	TREC-2005
Relax-T, Relax-E	< .0010*	< .0010*	< .0010*	< .0010*
Rigid-T, Relax-E	.9588	.3738	.5037	.0646
Relax-T, Rigid-E	.2034	.0402	.1276	.0043
Rigid-T, Rigid-E	< .0010*	< .0010*	< .0010*	< .0010*

(*) – indicates that the difference in MAP between the two cross-language experimental runs is statistically significant using the Wilcoxon two sample test with 99.9% C. I.

In Table 3.23, we see that the results in NTCIR-5 using the same judgment for training and evaluation (i.e., "Relax-T, Relax-E" and "Rigid-T, Rigid-E") are statistically different to all the results in the TREC collections based on 99.9% C.I. (i.e., $p < 0.001$). For other combinations, the differences are not significant at 99.9% C.I. The result of TREC-2 is the most similar to that of using rigid judgment for training and relax judgment for evaluation in NTCIR-5 ($p \leq 0.9588$).

3.5 Chapter Summary

In summary, we proposed a novel hybrid document-context retrieval model which uses existing successful techniques to explore the effectiveness of incorporating term locations inside a document into our retrieval model. We used the log-odds as based on by the well known BIR model [Robertson and Sparck-Jones, 1976] as the starting point for deriving our document-context based model. We extended the existing probabilistic model from the document level to the document-context level, in which relevant information are located using contexts in a document. For probability estimation, we use smoothing techniques [Chen and Goodman, 1996; Zhai and Lafferty, 2004] similar to that of the language modeling approach to information retrieval. When combining the context scores (i.e., combining the evidence of relevant information), we tried different aggregation operators (Section 3.2.4) including the extended Boolean operators, the Dombi's fuzzy operators and the ordered weighted averaging (OWA) operators (the MMM

model, the Paice model AND and OR operators) to aggregate the context scores. Following the work of Kong et al. [2004], we have not tried the probabilistic combination approaches such as the operators used in the Inquery/Indri [Strohman et al., 2004] systems. The probabilistic alternatives are interesting to try in future studies.

We tested the model extensively using the TREC-6 data collection with different context sizes, training methods, smoothing methods and context scores aggregation methods. We found out that context-training is preferred over document-training as the training method (Table 3.8). The context size $2n+1$ should be around 101 (i.e., n = 50) for balanced effectiveness and efficiency. We also compared different smoothing techniques (Table 3.9) for solving the problem of zero probability in the estimation step and found out the different smoothing techniques produce similar results when the optimal parameter is determined. From Table 3.9, we used the absolute discounting with δ_d = 0.1 (Equation 3.14). After comparing different aggregation methods, the extended Boolean OR operator produced the best result in our model-oriented experiments.

We also tested the model with different data collections. The experiments in this chapter showed that the model is effective for the different reference data collections with various sizes and languages (i.e., TREC-2, TREC-6, TREC-7, TREC-2005 and NTCIR-5). The main remaining problem in the model is to estimate the relevance model $M(R=1, q)$ which defines the relevant term set and its probability distribution. In Chapters 5 and 6, we are testing document-context models with less relevance information (e.g., relevance feedback with limited top N retrieved relevance information) in order to make them operate effectively in predictive experiments.

Chapter 4

Interpreting TF-IDF Term Weights as Making Relevance Decisions

In this chapter a novel probabilistic retrieval model is presented. It forms a basis to interpret the TF-IDF term weights as making relevance decisions. It simulates the "local" relevance decision-making for every document location of a document, and combines all of these local relevance decisions as the "document-wide" relevance decision for the document. The significance of interpreting TF-IDF in this way is the potential: (1) to establish a unifying perspective about information retrieval as relevance decision-making; and (2) to develop advanced TF-IDF-related term weights for future elaborate retrieval models. Our novel retrieval model is simplified to a basic ranking formula that directly corresponds to the TF-IDF term weights. In general, we show that the term frequency factor of the ranking formula can become different term-frequency factors of existing retrieval systems. In the basic ranking formula, the remaining quantity, $-\log P(R=0 \mid t \in d)$, is interpreted as the probability of randomly picking a non-relevant usage (denoted by $R = 0$) of term t. Theoretically, we show that this quantity can be approximated by the inverse document frequency (IDF). Empirically, we show that this quantity is related to IDF using four reference TREC ad hoc retrieval data collections.

4.1 Introduction

This chapter presents a basis to interpret the well-known TF-IDF term weights [Robertson and Sparck Jones, 1976; Yu and Salton, 1976] as making relevance decisions. This basis is our novel probabilistic retrieval model that simulates human relevance decision-making for two types of relevance. One new type is the "local" relevance that only applies to a specific document-location, and the other common type is the "document-

wide" relevance that applies to the entire document. The model combines the local relevance for every document location of a document to form the document-wide relevance decision of the document. The local relevance at location *k* is the outcome of the local relevance decision, which is made on the basis of the available information in the document-context centered at *k*. If the document is locally relevant at any document location, then the entire document is deemed document-wide relevant to the query. This way of combining local relevance at different document locations to arrive at a document-wide relevance decision is consistent with the TREC ad hoc evaluation policy [Harman, 2004] as described in Section 4.2.4.

We are motivated to justify theoretically and empirically that TF-IDF term weights can be the outcome of modeling relevance decision-making. The significance of this justification is that potentially there is a unifying perspective about information retrieval (IR) as relevance decision-making. Many past retrieval models are already related to relevance decision-making; for example, the binary independence retrieval (BIR) model [Robertson and Sparck Jones, 1976], the logistic regression model [Cooper et al., 1992], the vector space model [Salton et al., 1975], the Boolean model [Wong et al., 1986], and the extended Boolean model [Salton et al., 1983]. However, it is not known whether TF-IDF term weights are related to relevance decision-making because they were originally not conceived in this way. Instead, the term frequency factor was originally thought to be indicative of document topic [Luhn, 1958], and the inverse document frequency (IDF) is reasoned [Sparck Jones, 1972] on the basis of Zipf law.

The original TF-IDF term weights are thought to be attribute values of documents that are treated as an indivisible object in many IR models. From our novel perspective, TF-IDF term weights are treated as the outcome of local relevance decision-making at different document locations. This novel perspective is a new avenue to develop more novel retrieval models, and it extends the original TF-IDF term weights to model microscopic phenomena at the document location level, rather than the macroscopic phenomena at the document level. This new perspective also demands a new

representation of a document as a string of words instead of the common vector representation, because the string representation of a document exposes information in the document for the purpose of mathematical modeling.

The simplified basic ranking formula of our probabilistic retrieval model that provides a basis to interpret TF-IDF term weight is the probability of relevance $P(R_{d,q}=1)$ that is rank equivalent (i.e., denoted by $\propto$) to the sum of products:

$$P(R_{d,q}=1) \propto \sum_{t \in q \cap d} f(t,d) \times [-\log P(R=0 \mid t \in d)] \qquad (4.1)$$

where $f(t, d)$ is the occurrence frequency of term t in document d and the quantity $-\log(R=0 \mid t \in d)$ corresponds to IDF. Details of the symbols and their descriptions of the previous formula are listed in Table 4.1.

The previous basic ranking formula is consistent with the probability ranking principle [Robertson, 1977] because it ranks documents by the probability of relevance. The term frequency factor of the basic ranking formula is $f(t, d)$, and the remaining quantity $-\log(R=0 \mid t \in d)$ is theoretically approximated by IDF. This approximation of $-\log(R=0 \mid t \in d)$ is also supported empirically, using four reference TREC ad hoc test collections. For generality of modeling, the quantity, $-\log(R=0 \mid t \in d)$, can also be approximated by the inverse collection term frequency (ICTF) [Kwok, 1995], which has been found to correlate with IDF using those reference ad hoc test collections. An independent, empirical approach, using clustering to estimate the quantity, $-\log(R=0 \mid t \in d)$, illustrates the explanatory power of the above basic ranking formula.

The rest of this chapter is organized as follows. Section 4.2 describes our novel probabilistic retrieval model that forms a basis to interpret the TF-IDF term weights. The ranking formula of this model is simplified to the basic ranking formula that directly corresponds to the TF-IDF term weights.

Section 4.3 shows that the term frequency factor $f(t, d)$ can be rendered into different term-frequency factors in the literature [Salton and Buckley, 1988; Robertson and Walker, 1994] by normalizing the document length. Section 4.4 interprets the quantity $P(R = 0 \mid t \in d)$ of the basic ranking formula as the probability of randomly picking a non-relevant usage of term *t*. We show that $-\log(R = 0 \mid t \in d)$ can be approximated by IDF. Another independent, empirical approach directly estimates $P(R = 0 \mid t \in d)$ using a novel clustering algorithm. Section 4.5 reports on the experiments relating to this approach. Section 4.6 describes related works. Section 4.7 concludes this chapter.

Table 4.1: Mathematical symbols used and their descriptions

Symbols	Description
D	A collection of documents
R	A relevance variable (R=1 means relevant, R=0 means irrelevant)
$R_{d,q}$	Document-wide relevance variable for document d and query q
$R_{d,k,q}$	Local relevance variable at location k of document d for query q
$card(.)$	The cardinality of its argument
d	A document (which is typically considered as a string or a set of words)
$\|d\|$	The length (total number of terms) of the document d
$d[k]$	The term located at the k-th logical position in document d
$c(d, k)$	A context of size $2n$+1 terms located at position k in document d
q	A query
$\propto$	Rank-equivalence binary relation
$\nabla(d, q)$	Document-wide relevance decision function for document d and query q
$\partial_{d,k}(c(d, k), q)$	Local relevance decision function at location k in document d for query q
$C(.)$	The generic function that combines the outcome of the local relevance decisions
$f(t, d)$	The occurrence frequency of term t in document d
$f(t, D)$	The total occurrence frequency of term t in all the documents
$d_{REL,q}$	The set of documents relevant to q
$f(t, d_{REL,q})$	The total occurrence frequency of t in documents relevant to q
$Loc(t, d)$	The set of location of term t in document d
$df(t)$	The document frequency of t
$IDF(t)$	Inverse document frequency of term t
$v(.)$	The vector representation of its argument
$\Leftrightarrow$	Two way implication
$\wedge$	Conjunction operator
$m(t)$	The total number of usage of the term t
$\Lambda(t)$	The arrival rate of the new usages of t per document
$\eta(t)$	The number of new usages of t
$\bullet$	The dot product of two vectors
$W_E(t)$	The expectation weight of term t
$E(.)$	The expectation operator

4.2 Probabilistic Non-relevance Decision Model

We formulate our probabilistic model as follows. Section 4.2.1 specifies the general model using document-context ranking, and it justifies the use of

document-contexts. Section 4.2.2 develops our probabilistic model and derives the context-based ranking formula (Equation 4.6). In Section 4.2.3, this formula is simplified to the basic ranking formula that directly corresponds to the TF-IDF term weights.

4.2.1 General Model

In Chapter 3 (A retrospective study of a hybrid document-context based retrieval model), we implicitly distinguish two types of relevance: the common document-wide relevance, $R_{d,q}$, that applies to the entire document d for query q, and the new local relevance, $R_{d,k,q}$, that applies only at the document location k in d for q. Both local and document-wide relevance can be binary values (i.e., 0 or 1), or real values representing the degrees of local and document-wide relevance to the query q, respectively. Typically, these real values are normalized between zero and one, without loss of generality.

We simulate a human evaluator who scans the document for local relevance information (Figure 1.1). Scanning involves iterating through every document location, and deciding for each whether local relevance information is found. The local relevance for each document location is combined to form the document-wide relevance of the entire document. Mathematically, the document-wide relevance is specified by the following general equation:

$$R_{d,q} = C(\{R_{d,k,q} : k \in [1,|\, d \,|], k \in \mathrm{N}\}) \tag{4.2}$$

where N is the set of positive integers, $|d|$ is the length of document d, and $C(.)$ is the general mathematical function that combines the local relevance. We assume that the first location of any document starts at 1.

According to the previous chapter, the outcome of a local relevance decision at location k of document d is determined by the information in the context that is denoted by $c(d, k)$. This context has n terms on the left, and another n

terms on the right from location *k* in *d* (i.e., *n* is an implicit parameter in the context *c*(*d*, *k*)). Figure 1.1 shows some examples of information extracted as contexts from a document for the query "Hubble Telescope Achievements". The keyword or middle term of the context is the term being scanned at present, and the human evaluator decides whether the information in the context of the middle term is locally relevant at that location.

The use of document-context assumes that document information that is far away from location *k* has negligible impact on the local relevance decision at location *k*. This is supported by past studies which found that: (1) the *n*-dependence entropy asymptotically approaches towards the entropy of a random model of character sequences [Wong and Ghahraman, 1975]; and (2) the mutual information of English text [Lucassen and Mercer, 1984] and Chinese text [Hung et al., 2001] decreases as the distance increases between the term in the middle of the context and the other term in the context. In local context analysis (LCA) [Xu and Croft, 2000] or lexical cohesion [Vechtomova et al., 2006], it is implicitly assumed that terms far away from the terms in the middle of the context have negligible impact, and thus such terms are ignored in the LCA and lexical cohesion. Moreover, the results of the previous chapter directly support the use of document-contexts for local relevance decision-making.

After defining document-contexts and supporting their use in information retrieval, we assume the following to simplify the modeling of local relevance decision-making:

Context-based Local Relevance Decision Assumption: A local relevance decision at any location *k* in any document *d* for any query *q* is made on the basis of the information in the context that is centered at *k* in *d* for some minimal context size *n*.

To model relevance decisions, we denote $\partial_{d,k}(.)$ as the local relevance decision at location k of document d. It is location based because local relevance is location dependent. According to the previous assumption, the input of local relevance decisions consists of the context $c(d, k)$ and the query q. Its output is the decision preference (as in Yao et al. [1991]) of the local relevance assigned by the human evaluator. According to the ordinal value theory [Chapter 3 in French, 1986], this decision preference can be transformed into a real value in [0,1]. For notation simplicity, we assume that $\partial_{d,k}(.)$ produces such an ordinal value that represents the local relevance decision preference. Therefore, the local relevance, $R_{d,k,q}$, at k in d for q is the outcome of the corresponding local relevance decision at k in d as follows:

$$R_{d,k,q} = \partial_{d,k}(c(d,k),q) \tag{4.3}$$

If $\partial_{d,k}(.)$ only returns 0 for local non-relevance and 1 for local relevance, then $R_{d,k,q}$ will be a binary variable for local relevance. Although $\partial_{d,k}(.)$ can be a real value in [0,1], we restrict our discussion in the thesis to binary variables for simplicity and clarity of representation. Similar to the local relevance variable, we assign the document-wide relevance variable $R_{d,\,q}$ with $\nabla(d, q)$ that contains the binary, document-wide relevance.

$$R_{d,q} = \nabla(d,q) \tag{4.4}$$

Using the definitions of local- and document-wide relevance, Equation 4.2 is specified in terms of making relevance decisions as follows:

$$\nabla(d,q) = C(\{\partial_{d,k}(c(d,k),q) : k \in [1,|\,d\,|], k \in \mathrm{N}\}) \tag{4.5}$$

The previous equation provides a direct, general mathematical description of the human evaluator making relevance decision, using a document-context based model for local relevance decision-making. It generalizes the work in Chapter 3 (A retrospective study of a hybrid document-context based

retrieval model) which models the set of local relevance, $\{R_{d,k,q}\}$, as local decision preferences that are defined as the normalized log-odds [Robertson and Sparck Jones, 1976] of the local relevance of the corresponding document-contexts, $\{c(d, k)\}$. The previous chapter experimented with several different implementation of the combining function, $C(.)$ (e.g., the extended Boolean disjunction [Salton et al., 1983], fuzzy disjunction [Dombi, 1982], or order weighted averaging operators [Waller and Kraft, 1979; Paice, 1984]). In contrast, $R_{d,q}$ and $R_{d,k,q}$ in this chapter are formulated as (random) binary variables in our probabilistic formulation for binary relevance. Instead of determining the output of the local relevance decisions, our probabilistic formulation combines the probability of local relevance decisions with the desirable outcomes to estimate the probability of document-wide relevance.

We denote the probability of document-wide relevance as $P_{\nabla}(R_{d,q})$, where the subscript, ∇, specifies that the relevance value of $R_{d,k}$ is produced by the document-wide relevance decision, $\nabla(.)$. Detailed arguments for $\nabla(.)$ are not necessary because they are completely specified by $R_{d,q}$ according to its definition. Similarly, the probability of local relevance is denoted by $P_{\partial,n}(R_{d,k,q})$, where ∂ specifies that the local relevance value of $R_{d,k,q}$ is produced by the local relevance decision, $\partial(.)$, with context size $2n+1$. Detailed subscripts and arguments for $\partial(.)$ are not necessary because they are completely specified by $R_{d,k,q}$ according to its definition, apart from the context size, n.

4.2.2 Context-based Ranking Formula

We model the relevance decision with non-relevance outcomes (similar to Calado et al. [2003]), and we rank documents by the probability of non-relevance in reverse order. For binary relevance, $P_{\nabla}(R_{d,q}=1)$ can be expressed as:

$$P_{\nabla}(R_{d,q}=1) = 1 - P_{\nabla}(R_{d,q}=0) \tag{4.6}$$

The probability of document-wide non-relevance in Equation 4.6 can be expressed in terms of the probability of local non-relevance by using the TREC evaluation policy for ad hoc retrieval tasks. According to Harman [2004], if any part of a document is judged relevant to the topic, then the entire document is considered as relevant in a TREC ad hoc retrieval task. Such an evaluation policy for ad hoc retrieval is used because the ad hoc retrieval tasks. Such an evaluation policy for ad hoc retrieval is used because ad hoc retrieval tasks are supposed to be recall oriented, and because such an inclusive policy enables later research on more specific relevance judgments [Harman, 2004]. Given this understanding of the evaluation policy and that we are dealing with binary relevance, a document *d* will be deemed document-wide not relevant to a query if every local relevance decision in the document is not relevant.

Logically, the TREC ad hoc evaluation policy for ad hoc retrieval tasks is specified as a two-way implication as:

$$(R_{d,q}=0) \Leftrightarrow \bigwedge_{k=1}^{|d|}(R_{d,k,q}=0) \tag{4.7}$$

where = is the equality test that returns true if the values are the same, and false otherwise. The previous logical relationship is a Boolean logic version of Equation 4.2, where *C*(.) in Equation 4.2 is specified as a conjunction in Boolean logic. Based on this logical relationship and Equation 4.5, the probability that the document is not relevant can be assigned as the joint probability that all local relevance of individual document locations is not relevant:

$$P_{\nabla}(R_{d,q}=0) = P_{\partial,n}((R_{d,1,q}=0),\ldots,(R_{d,|d|,q}=0)) \tag{4.8}$$

Note that the event spaces on the left hand side (LHS) and on the right hand side (RHS) of the previous equation are different. This is because the equation relates the two types of relevance, the document-wide- and local

relevance. From the perspective of mathematical modeling, the joint probability on the RHS of the previous equation simulates the local relevance decision-making with non-relevance outcome for the document *d*. The estimated joint probability is assigned to the probability of document-wide non-relevance on the LHS. It is expected in mathematical modeling that this probability assignment (on the RHS) is unlikely to be exactly the same as the true probability (on the LHS), because we do not expect perfect retrieval effectiveness performance. The question is whether the error of this probability assignment will have some impact on the retrieval effectiveness. To reduce this impact of error and yet without loss of generality, the assigned probability (on the RHS) is made rank equivalent with the true probability (on the LHS). Using this rank equivalence relation, Equation 4.6 becomes:

$$P_{\nabla}(R_{d,q}=1) \propto -\log P_{\partial,n}((R_{d,1,q}=0),\ldots,(R_{d,|d|,q}=0)) \tag{4.9}$$

In order to simplify the previous equation, we assume that the local relevance decisions with non-relevance outcomes are independent. Specifically, we give the next assumption.

Non-Relevance Independence Assumption: For any document *d* and any query *q*, $P_{\partial,n}(R_{d,k,q}=0 \mid R_{d,k-1,q}=0,\ldots,R_{d,1,q}=0) = P_{\partial,n}(R_{d,k,q}=0)$ for $k \in [1, |d|]$.

Although we do not believe the previous assumption to be true in practice because the contexts for making local relevance decisions overlap, this assumption, together with the chain rule, simplifies the joint probability, $P_{\partial,q}((R_{d,1,q}=0),\ldots,(R_{d,|d|,q}=0))$ in Equation 4.9 into the sum of the logarithm of the probability of its individual event. This is as follows:

$$P_{\nabla}(R_{d,q}=1) \propto -\sum_{k=1}^{|d|} \log P_{\partial,n}(R_{d,k,q}=0) \tag{4.10}$$

For occurrences of document terms that are not query terms, we assume that the outcomes of the local relevance decisions for these occurrences are not locally relevant. Using the common string notation that denotes $d[k]$ as the term at the k-th location in document d, the Query-Centric Assumption in the previous chapter states:

Query-Centric Assumption: For any query q and any relevant document d, the relevant information for q locates only in the contexts $\{c(d, k)\}$ for $k \in [1, |d|]$ where $d[k] \in q$. (i.e., the relevant information locates around query terms).

The preceding assumption is similar to that assumed by the binary independence model [Robertson and Sparck Jones, 1976] where non-query terms in the document are assumed not relevant. The query-centric assumption was corroborated using various TREC ad hoc retrieval test collections in the previous chapter.

The query-centric assumption implies that $P_{\partial,n}(R_{d,k,q} = 0) = 1$ when $d[k]$ is not a query term. This means that $\log P_{\partial,n}(R_{d,k,q} = 0) = 0$ if $d[k]$ is not a query term, so Equation 4.10 can be simplified by ignoring all locations where the query terms do not occur. Using the query-centric assumption and the notation that $Loc(t, d)$ is the set of document locations given that term t occurred in document d (i.e., $t \in d$), Equation 4.10 is simplified as follows:

$$P_{\nabla}(R_{d,q} = 1) \propto -\sum_{t \in q \cap d} \sum_{k \in Loc(t,d)} \log P_{\partial,n}(R_{d,k,q} = 0) \qquad (4.11)$$

4.2.3 TF-IDF Correspondence

Our non-relevance decision model in Section 4.2.2 can be shown to correspond to the TF-IDF term weights as follows. We shrink the context size to unity (i.e., set $n = 0$) based on the following assumption:

Minimal Context Assumption: For any query, the local relevance at a location k in a document d is determined only by the single term $d[k]$.

That is when $n = 0$, $c(d, k) = d[k]$. This assumption is not realistic because the local relevance at location k in document d is not decided by the context, but by the term $d[k]$. From another perspective, such an unrealistic assumption may explain the performance limitations of TF-IDF term weights. Another assumption is that the evaluator makes the same relevance decisions at different locations if the corresponding contexts are the same.

Location-Invariant Decision Assumption: If $c(d, j) = c(e, k)$, then $\partial_{d,j}(c(d, j), q) = \partial_{e,k}(c(d, k), q)$ for any query q.

This assumption is used by the document-context model in the previous chapter and was not considered unrealistic. Including the previous two assumptions implies that the probabilities of local non-relevance for the same query are the same for different locations, provided that the same term t occurs at these locations. Mathematically, the previous two assumptions imply that if $d[j] = e[k] = t$, then $P_{\partial,0}(R_{d,j,q}=0) = P_{\partial,0}(R_{e,k,q}=0)$. Consequently, we are no longer concerned with the locations of local non-relevance, but with the presence of query terms in the document. For presentation clarity, we simplify our notation to reflect this as follows.

When the context size is unity (i.e., $n = 0$), the probability of local non-relevance is:

$$P_{\partial,0}(R_{d,k,q}=0) = P_{\partial,0}(\partial_{d,k}(c(d,k),q)=0) = P_{\partial,0}(\partial_{d,k}(t,q)=0) \qquad (4.12)$$

where $c(d, k, 0) = d[k] = t$. For presentation clarity, we simplify our notation of the previous probability as:

$$P_{\partial,0}(R=0 \mid t \in d, q) = P_{\partial,0}(\partial_{d,k}(t,q)=0) \qquad (4.13)$$

where the term *t* occurred in *d*. The new notation only retains the input and output of the local relevance decision, $\partial_{d,k}(.)$, because it is only based on the term *t* occurring in *d* and the query *q* after the context size is reduced to unity (i.e., $n = 0$). The new notation hides the random variable, $R_{d,k,q}$, because *d* and *q* already appeared in the condition part of the probability, $P_{\partial,0}(R=0 \mid t \in d, q)$. It also hides the location, *k*, because we are no longer concerned with the specific location *k* of the local non-relevance, but only with the presence of *t* in *d*. The new notation hides the local relevance decision, since this decision is neither directly dependent on the document nor on the location because of the minimal context assumption. Note that the probability using the new notation is not marginal, because it is the probability of local non-relevance at certain hidden location *k* where *t* occurred in *d*. The location-invariant decision assumption implies that if a term *t* has an occurrence frequency $f(t, d)$, then there will be an $f(t, d)$ number of times that the same probability $P_{\partial,0}(R=0 \mid t \in d, q)$ appears in Equation 4.11. Using this simpler notation, we can rewrite Equation 4.11 as:

$$P_{\nabla}(R_{d,q} = 1) \propto -\sum_{t \in q \cap d} f(t,d) \log P_{\partial,0}(R = 0 \mid t \in d, q) \tag{4.14}$$

where $P_{\partial,0}(R=0 \mid t \in d, q)$ is always defined since *t* is in $q \cap d$. If Equation 4.14 is interpreted as the TF-IDF term weight, then $f(t, d)$ will be the term frequency factor. The remaining term $-\log P_{\partial,0}(R = 0 \mid t \in d, q)$ is called the query-dependent IDF (QIDF):

$$QIDF(t,q) = -\log P_{\partial,0}(R = 0 \mid t \in d, q) \tag{4.15}$$

The following assumption makes the QIDF independent of the query.

Query-Independent Non-Relevance Probability (QINRP) Assumption: The conditional probability of non-relevance, given seeing a term *t*, is the same for all queries (i.e., $P_{\partial,0}(R = 0 \mid t \in d) = P_{\partial,0}(R = 0 \mid t \in d, q) = P_{\partial,0}(R = 0 \mid t \in d, q')$) for all possible query pairs, *q* and *q*'.

Note that $P_{\partial,0}(R=0 \mid t \in d)$ is not a marginal probability. Section 4.5.2 examines the validity of the previous assumption and assesses its impact on retrieval effectiveness.

Assuming that the QINRP assumption is valid, we simplify Equation 4.14 to:

$$P_{\nabla}(R_{d,q}=1) \propto -\sum_{t \in q \cap d} f(t,d) \log P_{\partial,0}(R=0 \mid t \in d) \tag{4.16}$$

For Equation 4.16 to correspond to TF-IDF term weights, the remaining quantity (given that *t* is in the document) after taking *f*(*t*, *d*) away should be the IDF, that is,

$$-\log P_{\partial,0}(R=0 \mid t \in d) = IDF(t) \tag{4.17}$$

We do not have to consider the case when *t* is not in *d*, because: (1) *f*(*t*, *d*) is zero, and (2) *t* must have appeared in *d* according to Equation 4.16. Section 4.4.3 derives the previous equation and, therefore, establishes Equation 4.17. Section 4.3 has details about the derivation of the term-frequency factor in the literature.

4.3 Term Frequency Correspondence

This section shows that our term frequency factor in Equation 4.16 can be rendered into different term frequency factors in the literature [Salton and Buckley, 1988; Robertson and Walker, 1994] by normalizing the document length. Using the normalized version Δ(*d*) of document *d*, the probability of relevance in Equation 4.16 becomes:

$$P_{\nabla}(R_{\Delta(d),q}=1) \propto \sum_{t \in q \cap d} f(t,\Delta(d)) \times IDF(t) \tag{4.18}$$

4.3.1 Proportion Approach

The weighted Minkowski *p*-norm length [Klir and Folger, 1988] of *d* is defined as:

$$| d |_p = \sqrt[p]{\sum_w [W(w) \times f(w,d)]^p} \tag{4.19}$$

with weight $W(w)$ for term w. This weighted *p*-norm length is related to the weighted generalized mean [Dykchoff and Pedrycz, 1984] that is used as the extended Boolean disjunction [Salton et al., 1983]. The vector space model [Salton et al., 1975] uses the weighted Euclidean (i.e., $p = 2$) length and the weight of a term is its IDF. For the unweighted *p*-norm length, $W(w)$ is set to 1 for all w.

The *p*-norm length of the normalized document $\Delta(d)$ is denoted by $|\Delta(d)|_p$, which is a constant independent of *d*. In the literature, $|\Delta(d)|_p$ is the average document length Δ, for $p = 1$. Since $|\Delta(d)|_p$ is a constant, we can deduce the following property of normalized documents:

Constant Length Property: For any two normalized documents, their weighted *p*-norm lengths are the same, given a particular weighted *p*-norm.

We define the *p-norm proportion* $g_p(t, d)$ of term *t* in *d* as:

$$g_p(t,d) = \frac{f(t,d)}{| d |_p} \tag{4.20}$$

so that we specify the following assumption:

Constant *p*-Norm Proportion Assumption: Given a particular weighted *p*-norm, $g_p(t, \Delta(d)) = g_p(t,d)$ for all terms and for all documents.

Based on the previous assumption, we can deduce that:

$$f(t,\Delta(d)) = \frac{|\Delta(d)|_p \times f(t,d)}{|d|_p} \tag{4.21}$$

Subsituting Equation 4.21 into 4.18, our basic ranking formula becomes:

$$P_{\nabla}(R_{\Delta(d),q}=1) \propto \sum_{t \in q \cap d} \frac{f(t,d)}{|d|_p} \times IDF(t) \tag{4.22}$$

It is possible to normalize the query term frequency as well as using the query length but we have not pursued this aspect for clarity of presentation.

When p = 1, $|d|_1$ is the number of terms in the document d. The quantity $f(t, d) / |d|_1$ is the relative frequency estimate of the occurrence probability of term t in document d. When p tends to infinity (i.e., ∞), $|d|_\infty = \max_w\{W(w) \times f(w, d)\}$ [Dykchoff and Pedrycz, 1984]. According to the constant length property, the maximum term frequency (say, $f_{max} = |\Delta(d)|_\infty$) of all normalized documents is the same (i.e., a constant). When p tends to infinity, the previous ranking formula becomes:

$$P_{\nabla}(R_{\Delta(d),q}=1) \propto \sum_{t \in q \cap d} \frac{f(t,d)}{\max_w\{W(w) \times f(w,d)\}} \times IDF(t) \tag{4.23}$$

When $W(w) = 1$ for all w, the term frequency factor of the previous equation appears in [Baeza-Yates and Ribeiro-Neto, 1999].

We generalize the p-norm proportion approach by linearly interpolating the term frequency of the normalized document and the normalized document length as:

$$f(t,\Delta(d)) = |\Delta(d)|_p \times \left[\alpha \times \frac{f(t,d)}{|d|_p} + (1-\alpha)\right] \tag{4.24}$$

where α is the mixture parameter. This interpolation captures the intuition that a document without any query terms has small chance of being relevant to the query. This small chance is controlled by α. When α = 1.0, the previous equation becomes the normalized term frequency in Equation 4.21 as specified by the *p*-norm proportion approach. When *p* tends to infinity and α = 0.5, then the previous equation becomes the normalized term frequency factor reported by Salton and Buckley [1988].

4.3.2 Weighted Term Frequency Approach

Similar to the work by Amati and van Rijsbergen [2002], this approach uses the Laplace law of succession [Feller, 1968] to derive the weighted term frequency (e.g., [Huang et al., 2003]) as the term frequency factor of BM term weights of the Okapi system [Robertson, 1997]. This approach derives the BM term weights in a way different from their original conception [Robertson and Walker, 1994].

The basic idea is that the term frequency is weighted by a factor, $P(f(t,d) \mid R=0)$ that takes into account the probability that all occurrences of term *t* in document *d* are locally non-relevant to a query. This probability is only a weight, and it is defined in another event space. Since each occurrence of a term has a weight $P(f(t,d) \mid R=0)$, the term *t* that occurred *f*(*t*,*d*) times in *d* has a weighted term frequency $\omega(t,d)$ of $f(t,d) \times P(f(t,d) \mid R=0)$.

The weight $P(f(t,d) \mid R=0)$ is a probability that is determined by the Laplace law of succession, as follows. We assume that terms are either locally relevant (*R*=1) or non-relevant (*R*=0), corresponding to two outcomes in the Laplace law of succession [Feller, 1968]. In this way, $P(f(t,d) \mid R=0)$ is the probability that all the outcomes of *f*(*t*,*d*) occurrences of *t* are non-relevant.

$$P(f(t,d) \mid R=0) \approx \frac{1}{f(t,d)+1} \tag{4.25}$$

The weighted term frequency $\omega(t, d)$ of t in d is:

$$\omega(t,d) = f(t,d) \times P(f(t,d) \mid R=0) \approx \frac{f(t,d)}{f(t,d)+1} \tag{4.26}$$

Similarly, the weighted normalized term frequency $\omega(t, \Delta(d))$ of t in the normalized-length document $\Delta(d)$ is:

$$\omega(t,\Delta(d)) \approx \frac{f(t,\Delta(d))}{f(t,\Delta(d))+1} \tag{4.27}$$

Assuming that the constant p-norm proportion assumption is true, Equation 4.21 is substituted into the previous equation as follows:

$$\omega(t,\Delta(d)) \approx \frac{f(t,d)}{f(t,d)+\frac{|d|_p}{|\Delta(d)|_p}} \tag{4.28}$$

Replacing $f(t,\Delta(d))$ in Equation 4.18 with the previous approximation of $\omega(t,\Delta(d))$ yields the BM11-like [Robertson and Walker, 1994] formula as follows:s

$$\begin{aligned} P_{\nabla}(R_{\Delta(d),q}=1) &\propto \sum_{t \in q \cap d} \omega(t,\Delta(d)) \times IDF(t) \\ &\approx \sum_{t \in q \cap d} \frac{f(t,d) \times IDF(t)}{f(t,d)+\frac{|d|_p}{|\Delta(d)|_p}} \end{aligned} \tag{4.29}$$

The previous formula is similar to the BM11 term weight [Robertson and Walker, 1994]. First, the original BM11 uses p = 1 for measuring document lengths [Robertson and Walker, 1994]. Second, the original BM11 has an additive factor, but the highest average precision of the Okapi system is obtained when this additive factor is eliminated (i.e., k_2 = 0 in [Robertson and Walker, 1994]). Hence, the additive factor is treated as non-existent in

the original BM11 term weight. Third, we do not derive the query term frequency factor in the original BM11 term weight, for clarity of presentation. Finally, the IDF factor of the original BM25 term weight is w_4 [Robertson and Sparck Jones, 1976] for retrospective experiments and it becomes IDF_{BM} for predictive experiments as follows.

$$IDF_{BM}(t) = \log \frac{card(D) - df(t) + 0.5}{df(t) + 0.5} \tag{4.30}$$

where *card*(*D*) is the cardinality of the collection *D* (i.e., the number of documents in *D*) and *df*(*t*) is the number of document containing term *t* (i.e., document frequency of term *t*).

The BM25-like term weight [Robertson et al., 1995] is derived by linearly interpolating the original *p*-norm- and normalized *p*-norm document lengths with a mixture parameter α, as [Sparck Jones et al., 2000]:

$$f(t, \Delta(d)) = \frac{|\Delta(d)|_p \times f(t,d) \times \frac{1}{k}}{(1-\alpha)|\Delta(d)|_p + \alpha |d|_p} \tag{4.31}$$

where $k > 0$ is a constant for scaling. Substituting the previous equation into $\omega(t, \Delta(d))$, we have:

$$\omega(t, \Delta(d)) \approx \frac{f(t,d)}{f(t,d) + k \times \left((1-\alpha) + \alpha \frac{|d|_p}{|\Delta(d)|_p}\right)} \tag{4.32}$$

The BM25-like formula is obtained by substituting the previous equation into our basic ranking formula of Equation 4.18.

$$\begin{aligned} P_{\nabla}(R_{\Delta(d),q} = 1) &\propto \sum_{t \in q \cap d} \omega(t, \Delta(d)) \times IDF(t) \\ &\approx \sum_{t \in q \cap d} \frac{f(t,d) \times IDF(t)}{f(t,d) + k \times \left(1 - \alpha + \alpha \frac{|d|_p}{|\Delta(d)|_p}\right)} \end{aligned} \tag{4.33}$$

The previous formula is similar to the original BM25 term weight [Robertson et al., 1995] (Equation 2.17). First, the original BM25 has an additive factor, but it was set to zero (i.e., $k_2 = 0$) [Robertson et al., 1995]. Second, the original BM25 term weight includes some multiplicative constants (e.g., $(k_1 + 1)$ and $(k_3 + 1)$) in [Robertson et al., 1995]) that do not affect ranking because the additive factor in the original BM25 term weight has disappeared. Third, we do not derive the query term frequency factor in the original BM25 term weight, for clarity of presentation. Finally, the IDF factor of the original BM25 term weight is w_4 [Robertson and Sparck Jones, 1976] for retrospective experiments and it becomes IDF_{BM} (Equation 4.30) for predictive experiments.

4.4 Inverse Document Frequency Correspondence

This section shows that the quantity, $-\log P_{\partial,0}(R = 0 \mid t \in d)$, in Equation 4.10 can be approximated by the inverse document frequency (IDF) [Sparck Jones, 1972]:

$$IDF(t) = \log \frac{card(D)}{df(t)} \tag{4.34}$$

where $card(D)$ is the cardinality of the collection D (i.e., the number of documents in D) and $df(t)$ is the document frequency of the term t. This approximation simplifies our ranking formula to the TF-IDF term weights. We carried out an experiment using four TREC ad hoc retrieval collections and found almost no mean average precision differences between ranking using IDF (Equation 4.34) and using IDF_{BM} (Equation 4.30).

4.4.1 Basic Random Match Model

Our approach in this section regards $-\log P_{\partial,0}(R = 0 \mid t \in d)$ as a measure of the non-specificity of term usage of t found in the collection D. Non-specificity refers to the number of alternatives that one needs to select.

Usage refers to the meaning of the term *t* and the use of *t* in the context. If the term *t* occurs at two different document locations with different meanings, then the two usages of *t* are different. However, the term *t* at different document locations can have the same meaning but its usages are still different because the way the terms are used can affect the relevance of the usage. For example, the term "telescope" found in two different locations can refer to the same Hubble telescope, but one usage can be about how to repair it and the other usage can be about what it has discovered. Therefore, the number of usages of a term is at least the number of meanings that term has in the collection.

The probability $P_{\partial,0}(R=0 \mid t \in d)$ is assigned by our basic random match model of term usages. This model specifies that matching the usage of the query term and the matched document term is done in a random manner, similar to drawing a color ball from an urn [Feller, 1968] at random. In general, more than one usage can be locally relevant to the query, but we make the following assumption to simplify our modeling:

Single Locally Relevant Usage Assumption: A term *t* has one locally relevant usage for any query out of a set of possible usages of *t*.

Although this simplifying assumption is not likely to be realistic, it simplifies our basic random match model so that there is only a single parameter to estimate. If the total number of usages of term *t* is *m*(*t*), then our basic random match model specifies the probability of non-relevance, given *t*, as:

$$P_{\partial,0}(R=0 \mid t \in d) = \frac{m(t)-1}{m(t)} \tag{4.35}$$

Our basic random match model is similar to and inspired by, but not the same as, the probabilistic models based on divergence from randomness [Amati and Van Rijsbergen, 2002]. To estimate *m*(*t*), we estimate the arrival rate Λ(t), which is discussed next.

4.4.2 New-Usage Arrival-Rate Estimation

Consider a hypothetical human evaluator looking up the contexts of our query term *t*, as in Figure 1.1, and deciding the relevance of each context to this query. The middle term *t* in the context is a query term according to the query-centric assumption because contexts of non-query terms are assumed not locally relevant. The evaluator scans through the set of contexts and collects a set $B(t)$ of unique usages of *t* from the contexts. Hence, $card(B(t)) = m(t)$. A new usage of *t* is collected if it is different from the set of usages found in $B(t)$ so far. For simplicity, we assume the following:

Poisson Distributed New Term-Usage Assumption: The number of arrivals of new usages of any term in a unit-time interval follows a Poisson distribution.

It follows from the previous assumption that the arrival rate $\Lambda(t)$ of new usages of a term *t* is a constant. Note that different terms have different arrival rates of new usages.

The conventional estimation of $\Lambda(t)$ counts the number of arrivals of unique usages divided for *t* by the number of intervals. This estimation is known to be a maximum-likelihood estimator. However, the number of unique usages of a term is not the same as the total number of occurrences of this term, since occurrences of the same term with the same usage are counted only once. Therefore, someone is needed to collect the set of unique usages of a term in the collection, and this collection process is labor intensive and error prone. In addition, the manual identification of similar contexts representing similar term usages can be subjective.

To estimate $\Lambda(t)$ automatically, we regard each document as a constant unit-time interval (which suggests that document lengths should be normalized). If a term is absent in the document, then there will be no new term-usage

arrivals in the document. Therefore, we estimate $\Lambda(t)$ by equating the probability that no new term-usage arrived in the document, according to the Poisson distribution with the proportion of documents that do not contain term *t* as:

$$P_{Poisson(\Lambda(t))}(\eta(t)=0) = e^{-\Lambda(t)} = \frac{card(D) - df(t)}{card(D)} \tag{4.36}$$

where $\eta(t)$ is the number of new term-usages of *t*, and $P_{Poisson(\Lambda(t))}(.)$ is the probabililty based on the Poisson model of new-usage arrival. After some algebraic manipulation, we have an estimate of $\Lambda(t)$:

$$\Lambda(t) = \ln \frac{card(D)}{card(D) - df(t)} \tag{4.37}$$

We call the previous equation the *zero occurrence* estimate of $\Lambda(t)$. This estimate of $\Lambda(t)$ has a number of problems. First, $\Lambda(t)$ may be a biased estimate. Second, as $df(t)$ approaches $card(D)$, $\Lambda(t)$ tends to infinity. This is because the small relative-frequency counts are not reliable estimates of probabilities. Having indicated the problems with this estimate of $\Lambda(t)$, we are not aware of any theoretical alternative to estimate $\Lambda(t)$ without manually identifying the specific usage of each term occurrence. Therefore, we use this estimate of $\Lambda(t)$ assuming that $df(t)$ is not close to $card(D)$ in order to avoid singularities.

4.4.3 Expectation Approach

Let $E(.)$ be the expectation operator and $\eta(t)$ be the number of unique usages of term *t*. The expectation approach uses the conditional expected number $E(\eta(t) | \ \eta(t) > 0)$ of unique usages of term *t* in document *d*, given that *t* occurred in *d*, as an estimate of the number $m(t)$ of colored balls in an urn in our basic random match model. The conditional expectation is used because the probability, $P_{\hat{\partial},0}(R = 0 \mid t \in d)$, in Equation 4.35 is a conditional

probability where *t* is present in *d*. According to the Poisson distributed new term-usage assumption, the number of unique usages follows a Poisson distribution, so the conditional expectation $E(\eta(t)|\eta(t)>0)$ is calculated as:

$$m(t) \approx E(\eta(t) \mid \eta(t) > 0) = \frac{\Lambda(t)}{1 - e^{-\Lambda(t)}} \tag{4.38}$$

by averaging all possible numbers of new term-usage arrivals in the entire population. Although the number of new term-usage arrivals is bounded by the number of term occurrences in the given document in practice, this bound is not used because the calculated expected number of new-usage arrivals is for the population, and not for a particular document. This treatment is consistent with our minimal context assumption, where $P_{\partial,0}(R = 0 \mid t \in d)$ depends only on the term and its presence in the document, but not on the particular document d in which *t* occurred.

Using the previous calculation of $E(\eta(t)|\eta(t)>0)$, the usages of a term are considered as colored balls drawn from an urn in our basic random match model. Such an urn has $E(\eta(t)|\eta(t)>0)$ unique usages, where one of the unique usages is assumed the desired usage according to the single locally relevant usage assumption. If the usage of the term in the document is the desired usage matching the usage of *t*, then the document will be locally relevant to the query. This single local relevance occurrence becomes the document-wide relevance according to the TREC ad hoc evaluation policy [Harman, 2004]. Likewise, if the usage of the term *t* is not the usage of the matched query term, then the document location, where the query term occurred in the document, will be locally not relevant to the query. Assuming that each usage of the term *t* has equal likelihood of occurrence and using the zero occurrence estimate of $\Lambda(t)$ in Equation 4.37, the probability of local non-relevance for a document location where the query term *t* occurred is assigned:

$$P_{\partial,0}(R=0 \mid t \in d) = \frac{E(\eta(t) \mid \eta(t)>0)-1}{E(\eta(t) \mid \eta(t)>0)}$$
$$= 1 - \frac{df(t)}{card(D)\ln[\frac{card(D)}{card(D)-df(t)}]} \quad (4.39)$$

Using the aforesaid result, we define the expectation weight $W_E(.)$ as a replacement of the IDF for document ranking:

$$W_E(t) = -\log\left[1 - \frac{df(t)}{card(D)\ln[\frac{card(D)}{card(D)-df(t)}]}\right] \quad (4.40)$$

In Figure 4.1, the dotted curve shows the expectation weight given a specific IDF value. This curve shows the deviation of the IDF value from the expectation weight, since herein the IDF value is supposed to be approximating the expectation weight. In Figure 4.1, a solid straight line is drawn to serve as a reference for highlighting the deviation of IDF from the expectation weight (i.e., the circles). Notice that the IDF value begins to differ from the expectation weight when the former rises above 0.3 (using a logarithm of base 10). Later, this can be explained by deriving the IDF based on a Taylor series expansion of the expectation weight, and will be discussed later.

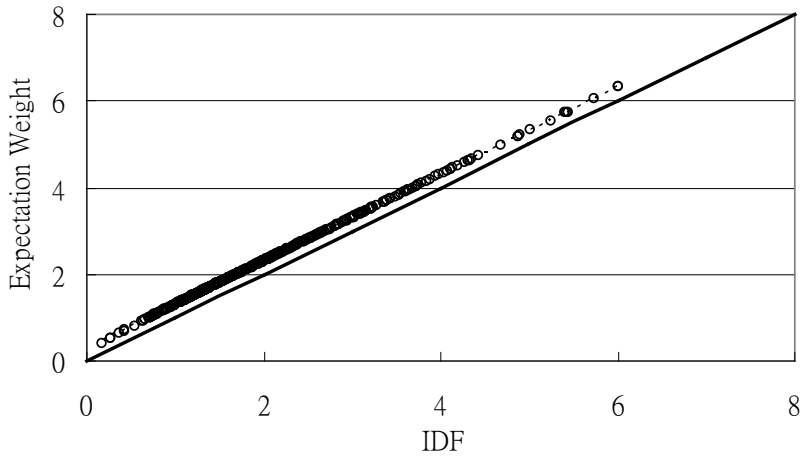

Figure 4.1: Relationship between IDF and the expectation weight. Each circle is the IDF and the corresponding expectation weight of a query term in the 200 TREC title queries (see Section 4.5.1 for details).

The circles in Figure 4.1 represent the IDF values and their corresponding expectation weights of query terms found in the set of 200 TREC title queries in TREC-2, TREC-6, TREC-7 and TREC-2005 ad hoc test collections. Notice that the spread of the expectation weights and the corresponding IDF values of these query terms are from 0.3 and above 5.0 so that most of the expectation weights are almost the same as their corresponding IDF values. We observe the difference between the expectation weight and corresponding IDF to slowly increases from 0.2 to 0.3 as the IDF value increases.

We carried out an experiment to observe if there is any impact on retrieval effectiveness using IDF as an approximation to the expectation weight (Equation 4.40). We used the title queries of TREC-2, TREC-6, TREC-7 and TREC-2005 ad hoc retrieval test collections. The details of these collections can be found in Section 4.5.1. In this experiment, the term frequency factor is based on BM11 [Robertson and Walker, 1994] which is multiplied by the IDF, or by the expectation weights, to form the term weights for ranking. We have tested the IDF_{BM} factor used in the BM11 term weight and the IDF here. Since we could not find any performance differences between them, we did not report their results here.

We measured the retrieval effectiveness of ranking based on IDF and on expectation weights using data from TREC-2, TREC-6, TREC-7 and TREC-2005 ad hoc retrieval tasks. For all test collections used in this experiment, all the performances are almost the same for ranking based on IDF and on the expectation weights, so numerical details are omitted here. The similar performance may be due to the fact that there are an equal numbers of good and bad queries to balance out the performance differences. However, we found that the MAPs of individual queries using ranking based on IDF and the corresponding expectation weights are almost the same. This is substantiated by fitting a linear regression line to the data in Figure 4.1 where the correlation is 1.00 (almost perfect regression), the

gradient is 0.9999 (which is approximately 1.0) and the regression curve crosses over the *y*-axis at 0.00003 (which is close to zero).

We suspect that there are at least two reasons why the retrieval effectiveness of individual queries are similar between ranking based on IDF and the corresponding expectation weights. First, there is almost a constant difference between the expectation weights and the IDF values. This difference is about 0.3, small compared with those large expectation weights that usually contribute most in document ranking. Second, if this approximation error of the expectation weight by the IDF value affects all the documents, this error has no impact on ranking. Such approximation errors occur when the document frequency of the query term is large. This implies that almost all the retrieved documents have this query term so that the approximation errors have little impact on ranking the retrieved documents.

In our previous experiment, the IDF is found to be a good approximation of the expectation weights in practice. This good approximation can be shown to hold mathematically. More specifically, the expectation weight in Equation 4.40 is simplified to IDF using the Taylor series:

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots \quad \text{for } -1 \le x \le 1 \tag{4.41}$$

by taking only the first term in the Taylor series expansion:

$$\begin{aligned} P_{\partial,0}(R=0 \mid t \in d) &= 1 - \frac{df(t)}{card(D)\ln[\frac{df(t)}{card(D)-df(t)} + \cdots]} \\ &\approx 1 - \frac{df(t)}{card(D)\frac{df(t)}{card(D)-df(t)}} \\ &= \frac{df(t)}{card(D)} \end{aligned} \tag{4.42}$$

Note that the above approximation (≈) and equality (=) are not distributive over each other and therefore can only be interpreted as related to the

previous derivation. The aforesaid approximation of $P_{\partial,0}(R=0 \mid t \in d)$ becomes the IDF if we take the negative logarithm of this approximation. The approximation is valid for $-1 \leq df(t)/[card(D) - df(t)] \leq 1$. Therefore, we can simplify the condition for the valid approximation to $df(t) \leq card(D) / 2$. Although the major error potentially occurs at the singularity when $df(t) = card(D)$, the quantity $P_{\partial,0}(R=0 \mid t \in d)$ tends to 1. Consequently, the minus logarithm of $P_{\partial,0}(R=0 \mid t \in d)$ tends to zero which is the same as the inverse document frequency (IDF) value for this particular case (i.e., log [*card*(D)*/card*(*D*)] for $df(t) = card(D)$). In practice, the previous experiment shows that the approximation errors (Figure 4.1) to have little impact on retrieval effectiveness performance, and this previous condition explains why IDF deviates from the expectation weight when the IDF value is larger than 0.3.

4.4.4 Clustering Approach

The expectation approach in the previous subsection shows that $P_{\partial,0}(R=0 \mid t \in d)$ can be approximated by IDF, after assuming a random match model of picking a non-relevant usage and that the new usage of a term is generated by a Poisson process. In this subsection, the clustering approach still assumes the validity of using the random match model, but does not assume that the new usages are generated by a Poisson process. In addition, it assumes that the number of new usages of *t* is equal to the number of clusters of similar contexts of *t*. These clusters are found by a novel clustering algorithm that is described first. Next we present a more general form of the random match model. Details of the experiments concerning the clustering approach are described in Section 4.5.

4.4.4.1 Context Clustering

Previous research [Lau and Luk, 1999] has identified different usage of a term by clustering the contexts where the term occurred. Results of finding

different usages of a term are encouraging, as the performance of identifying different usages is similar to human identification of various usages. This method of finding different usages of a term is based on the following assumption:

Similar-Context Similar-Usage Assumption: Terms that have similar usages tend to occur in similar document contexts.

This assumption is similar to the clustering hypothesis [Van Rijsbergen, 1975] because similar contexts have similar usages and some usages are relevant to a query.

While the results in [Lau and Luk, 1999] are obtained for Chinese data, we believe that the previous assumption is also valid to the same extent in written languages other than Chinese. This is because word sense disambiguation algorithms (e.g., [Gale et al, 1992]) also assign similar senses to a term that is in similar contexts. So, we can treat the problem of estimating the number of usages of a term as the problem of estimating the number of clusters of contexts of a term, where each cluster is assumed to correspond to a unique usage of the term, as in the previous assumption. Using the notation that $v(.)$ returns the vector representation of its argument, $|.|_2$ returns the Euclidean distance of its arguments, and $\bullet$ is the dot product of two vectors, the (cosine) similarity between contexts is computed as follows:

$$sim(v(c(d,k)), v(c(d',k'))) = \frac{v(c(d,k)) \bullet v(c(d',k'))}{|\, v(c(d,k))\,|_2 \times |\, v(c(d',k'))\,|_2} \qquad (4.43)$$

The weight of term t in the vector $v(c(d, k))$ is the standard TF-IDF term weight (i.e., $f(t, d) \times IDF(t)$).

We use a less popular clustering algorithm based on the idea of the minimum spanning tree (MST) [Zahn, 1971; van Rijsbergen, 1975]. This algorithm finds a forest, instead of a single tree, that connects all the nodes

in the graph. In our case, each node is a context and the edge weight between two nodes is the cosine similarity score between the contexts of these two nodes.

Figure 4.2 shows the major steps in finding the number of clusters. First, the similarity score of each pair of nodes is calculated. Second, these similarity scores are sorted from large to small. Iteratively, the two nodes, say *a* and *b*, of the current highest similarity score are checked as to whether they belong to any existing trees formed by the algorithm. If both nodes *a* and *b* belong to the same tree, then this tree structure will be destroyed if an edge connecting *a* and *b* is added to the tree. Hence, the edge connecting *a* and *b* is discarded. If either node *a* or *b* is connected to some existing tree, then the existing tree will be extended, with a new edge connecting *a* and *b*. If there are no trees that have nodes *a* or *b*, then *a* and *b* will form a new tree. This iterative process repeats until all nodes are connected. At the end, the algorithm returns the number of trees formed as the number of clusters found using this modified MST algorithm.

Algorithm: Modified Minimum Spanning Tree (MST) Clustering

```
Step 1  Compute the similarity scores of each pair of nodes
        (or contexts)
Step 2  Sort the similarity scores from large to small
Step 3  From the edge (a, b) with the largest similarity score to the
        smallest do
Step 4          if there is a tree that has both node a and node b then
Step 5                  goto step 3 {i.e., skip}
Step 6          if there is a tree that has node a or node b then
Step 7                  add (a, b) to the tree
Step 8          else add a new tree with a single edge (a, b)
Step 9          if all the nodes in the graph are connected then goto step 10
Step 10 Count the number of trees as the number m of clusters
Step 11 return m
```

Figure 4.2: Algorithm for the modified minimum spanning tree clustering algorithm that determines the number of clusters as the number of trees formed by the clustering algorithm.

While other clustering algorithms may be used, the proposed algorithm is a simple approach to estimate the number of clusters. Since the best estimate of $m(t)$ is difficult to obtain, the algorithm finds a simple estimate of $m(t)$. The terminating condition of this algorithm assumes that each node is

connected with at least one other node. Such a constraint may not be the case if some context (i.e., some node) of a term has a unique usage that no other contexts have. Even if this constraint is not valid, this means that the estimate of the number $m(t)$ of usages of t is less accurate. This constraint affects all terms, so errors due to violation of this constraint are compensated for, to some extent. Since there are also other kinds of errors introduced in the estimation (e.g., similarity score used), the impact of this constraint may not be significant. Experiments detailed in Section 4.5 investigate whether this clustering algorithm can make good estimates of $P_{\partial,0}(R = 0 \mid t \in d)$.

4.4.4.2 General Random match model

The general random match model method is similar to the basic random match model, except that the former does not make the single locally relevant usage assumption (see Section 4.4.1). Assuming that the similar-context similar-usage assumption is true, one cluster of similar contexts corresponds to one unique usage, and for a given term *t*, the number of different usages is the same as the number $m(t)$ of clusters of similar contexts to *t*.

For estimations, we have to make two further simplifying assumptions as follows:

Equal Probability Cluster Assumption: Each cluster of similar contexts (or each usage) is equally likely to occur.

Suppose that only $h(t, q)$ number of unique usages (or clusters of similar contexts) out of $m(t)$ is relevant to query *q*. Also, suppose that the equal probability cluster assumption is true. Then, $P_{\partial,0}(R = 0 \mid t \in d, q)$ is the number of unique usages not relevant locally to the query *q*, divided by the number of unique usages of term *t*:

$$P_{\partial,0}(R=0 \mid t \in d, q) = \frac{m(t) - h(t,q)}{m(t)} \tag{4.44}$$

because each unique usage, or each cluster of similar contexts, has an equal likelihood of occurrence according to the equal probability cluster assumption. Note that $m(t)$ is independent of the query because it is the number of possible usages. Given that Equation 4.44 is constrained by the algebraic form of Equation 4.35 for the random match model, the only variable in Equation 4.44 that needs to be dependent on the query is the number $h(t, q)$ of relevant usages to query q.

To estimate $P_{\partial,0}(R=0 \mid t \in d)$, we need to change Equation 4.44 to be independent of the query q. The variable $m(t)$ in Equation 4.44 depends on the term t, and not on q. The only variable left in Equation 4.44 is $h(t, q)$ which is dependent on q. Therefore, to make Equation 4.44 independent of q, we parameterize $h(t, q)$ by $\alpha(t)$:

Parameterized Number of Relevant Usage Assumption: For any term t, only $\alpha(t)$ number of usages (or $\alpha(t)$ number of clusters of similar contexts) is relevant to any query and $\alpha(t)$ is independent of the query.

While the preceding simplifying assumption may not be valid in practice, the assumption implies the query-independent non-relevance probability (QINRP) assumption, because Equation 4.44 becomes independent of the query when $h(t, q)$ is replaced by $\alpha(t)$. Therefore, $P_{\partial,0}(R=0 \mid t \in d)$ is estimated as follows:

$$P_{\partial,0}(R=0 \mid t \in d) = \frac{m(t) - \alpha(t)}{m(t)} \tag{4.45}$$

Note that when $\alpha(t) = 1$, the estimation of $p_{\partial,0}(\bar{r} \mid t \in d)$ using Equation 4.45 is the same as that of the basic random match model (Equation 4.35 in Section 4.4.1).

Intuitively, when a clustering algorithm only forms tight clusters, probably more than one cluster is relevant to the query and the number of clusters not relevant to the query may be scaled up accordingly. The parameter $\alpha(t)$ can be used to scale back the number of relevant clusters to unity so that the tight clustering effect of the clustering algorithm can be compensated for by $\alpha(t)$. To appreciate this scaling effect, we rewrite Equation 4.45 as follows:

$$P_{\partial,0}(R=0 \mid t \in d) = \frac{(m(t)/\alpha(t))-1}{(m(t)/\alpha(t))} \tag{4.46}$$

where $m(t)$ is scaled down to $m(t)/\alpha(t)$, and the number of clusters relevant to the query is always normalized to unity.

4.5 Clustering Approach Experiments

This section reports on the experiments of the clustering approach to estimate the quantity $-\log P_{\partial,0}(R=0 \mid t \in d)$ using the general random match model. Several reference TREC ad hoc retrieval data collections are used.

4.5.1 Set Up

We test our models with four TREC data collections (i.e., TREC-2, TERC-6, TREC-7, TREC-2005). The TREC-7 documents belong to a subset of the TREC-6 documents. Table 4.2 shows some statistics about the data collections and the topics (queries) used for the data collections. Title (short) queries are used in the experiments because they have few (i.e., one to four) query terms, similar to the lengths of Web queries. For statistical inference, we also performed various non-parametric (Wilcoxon) statistical significance tests.

Our retrieval system used the BM11 term weight [Robertson and Walker, 1994]. No pseudo-relevance feedback is used. All terms in the documents and queries are stemmed using the Porter stemming algorithm [Porter, 1980]. Stop words are removed in both documents and queries.

Table 4.2: Statistics of the collections used in the experiments.

	TREC-2	TREC-6	TREC-7	TREC-2005
Language	English	English	English	English
Topics	101-150	301-350	351-400	50 past hard topics
No. of documents	714,858	556,077	528,155	1,033,461
No. of relevant documents	11,645	4,611	4,674	6,561
Storage (GB)	3.9	3.3	3.0	5.3

4.5.2 Query-Independent Non-Relevance Probability Assumption Validation

Section 4.2.3 makes three assumptions when the context-based ranking formula in Section 4.2.2 is simplified to the basic ranking formula (Equation 4.16). One assumption, the location-invariant decision assumption, is implied by the minimal context assumption when the local relevance decision depends only on the context content, so there are only two assumptions left to validate. In this subsection, we validate the remaining assumption called query-independent non-relevance probability (QINRP) assumption. It assumes that the non-relevant conditional probability $P_{\partial,0}(R=0 \mid t \in d, q)$ depends on the term t and not on the query q because IDF is dependent on t and not on q. The significance of this assumption is that is supports the following:

(1) The minimal context assumption is mainly responsible for the performance degradation and modeling inaccuracies

(2) This assumption allows derivation of Equation 4.16 that forms the basis of the TF-IDF term weights.

(3) It gives the parameterized number of relevant usages assumption of the clustering approach (in Section 4.4.4.2), for the estimation of $P_{\partial,0}(R=0 \mid t \in d)$ using Equation 4.45.

(4) Finally, it is the focus of our subsequent experiments on Equations 4.35 and 4.45 instead of Equation 4.44.

To validate the QINRP assumption, we plot the IDF against the query-dependent IDF, namely QIDF, which is based on an estimate of $P_{\partial,0}(R=0 \mid t \in d, q)$ according to Equation 4.15. The conditional probability's relative-frequency estimate is the number of non-relevance contexts divided by the total number of contexts of *t*. The total number of contexts of *t* is the total occurrence frequency *f*(*t*, *D*) of term *t* in all the documents of the collection *D* because one occurrence of *t* corresponds to one context. The number of non-relevance contexts is deduced by subtracting *tf*(*t*, *D*) from the number of relevant contexts of *t* for query *q*. To simplify the estimation of the number of relevant contexts, we make the following simplifying assumption by Chapter 3:

Context-Training Assumption: Given a query *q*, all contexts of all the query terms of *q* in the relevant documents are relevant.

We make this simplifying assumption even though we know that not every context of a query term in a relevant document is necessarily relevant (see Figure 1.1 for instance). Using the previous assumption, the conditional probability, $P_{\partial,0}(R=0 \mid t \in d, q)$, is estimated by relative frequency counting as follows:

$$P_{\partial,0}(R=0 \mid t \in d, q) = \frac{f(t,D) - f(t,d_{REL,q})}{f(t,D)} \tag{4.47}$$

where $f(t, d_{REL,q})$ is the total occurrence frequency of *t* in all the documents that are relevant to *q* ($d_{REL,q}$). Note that the above approximation of $P_{\partial,0}(R=0 \mid t \in d, q)$ depends on the query because $f(t, d_{REL,q})$ depends on the query *q* and this approximation is retrospective because we know which document is relevant to facilitate relative frequency counting.

Figure 4.3 plots the IDF and the corresponding estimated QIDF of all query terms in the 200 TREC queries of TREC-2, TREC-6, TREC-7 and TREC-2005 ad hoc data collections. It seems that IDF is positively correlated with QIDF. We find that the exponential regression (the solid line in Figure 4.3) fits the data points with a correlation of 71.1%, which is higher than the correlations of other regression curves that we tried (i.e., linear, logarithmic and power regression curves). The multiplicative constant in the exponential regression has no impact on ranking because this multiplicative factor is factored out in the basic ranking formula in Equation 4.16. However, the exponential function cannot be factored out from Equation 4.16, so we cannot replace QIDF by IDF directly. Consequently, we validate the QINRP assumption by examining whether there are any statistically significant differences in retrieval effectiveness using ranking based on QIDF and that based on IDF for the four reference TREC data collections. In this validation, the BM11 term frequency factor is used.

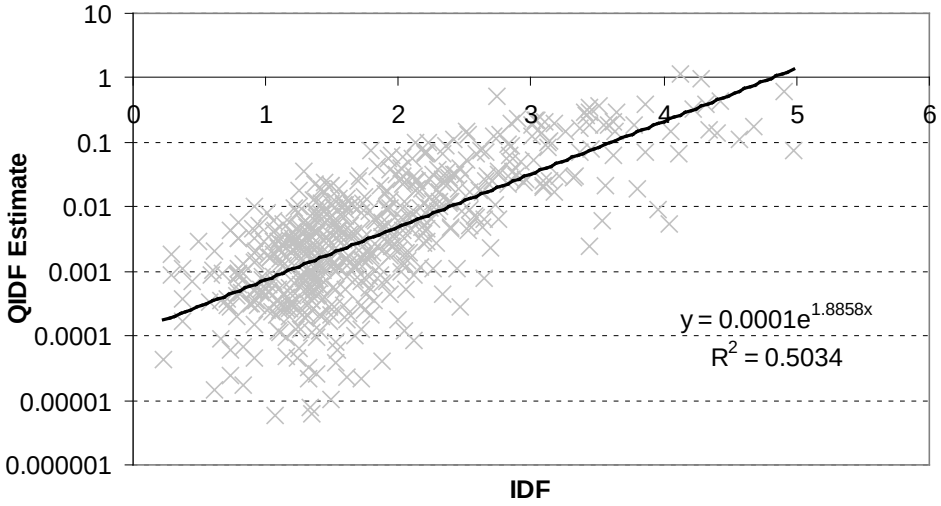

Figure 4.3: Scatter diagram of IDF- and corresponding estimated QIDF values of title query terms in the four reference TREC collections.

Table 4.3 shows the retrieval effectiveness of ranking using the basic ranking formula with an estimate of QIDF and with IDF for the quantity, $-\log P_{\partial,0}(R=0 \mid t \in d, q)$ in Equation 4.14. The mean average precision (MAP) differences between ranking using QIDF and using IDF are not statistically significant, with a *p*-value of less than 0.5347 for all four TREC

reference collections. This empirically supports the QINRP assumption, at least for the four reference TREC data collections.

Table 4.3: Comparison of traditional IDF (*IDF*(*t*)) and query-dependent IDF (*QIDF*(*t*)) performance in different TREC data collections.

	P@10		P@30		MAP		R-Precision	
TREC	IDF	QIDF	IDF	QIDF	IDF	QIDF	IDF	QIDF
2	.438	.456	.399	.404	.193	.193 (*p*=.6328)	.267	.263
6	.388	.386	.284	.281	.218	.207 (*p*=.9826)	.266	.243
7	.414	.412	.300	.296	.191	.183 (*p*=.5347)	.236	.222
2005	.358	.342	.312	.300	.175	.175 (*p*=.7654)	.239	.237

4.5.3 Estimating Number of Usages

This section examines whether the quantity $-\log P_{\partial,0}(R = 0 \mid t \in d)$ is better estimated using Equation 4.35, which is called the CLU-term weight in this section. Using the modified minimum spanning tree clustering algorithm described in Section 4.4.4, we obtain the value of $m(t)$ (i.e., number of clusters) for each of the query terms in TREC-6. However, we found that there were too many contexts for clustering and the computational resources ran out quickly. To estimate $m(t)$ with less computational resources, we systematically sampled the set of contexts of a term. If the number of contexts is more than 1000, the systematic sampling ensures that we have a sample of 1000 contexts. Otherwise, all the contexts are used.

An important parameter when clustering similar contexts is the context size, which is $2n + 1$ terms because there are n terms on each side of the term in the middle of the context. Table 4.4 shows the retrieval effectiveness using CLU weight (Equation 4.35) in comparison to IDF for TREC-6 data. The parameter n controlling the context size varies between five and one hundred, but the mean average precision (MAP) of ranking using the CLU weight differed by no more than one percentage point except for $n = 5$. This suggests that the clustering results are insensitive to context size. For efficiency, our subsequent experiments use a context size of 31 (i.e., $n = 15$).

Table 4.4: Performance of CLU(t) in TREC-6 with different context sizes used in the clustering algorithm.

	n	P@10	P@30	MAP	R-Precision
TREC-6	5	.3460	.2627	.1727	.2086
	15	.3560	.2687	.1836	.2248
	25	.3560	.2660	.1829	.2191
	50	.3520	.2640	.1842	.2210
	100	.3540	.2647	.1835	.2233

Table 4.5: Comparison of traditional *IDF*(t) and clustering approach (CLU(t)) performance in different TREC data collections.

	P@10		P@30		MAP		R-Precision	
TREC	IDF	CLU	IDF	CLU	IDF	CLU	IDF	CLU
2	.438	.370	.399	.349	.193	.165 (p=.0048)	.267	.226
6	.388	.356	.284	.268	.218	.183 (p=.0090)	.266	.224
7	.414	.362	.300	.252	.191	.167 (p=.0205)	.236	.213
2005	.358	.288	.312	.272	.175	.153 (p=.0144)	.239	.211

We evaluated the CLU weights using other TREC collections (i.e., TREC-2, TREC-7 and TREC-2005). The retrieval effectiveness of the CLU weights is shown in Table 4.5. Compared with IDF, the MAPs of the system using CLU-term weights are lower than MAPs of the same system using IDF for all the reference TREC data collections. At 99.9% confidence level, none of the collections showed significant difference between the MAP of the system using CLU weights and that using IDF. However, at 99% confidence level, TREC-2 and TREC-6 data showed a significant difference. It seems that CLU is inferior compared with IDF for these cases.

4.5.4 Optimal Performance

We estimated the optimal CLU weight which is estimated by finding the combination of machine enumerated CLU weights that produce the best MAP for a query. These CLU weights are generated by feeding an integer value between one and ten to $m(t)$ in order to calculate CLU using Equation 4.35. Let us denote $m_{opt}(t, q)$ to be the empirically identified optimal integer value of $m(t)$ for query q. The resulting estimated optimal CLU weight is now defined as the OPT weight as follows:

$$OPT(t,q) = -\log \frac{m_{opt}(t,q) - 1}{m_{opt}(t,q)} \tag{4.48}$$

If the best MAPs of the system using these machine generated OPT weights are lower than the corresponding MAPs of the same system using IDF, then we can conclude that the clustering approach to identify $m(t)$ fails because no other combination of CLU weights can produce better MAPs than IDF.

Table 4.6 shows the MAPs using the OPT weights (OPT columns) and the MAPs using the IDF. For every reference TREC collection, the MAP using the OPT weights is statistically significantly better than the MAP using the IDF at 99.9% confidence level. This suggests that some method based on the clustering approach may still have the potential to achieve MAPs as high as, if not better than, the MAPs using IDF.

Table 4.6: Performance comparison of traditional *IDF* and *OPT* weights using different TREC data collections.

	P@10		P@30		MAP		R-Precision	
TREC	IDF	OPT	IDF	OPT	IDF	OPT	IDF	OPT
2	.4240	.4860	.3840	.4426	.1863	.2261*	.2665	.2993
6	.3960	.4340	.3080	.3346	.2376	.2749*	.2844	.3063
7	.3780	.4780	.2820	.3439	.1711	.2142*	.2256	.2601
2005	.3380	.4180	.3140	.3686	.1673	.2104*	.2365	.2807

(*) – indicates that the difference in MAP between *IDF* and *OPT* is statistically significant using the Wilcoxon matched-pairs signed-ranks test with 99.9% C. I. (i.e., $p < 0.001$)

The MAP difference between the system using the OPT weights and IDF for some queries are zero and these queries have single terms. This is expected since OPT weight and IDF distinguish between terms. For single term queries, both OPT weights and IDF have no document discrimination capability. For multiple term queries, the MAP using OPT weights is higher than the corresponding MAP using IDF, although the extent of MAP improvement varies from one query to another. Therefore, we conclude that the clustering approach has some potential in achieving retrieval effectiveness as high as using IDF.

4.6 Related Work

We believe that term locations play an important role in determining relevance of documents to queries. The local relevance at certain location is thought to depend on the document-context at that location. By shrinking the context size to unity, we derive the well-known TF-IDF term weights after making some further simplifying assumptions that are similar to the derivations in the language model [Ponte and Croft, 1998], the binary independence model [Robertson and Sparck Jones, 1976] and the logistic regression model [Cooper et al., 1992; 1993]. From another perspective, these document-context based models can be thought of as an extension of existing TF-IDF term weights.

Inspired by the divergence model [Amati and van Rijsbergen, 2002] that made use of random models, we derived the inverse document frequency as the information content of the relevance decision (i.e., $-\log P_{\partial,0}(R=0 \mid t \in d)$) when a query term matches a document term. This information content is interpreted as the non-specificity of term usage. This non-specificity is derived by assuming a new usage of a term is generated by a Poisson process or by counting clusters of similar contexts as clusters of similar usages.

IDF was introduced by Sparck Jones [1972]. It is reasoned on the basis that term occurrences follow a Zipf distribution. A more theoretically motivated term weight w_4 was introduced by Robertson and Sparck Jones [1976] as a generalization of the IDF weights, and w_4 also appears in another context of improving the coordination matching scheme by Yu and Salton [1976]. Since w_4 requires statistics about relevant documents, it is used in retrospective experiments. Croft and Harper [1979] proposed the Combination Match Model (CMM) that relates w_4 with IDF under specific conditions. Later, Robertson and Walker [1997] stated a more general formula (i.e., constant + IDF by Croft and Harper [1979]). IDF is still a subject of current research [Joachims, 1997; Amati and Van Rijsbergen,

1998; Hiemstra, 1998; Papineni, 2001; Aizawa, 2003; Roelleke, 2003] where Robertson [2004] and Sparck Jones [2004] responded to recent developments on interpreting IDF. More recent works (e.g., [de Vries and Roelleke, 2005]) extend the TF-IDF term weights with more elaborate variations. Given the many variations and improvements on the original IDF, this chapter shows that the quantity, $-\log P_{\partial,0}(R=0 \mid t \in d)$, of our basic ranking formula (Equation 4.35) can be approximated by IDF [Sparck Jones, 1972] by assuming that the number of new term-usages follows a Poisson distribution.

4.7 Chapter Summary

This chapter shows that TF-IDF term weights can be interpreted as making relevance decisions. From this perspective, TF-IDF term weights are the result of simplifying our novel probabilistic retrieval model that simulates human relevance decision-making. This model distinguishes two types of relevance: one common type is the document-wide relevance that applies to the entire document, and the new type is the local relevance that only applies to certain document locations. The model makes local relevance decisions for every document location of a document and combines these local relevance decisions into a document-wide relevance decision for the document.

The significance of interpreting TF-IDF as making relevance decisions is its potential as a catalyst for different retrieval models and term weights can be interpreted by a unifying perspective: that information retrieval (IR) is about relevance decision-making. Also, our novel probabilistic retrieval model extends TF-IDF term weights to be dependent on the document locations wherein the query terms occurred. These location-dependent TF-IDF term weights (as in Equation 4.11) have the potential to form a basis for developing more elaborate retrieval models for detailed simulation of human relevance decision-making.

Our probabilistic retrieval model ranks documents on the basis of the probability of relevance. Hence, our model complies with the probability ranking principle [Robertson, 1977]. When our model is simplified to the basic ranking formula (Equation 4.16), it contains two major factors. The term frequency factor is the occurrence frequency of the query terms in the document. The remaining quantity, $-\log P_{\partial,0}(R=0 \mid t \in d)$, is shown to be IDF if we assume that: (a) a new usage of a term arrives at a constant rate following a Poisson distribution; and (b) the probability of non-relevance given term *t* is specified by our random match model of term usage. This random match model assumes that: (a) the probability of selecting a particular usage out of a set of possible usages is equally likely; and (b) a term has at most one usage that is relevant to the query.

We experimented with another approach that estimates the quantity $-\log P_{\partial,0}(R=0 \mid t \in d)$ for validating our general random match model, without assuming that the new usage of a term arrives at a constant rate following a Poisson distribution. This approach groups similar contexts into clusters and assumes that similar contexts in a cluster refer to similar usage of the term. We propose a novel modified minimum spanning tree clustering algorithm to find the number of clusters as the number of unique usages of a term. Empirically, we found that the retrieval effectiveness of this approach inferior to that using IDF. The problem is that our basic random match model assumed that only one cluster is relevant to a query, but in reality more than one cluster is relevant.

Recently, Lee et al. [2008] proposed a cluster-based re-sampling method which is effective for pseudo-relevance feedback. In their approach, they cluster the top-ranked documents and allow overlapping clusters while we do not allow overlapping clusters. We can also try other clustering methods which allow overlapping clusters in future studies.

Chapter 5

Probabilistic Document-Context Based Retrieval Model

In the previous chapter, we have shown that by assuming the minimal context assumption (i.e., shrinking the context size to unity), the document-context model can be interpreted as traditional TF-IDF term weighting models. In this chapter, we no longer assert the minimal context assumption and develop a probabilistic retrieval model based on the local relevance decisions. The probabilistic model is based on the log-odds ratio that combines two relevance decision component models which are designed to mimic human relevance decision-making. They simulate what a human evaluator does and make local relevance decisions at each document location. These local relevance decisions of a document are combined to produce the final document-wide relevance decision for the document. Retrospective experiments with our models have produced mean average precisions between 70% and 80% using various reference TREC ad hoc retrieval test collections. For relevance feedback using the top 20 ranked, judged documents, our model using fixed parameter values performs statistically significantly better than support vector machines and the highly effective, modified Markov random field model with a 90% confidence interval across different TREC collections. These results show that the proposed theory and its retrieval model are promising.

5.1 Introduction

In this chapter, we integrate the contextual information into our retrieval model using a window, called a context, which is centered on a term in the document. Essentially, a document-context is a concordance or a keyword in context (KWIC) [Kupiec et al., 1995] (see Figure 1.1).

We denote $c(d, k)$ as the context in document d at location k. The context size is specified implicitly because: (1) we want to formulate more general models and make more general statements about contexts than committing our models and statements to a particular form of a context (i.e., in this case it is a string); and (2) we want to distinguish the context size n as a parameter of the context $c(d, k)$ from d and k, which are input variables. In general, a context can be defined in many different ways.

Denote d[k] as the k-th term in document d, a context, $c(d, k)$, at location k in document d is a string of length $2n + 1$ terms:

$$d[k - n] \,.\, d[k - n + 1] \ldots d[k] \ldots d[k + n - 1]\, d[k + n]$$

where the term $d[k]$ is called the center or middle term of the context. In general, the context size may not necessarily be defined by the number of terms, and the left and right contexts do not need to have the same size. For developing elementary models, we assume all contexts are of the above form in this thesis.

Similar to Chapter 4, we make the context-based local relevance decision (CLRD) assumption which limits the quantity of information within a context of size n for making local relevance decisions. This is a simplifying assumption for developing retrieval models, and it is not intended to be true all the time. It is validated here by evaluating the retrieval effectiveness of the retrieval models (see Section 5.3). The evaluator combines the local relevance ($R_{d,k,q}$) of all valid document locations into a single document-wide relevance ($R_{d,q}$). This can be defined mathematically as:

$$R_{d,q} = C(\{R_{d,k,q} : k \in [1, |\, d \,|, k \in \mathrm{N}\}) \tag{5.1}$$

where N is the set of positive integers, $|d|$ is the document length, and $C(\cdot)$ is the general mathematical function that combines or aggregates the local relevance.

More specifically, $\partial_{d,k}(\cdot)$ denotes the local relevance decision at location k in document d. According to the CLRD assumption, the input of local relevance decisions is the context, $c(d, k)$, and the query q. The local relevance, $R_{d,k,q}$, at k in d for q is specified as follows:

$$R_{d,k,q} = \partial_{d,k}(c(d,k), q) \tag{5.2}$$

and the document-wide relevance, $R_{d,q}$ as in Chapter 4:

$$\nabla(d,q) = R_{d,q} = C(\{\partial_{d,k}(c(d,k), q) : k \in [1,|\, d\,|], k \in \mathrm{N}\}) \tag{5.3}$$

by substituting Equation 5.2 into 5.1, where $\nabla(d, q)$ is the document-wide relevance decision of document d for query q. For simplicity, local relevance and document-wide relevance take values in the closed real interval between zero and one [Robertson, 1976].

For occurrences of document terms that are not related to the query, we assume that the outcomes of the user decision for these occurrences are not locally relevant. Document terms that are related to the query may be the synonyms, hypernyms or hyponyms of the query terms, as well as the query terms themselves. Suppose that there is a function, $G(q)$, that returns terms related to q. We assume the following:

Generalized Query-Centric Assumption: For any query q and any document d relevant to q, the relevant information for q locates only in the contexts $c(d, k)$ where $k \in [1, |d|]$, and $d[k] \in G(q)$. (i.e., the relevant information locates around terms related to the topic).

When $G(q) = q$, the previous assumption is the same as the query-centric assumption in Chapter 3 (A retrospective study of a hybrid document-context based retrieval model).

The query-centric assumption and its generalized version are simplifying assumptions that are not intended to be true all the time. They substantially simplify retrieval model construction, because they enable us to ignore all the occurrences, where the documents terms are neither query terms nor their related terms. Since the query-centric assumption has been validated using various TREC ad hoc retrieval test collections in Chapter 3 that are also used in this chapter, so it is not validated in this chapter. Assuming that the generalized query-centric assumption is true, $R_{d,k,q}$ is set to zero whenever the term $t = d[k]$ at location k in document d does not belong to $G(q)$ (i.e., $\{R_{d,k,q} = 0 : d[k] \notin G(q)\}$).

For combining local relevance decisions, the DRD principle (see P. 45) in [Kong et al., 2004] is based on TREC ad hoc retrieval evaluation policy [Harman, 2004].

To interpret the DRD principle for our retrieval models, we need to specify that a document part is a context. In order to express this principle using a Boolean expression, the local or document-wide relevance is considered as the value "true" and the local or document-wide irrelevance is considered as the value "false". In this way, the document-wide relevance decision $\nabla(d, q)$ for document d given q can be specified as a Boolean expression of local relevance decisions as follows:

$$R_{d,q} = \bigvee_{k=1}^{|d|} R_{d,k,q} \tag{5.4}$$

This is based on the DRD principle by disjoining the local relevance decisions at each location of the document. The previous equation is a realization of Equation 5.1 where $C(\cdot)$ in Equation 5.1 is realized as disjunctions (i.e., $\vee$). According to the CLRD assumption, we have:

$$\nabla(d,q) = \bigvee_{k=1}^{|d|} \partial(c(d,k),q) \tag{5.5}$$

The previous equation can be applied to describe the document context model in Chapter 3 where $R_{d,k,q}$ is the normalized log-odds value of the document-context (i.e., $w(d, k)$ in Equation 3.16), and the disjunction of the previous equation is realized as the fuzzy disjunction [Dombi, 1982].

The ARD principle (see P. 45) in [Kong et al., 2004] captures the notion that the user accumulates her or his evidence until at some point that the evidence is overwhelming enough to enable her/him to make the document-wide relevance decision. This principle is not directly based on TREC ad hoc retrieval evaluation policy but intuitively it seems plausible that this principle can be applied to ad hoc retrieval evaluation policy.

We formulate the ARD principle in terms of an arithmetic expression. In this case, we need to define an aggregation operator, say denoted by $\oplus$, that aggregates the local relevance decision preferences at each location in the document d as follows:

$$R_{d,q} = \bigoplus_{k=1}^{|d|} R_{d,k,q} \tag{5.6}$$

which is a realization of Equation 5.1 where $C(\cdot)$ in Equation 5.1 is realized as the aggregation operator (i.e., $\oplus$). According to the CLRD assumption, the above becomes:

$$\nabla(d,q) = \bigoplus_{k=1}^{|d|} \partial(c(d,k),q) \tag{5.7}$$

The previous equation can be applied to describe the document context model in Chapter 3, where $R_{d,k,q}$ is the normalized log-odds value of the document-context (i.e., $w(d, k)$ in Equation 3.16), and the aggregation operator is realized as the ordered-weighted aggregation (OWA) operators (i.e., the PAICE [1984] model and the Waller and Kraft [1979]). Note that these aggregation operators are n-ary rather than binary, and these operators may not be associative.

5.2 Probabilistic Relevance Decision Model

We can generate many different probabilistic models based on the notion of (local) relevance decision-making. In this article, we have chosen to combine two relevance decision component models into a log-odds ratio that already forms the basis of existing probabilistic retrieval models (e.g., Binary Independence Model [Robertson and Sparck Jones, 1976]). The two component models are the irrelevance decision model and the aggregate relevance decision model.

Using our notation, the log-odds ratio [Fuhr, 1992] of the binary independence model (BIM) by Robertson and Sparck-Jones [1976] is

$$P(R=1 \mid d,q) \propto \log \frac{P_{\nabla}(R_{d,q}=1)}{P_{\nabla}(R_{d,q}=0)} \tag{5.8}$$

The probability of relevance in the numerator of Equation 5.8 is determined by applying the ARD principle whereas the probability of nonrelevance in the denominator of Equation 5.8 is determined by applying the DRD principle as follows. Effectively, the log-odds ratio is pooling the aggregate relevance decision principle and the disjunction relevance decision principle.

By the ARD principle, the aggregate relevance decision component model aggregates the evidence found in events at each location in the document. These pieces of evidence can be grouped into two types. One type, $E_1(d, q)$, contains events, $\{(R_{d,k,q}=1) : d[k] \in G(q)\}$, of query term or query-related term occurrences in the document, and these events are expected to be locally relevant to the query q. Another type, $E_2(d, q)$, consists of events, $\{(R_{d,k,q}=0) : d[k] \notin G(q)\}$, of non-query or non-query-related term occurrences in the document, and these events are expected to be locally non-relevant to q according to the generalized query-centric assumption. Using these two types of events, the probability of relevance in Equation 5.8 based on the ARD principle is

$$P_{\nabla}(R_{d,q}=1) = P_{\partial,n}(E_1(d,q), E_2(d,q))$$
$$= P_{\partial,n}\left(\left[\bigwedge_{t\in G(q)}\bigwedge_{k\in Loc(t,d)}(R_{d,k,q}=1)\right]\wedge\left[\bigwedge_{t\notin G(q)}\bigwedge_{k\in Loc(t,d)}(R_{d,k,q}=0)\right]\right) \quad (5.9)$$

where $Loc(t,d)$ returns the set of locations of t in d. We assume that the events in the previous equation are all mutually independent in order to simplify that equation as follows:

$$P_{\nabla}(R_{d,q}=1) = \prod_{t\in G(q)\cap d}\prod_{k\in Loc(t,d)} P_{\partial,n}(R_{d,k,q}=1)\times\prod_{t\notin G(q)}\prod_{k\in Loc(t,d)} P_{\partial,n}(R_{d,k,q}=0) \quad (5.10)$$

The above expression of combining relevance information can be considered as aggregating evidence of relevance and non-relevance information by conjunction, where the aggregation operator in Equation 5.6 is the multiplications in the previous equation. In addition, the ARD principle effectively specifies the relevance values that the local non-relevance variables can take in Equation 5.4 based on the two types of events, $E_1(d, q)$ and $E_2(d, q)$.

Assigning $P_{\partial,n}(R=1\,|\,c(d,k),t,q)$ to $P_{\partial,n}(R_{d,k,q}=1)$ after assuming that the CLRD assumption is true, the previous equation is re-arranged as follows.

$$P_{\nabla}(R_{d,q}=1)=\prod_{t\in G(q)}\prod_{k\in Loc(t,d)} P_{\partial,n}(R=1|c(d,k),t,q)\times\prod_{t\notin G(q)}\prod_{k\in Loc(t,d)} P_{\partial,n}(R=0|c(d,k),t,q) \quad (5.11)$$

For the irrelevance decision component model, the probability of nonrelevance in Equation 5.8 is derived according to the DRD principle which is formulated according to TREC ad hoc evaluation policy [Harman, 2004]. The logical form of the DRD principle is Equation 5.4 which can be rewritten as:

$$R_{d,q} = \overline{\bigwedge_{k=1}^{|d|}\overline{R}_{d,k,q}} \quad (5.12)$$

Its probabilistic version is rank equivalent to:

$$P_{\nabla}(R_{d,q}=1) \propto -\sum_{k=1}^{|d|} \log P_{\partial,n}(R_{d,k,q}=0) \tag{5.13}$$

where each $\overline{R}_{d,k,q}$ maps to $P_{\partial,n}(R_{d,k,q}=0)$. These probabilities of local nonrelevance are partitioned into two groups by the generalized query centric assumption: one group for terms in $G(q)$ and the other group for terms not in $G(q)$:

$$-\log P_{\nabla}(R_{d,q}=0) = -\sum_{t\in G(q)} \sum_{k\in Loc(t,d)} \log P_{\partial,n}(R_{d,k,q}=0) - \sum_{t\notin G(q)} \sum_{k\in Loc(t,d)} \log P_{\partial,n}(R_{d,k,q}=0) \tag{5.14}$$

Assigning $P_{\partial,n}(R=0 \mid c(d,k),t,q)$ to $P_{\partial,n}(R_{d,k,q}=0)$ after assuming that the CLRD assumption is true, the previous equation becomes

$$-\log P_{\nabla}(R_{d,q}=0) = -\sum_{t\in G(q)} \sum_{k\in Loc(t,d)} \log P_{\partial,n}(R=0 \mid c(d,k),t,q) - \sum_{t\notin G(q)} \sum_{k\in Loc(t,d)} \log P_{\partial,n}(R=0 \mid c(d,k),t,q) \tag{5.15}$$

Substituting Equations 5.11 and 5.15 into the log-odds ratio in Equation 5.8, this ratio is rank equivalent to:

$$P(R=1 \mid d,q) \propto \sum_{t\in G(q)} f(t,d) \log \frac{P_{\partial,n}(R=1 \mid t,q)}{P_{\partial,n}(R=0 \mid t,q)} + \sum_{t\in G(q)} \sum_{k\in Loc(t,d)} \log \frac{p_{\partial,n}\left(c(d,k) \mid t,q,R=1\right)}{p_{\partial,n}\left(c(d,k) \mid t,q,R=0\right)} \tag{5.16}$$

The above formula consists of two major components. The left component may be considered as the product of the term frequency and the log-odds that is similar to w_4 in [Sparck-Jones and Robertson, 1976]). In here, we assign the probability of a half to both $P_{\partial,n}(R=1 \mid t, q)$ and $P_{\partial,n}(R=0 \mid t, q)$ since we are uncertain of the relevance given only the term t and the query q. In this case, the left component in Equation 5.16 vanishes after taking the logarithm. The right component is similar to the log-odds ratio of the

document-context decision that appears in Chapter 3. The probabilities of this component are computed similar to language models where they are the product of the probabilities of the individual term occurrences. Therefore, we call our model the Binary Independence Language Model (BILM).

In this article, the query terms and their related terms (i.e., $G(q)$) are the union of (1) single query terms (i.e., $S(q)$), (2) coverage terms (i.e., $C(q)$), and (3) expansion terms (i.e., $E(q)$). That is, $G(q) = S(q) \cup C(q) \cup E(q)$. The single query term (i.e., $S(q)$) refers to the original individual query terms of the topic. The coverage term (i.e., $C(q)$) refers to the set of selected terms according to their number of occurrences with the single query terms. That is, terms occur frequently with query terms. For each topic, the coverage terms are selected by the number of occurrences of the term in the contexts of the original query terms in the relevant documents from the top X. In other words, the *coverage* of a term means the number of contexts of query terms containing the term. After the *coverage* of all terms occurred in the relevant documents from the top X are calculated, top k_{cov} terms are selected. We believe that the higher the *coverage* of a term, the higher is the correlation between the term and the query terms. Lastly, the expansion query term (i.e., $E(q)$) are the terms obtained from the relevant documents from the top X according to the relevance model (RM) [Lavrenko and Croft, 2001]. Top k_{exp} expansion terms are selected. The main difference between coverage terms and expansion terms is that coverage terms occur frequently with query terms while expansion terms may not.

Given the three sets of terms which are believed to be highly related to the topic, we define five types of contexts according to their middle term; they are (1) contexts with a query term $t \in S(q)$ in the middle, (2) contexts with a query term $t \in S(q)$ in the middle and there is another query term $s \in S(q)$ where $s \neq t$ occurs within a window size W with t, (3) contexts with a query term $t \in S(q)$ in the middle and immediately followed by another query term $s \in S(q)$ where $s \neq t$, (4) contexts with a coverage term $t \in C(q)$ in the middle and (5) contexts with an expansion term $t \in E(q)$ in the middle.

The first three types of contexts have an original query term (i.e., $S(q)$) as the middle term. The second type allows two different original query terms occur within a distance W while the third type requires the two different original query terms to occur as a phrase. To define the second and third types of contexts, we define the locations where such contexts occur as follows. Let $Loc_p(t, q, d)$ returns the set of locations of term t in document d such that there is another term $s \in S(q)$ where $s \neq t$ immediately follows t, that is, a 2-term phrase $t \cdot s$ occurred in the locations:

$$Loc_p(t,q,d) = \{k : 1 \le k \le |d|, d[k] = t, d[k+1] \in S(q), d[k+1] \neq t\} \quad (5.17)$$

Let $Loc_w(t, q, d)$ returns the set of locations of term t in document d such that there is another term $s \in S(q)$ where $s \neq t$ occurs with the term t within a distance of W:

$$Loc_w(t,q,d) = \{k : 1 \le k \le |d|, d[k] = t, d[k \pm x] \in S(q), d[k \pm x] \neq t, x \le W\} \quad (5.18)$$

From Equation 5.16, the right component used in the rank function of BILM is the log-odds ratio of the document-context decision. In practice, we can only obtain an estimate of these probabilities, and we make a weaker assumption that the estimates are only rank equivalent to the actual probabilities as follows:

$$\begin{aligned}\sum_{t \in G(q)} \sum_{k \in Loc(t,d)} \log \frac{P_{\partial,n}(c(d,k) \mid t,q,R{=}1)}{P_{\partial,n}(c(d,k) \mid t,q,R{=}0)} \propto & \sum_{t \in S(q)} \sum_{k \in Loc(t,d)} \log \frac{\left(\hat{P}_{\partial,n,S}(c(d,k) \mid t,q,R{=}1)\right)^{w_s(t)}}{\left(\hat{P}_{\partial,n,S}(c(d,k) \mid t,q,R{=}0)\right)^{w_{s'}(t)}} + \\ & \sum_{t \in S(q)} \sum_{k \in Loc_w(t,q,d)} \log \frac{\left(\hat{P}_{\partial,n,W}(c(d,k) \mid t,q,R{=}1)\right)^{w_w(t)}}{\left(\hat{P}_{\partial,n,W}(c(d,k) \mid t,q,R{=}0)\right)^{w_{w'}(t)}} + \\ & \sum_{t \in S(q)} \sum_{k \in Loc_p(t,q,d)} \log \frac{\left(\hat{P}_{\partial,n,P}(c(d,k) \mid t,q,R{=}1)\right)^{w_p(t)}}{\left(\hat{P}_{\partial,n,P}(c(d,k) \mid t,q,R{=}0)\right)^{w_{p'}(t)}} + \\ & \sum_{t \in C(q)} \sum_{k \in Loc(t,d)} \log \frac{\left(\hat{P}_{\partial,n,C}(c(d,k) \mid t,q,R{=}1)\right)^{w_c(t)}}{\left(\hat{P}_{\partial,n,C}(c(d,k) \mid t,q,R{=}0)\right)^{w_{c'}(t)}} + \\ & \sum_{t \in E(q)} \sum_{k \in Loc(t,d):T(d,k,E(q))} \log \frac{\left(\hat{P}_{\partial,n,E}(c(d,k) \mid t,q,R{=}1)\right)^{w_e(t)}}{\left(\hat{P}_{\partial,n,E}(c(d,k) \mid t,q,R{=}0)\right)^{w_{e'}(t)}}\end{aligned} \quad (5.19)$$

where $T(d, k, E(q))$ is the condition that the number of unique expansion terms in $c(d, k)$ is greater than two plus the context has a query term. When the context of an expansion term has less than three different expansion terms or does not have a query term, this context is assumed to be not related to the query, so it is ignored. Equation 5.19 is used in retrieval for ranking documents. There are 5 components on the right hand side as $G(q) = S(q) \cup C(q) \cup E(q)$. Components other that the mentioned 5 ones can also be used but experiment results show that using the 5 components can produce a better result. For $S(q)$, it is further divided into single query term, query terms occurs in proximity and query terms occurs in a phrase. As a result, the log of the 10 probabilities (each log-odds has 2 probabilities) are interpolated through the weights $w_s(t)$, $w_{s'}(t)$, $w_w(t)$, $w_{w'}(t)$, $w_p(t)$, $w_{p'}(t)$, $w_c(t)$, $w_{c'}(t)$, $w_e(t)$ and $w_{e'}(t)$. In the following discussion, we only discuss $w_s(t)$ and $w_{s'}(t)$ as others are done similarly. We added the weights where $w_s(t) > 0$ and $w_{s'}(t) > 0$ to the probabilities such that these weights can be calibrated to enhance the retrieval performance. If $w_s(t)$ equals $w_{s'}(t)$, then the estimate becomes the original maximum-likelihood estimate of the probabilities. $w_s(t)$ and $w_{s'}(t)$ control the weights of individual relevance model (i.e., the numerator of the ratio) and irrelevance model (i.e., the denominator of the ratio), respectively. We believe $w_s(t)$ and $w_{s'}(t)$ are connected to the frequency of term t in the training data such that the more occurrence of the term t in the training data, the more the importance of the context having term t as the middle term. This means that the weight is monotonically increasing with respect to the term frequency. We express this in a form similar to the BM term frequency factor [Robertson and Walker, 1994] as follows:

$$w_s(t) = w_s \times \frac{freq_s(t) + \delta_s}{freq_s(t) + \delta_s + \alpha_s} \tag{5.20}$$

$$w_{s'}(t) = w_{s'} \times \frac{freq_{s'}(t) + \delta_{s'}}{freq_{s'}(t) + \delta_{s'} + \alpha_{s'}} \tag{5.21}$$

where $w_s > 0$ and $w_{s'} > 0$ are constants which can be calibrated in the experiments. The functions $freq_s(t)$ and $freq_{s'}(t)$ are the normalized frequencies of the term t according to the occurrences of t in relevant documents and irrelevant documents respectively. The parameter δ_s in (0,1) is used for smoothing so that $w_s(t)$ does not equal to 0, and similarly for $\delta_{s'}$. The parameters α_s and α_s are used to control the corresponding curvatures or bendings of the monotonic curves, respectively. We normalize the raw frequencies $f_s(t)$ and $f_{s'}(t)$ for term t occurring in relevant and irrelevant documents, respectively, by dividing them by the corresponding maximum frequencies scaled by the parameters, c_s and $c_{s'}$, respectively, as follows:

$$freq_s(t) = \frac{f_s(t)}{\max_t\{f_s(t)\}} \times c_s \tag{5.22}$$

$$freq_{s'}(t) = \frac{f_{s'}(t)}{\max_t\{f_{s'}(t)\}} \times c_{s'} \tag{5.23}$$

Note that the parameters, c_s and $c_{s'}$, are greater than zero.

The context probabilities are the multiplication of the probabilities of individual context terms:

$$\hat{P}_{\partial,n,S}\left(c(d,k) \mid t,q,R=1\right) = \prod_{l=1}^{2n+1} \hat{P}_{\partial,n,S}\left(c(d,k)[l] \mid c[n+1]=t,q,R=1\right) \tag{5.24}$$

$$\hat{P}_{\partial,n,S}\left(c(d,k) \mid t,q,R=0\right) = \prod_{l=1}^{2n+1} \hat{P}_{\partial,n,S}\left(c(d,k)[l] \mid c[n+1]=t,q,R=0\right) \tag{5.25}$$

similarly determined for the other four types of context probabilities (i.e., $\hat{P}_{\partial,n,W}(\cdot)$, $\hat{P}_{\partial,n,P}(\cdot)$, $\hat{P}_{\partial,n,C}(\cdot)$ and $\hat{P}_{\partial,n,E}(\cdot)$) in Equation 5.19.

Using the notation that u refers to some context term $c(d, k)[l]$, let $f(u, c(d, k))$ be the raw frequency of the term u in the context $c(d,k)$. Let R_X and I_X be the top X relevant and irrelevant documents from the initial retrieval list, respectively. The conditional relative frequency estimates of u are:

$$\hat{P}_{freq,S}(u \mid t, q, R=1) = \frac{\sum_{d \in R_X} \sum_{k \in Loc(t,d)} f(u, c(d,k))}{\sum_{d \in R_X} \sum_{k \in Loc(t,d)} \sum_{v \in c(d,k)} f(v, c(d,k))} \quad (5.26)$$

$$\hat{P}_{freq,S}(u \mid t, q, R=0) = \frac{\sum_{d \in I_X} \sum_{k \in Loc(t,d)} f(u, c(d,k))}{\sum_{d \in I_X} \sum_{k \in Loc(t,d)} \sum_{v \in c(d,k)} f(v, c(d,k))} \quad (5.27)$$

The conditional relative frequency estimates of a term *u* may be zero, when the term *u* does not occur in the contexts of relevant or irrelevant documents, during re-ranking. The zero values will propagate to the context probabilities which can cause anomalies in ranking of the documents during retrieval. This is the problem of zero probability similarly found in the language modeling approach [Ponte and Croft, 1998], and smoothing [Chen and Goodman, 1996; Zhai and Lafferty, 2004] of the distribution of terms is a solution to this problem. The basic idea of smoothing is to adjust the distribution of terms so that zero probability will not assign to unseen terms. In Chapter 3, we have tested a similar model using three interpolation-based smoothing techniques namely additive smoothing [Lidstone, 1920; Johnson, 1932; Jeffreys, 1948], Jelinek-Mercer smoothing [Jelinek and Mercer, 1980; Zhai and Lafferty, 2004] and absolute discounting [Ney et al., 1994; Zhai and Lafferty, 2004] and found that the performance of the three smoothing techniques are close to each other when the parameters are set appropriately. In this chapter, we used Jelinek-Mercer smoothing:

$$\hat{P}_{\partial,n,S}(u \mid t, q, R{=}1) = \delta_{jm\_rel} \times \hat{p}_{freq,S}(u \mid t, q, R{=}1) + (1 - \delta_{jm\_rel}) \left(\frac{\sum_{d} \sum_{k \in Loc(t,d)} f(u, c(d,k))}{\sum_{d} \sum_{k \in Loc(t,d)} \sum_{v \in c(d,k)} f(v, c(d,k))} \right) \quad (5.28)$$

$$\hat{P}_{\partial,n,S}(u \mid t, q, R{=}0) = \delta_{jm\_irl} \times \hat{p}_{freq,S}(u \mid t, q, R{=}0) + (1 - \delta_{jm\_irl}) \left(\frac{\sum_{d} \sum_{k \in Loc(t,d)} f(u, c(d,k))}{\sum_{d} \sum_{k \in Loc(t,d)} \sum_{v \in c(d,k)} f(v, c(d,k))} \right) \quad (5.29)$$

where $\delta_{jm\_rel} \in [0, 1]$ and $\delta_{jm\_irl} \in [0, 1]$ are the corresponding smoothing parameters. The probabilities for the other four types of contexts are determined similarly.

Note that $\hat{P}_{\partial,n,S}(u \mid t,q,R=1)$ and $\hat{P}_{\partial,n,S}(u \mid t,q,R=0)$ are computed differently because of the different number of training data. When estimating the irrelevance probability, we make use of the bottom end documents. The *IrlBotStart* parameter controls the number of bottom end documents used. For documents ranked below *IrlBotStart*, the contexts of these documents are treated as irrelevant and add to the irrelevance model. Since the number of contexts in bottom end documents is greater than the number contexts in top X judged irrelevant documents, we weight the frequency count of terms in the contexts of bottom end documents with $IrlBotWeight \in [0,1]$ when used to estimate the irrelevance probability. As a result, the number of training data for the irrelevance model will not be too small. When the number of relevant contexts of a term $t \in G(q)$ is too small, the relative frequency estimate, $\hat{p}_{freq,S}(u \mid t,q,r)$, will be inaccurate. In order to solve this problem, we bootstrap using the relevant contexts of term $s \in G(q)$ other than term t where such contexts are similar to the contexts of t. The similarity of contexts is calculated using log-odds. This log-odds score of other relevant contexts $c(d, k)$ where $d \in R_X$, $d[k] \in G(q)$ and $d[k] \neq t$ is

$$\log P(c(d,k) \mid t,q,R=1) - \log P(c(d,k) \mid t,q,R=0) \qquad (5.30)$$

These contexts are ranked by this log-odd score, and their top T% is also considered as the contexts of t for raw frequency counting (i.e., $f(u, c(d, k))$ and $f(v, c(d, k))$) when the number of relevant contexts of a term $t \in G(q)$ is below a threshold, *relCon*.

When there is no relevant document in the top X ranked documents, the best performing parameter values are quite different from the ones when there are relevant documents. Therefore, we use two sets of parameter values: one set calibrated when there is at least a relevant document in the top X and another set calibrated when there is no relevant document in the top X.

5.3 Experiments

We performed two sets of experiments. One set is relevance feedback (RF) experiments which use the top 20, judged documents (i.e., $X = 20$) from the initial retrieval for training. Another set is retrospective experiments which uses all the judged documents for training.

5.3.1 Relevance Feedback Experiments

The proposed model is trained using the TREC-2005 ad-hoc retrieval text collection and we perform experiments on TREC-6, -7, -8 and -2005 collections using fixed, calibrated parameter values. TREC-7 and TREC-8 use the same text collection which is a subset of the TREC-6 text collection. Title queries are used in the initial retrieval which is performed using the query likelihood (QL) model [Lafferty and Zhai, 2001] of the Indri retrieval system [Strohman et al., 2004]. The results of the initial retrievals are shown in Table 5.1. Top 20 documents from the initial retrieval list are used for relevance feedback. The relevance judgements are from the TREC judgement files for the corresponding collections.

Table 5.1: Baseline results using the query likelihood (QL) model of the Indri system

TREC	P@10	MAP	R-Precision
6	.400	.247	.292
7	.454	.200	.250
8	.446	.253	.300
2005	.452	.207	.263

We compare our results with those produced by the support vector machine (SVM) using the SVM_Light package [Joachims, 1999]. After testing on TREC-2005, we use the radial basis kernel function for SVM. We also compare our results with the combination of query expansion (RM3) algorithm [Lavrenko and Croft, 2001] with Markov random field modeling (MRF) [Metzler and Croft, 2005] as in [Lease, 2008] which produced the

best results in the relevance feedback track in TREC-2008. We produced the residue result, of which the judged documents are removed from the judgement list when calculating the performance measures such as the mean average precision (MAP).

Table 5.2: RF results from our proposed model, SVM and the modified MRF

TREC	P@10			MAP			R-Precision		
	Ours	SVM	MRF	Ours	SVM	MRF	Ours	SVM	MRF
6	.423	.405	.414	$.242^{\alpha\beta}$	.216	.229	.278	.259	.264
7	.471	.468	.472	$.275^{\alpha\beta}$	$.236^{\beta}$	$.247^{\alpha}$	.295	.291	.295
8	.482	.475	.480	$.267^{\alpha\beta}$	$.228^{\beta}$	$.248^{\alpha}$	.285	.271	.274
2005	.579	.566	.576	$.340^{\alpha\beta}$	.310	.318	.357	.338	.344

α - The result compared with SVM is statistically significantly different with a 90% C.I.

β - The result compared with RM3 is statistically significantly different with a 90% C.I.

Table 5.2 shows the results of our model, SVM and the modified MRF. All the three methods use the same amount of relevance information which is the top 20 judged documents from the initial retrieval list. For our model and SVM, they use both relevant and irrelevance documents from the top 20 during training. However, for the modified MRF algorithm, only relevant documents from the top 20 are considered. From the results, our model performed significantly better than the effective SVM and the highly effective, modified MRF model with a 90% confidence interval (C.I). This is achieved for TREC-6, TREC-7 and TREC-8 test collections using fixed parameter values that are calibrated by the TREC-2005 retrieval performance. This demonstrates that our model is highly effective which is not very sensitive to the calibrated parameter values.

5.3.2 Retrospective Experiments

In the retrospective experiments, we use the whole initial retrieval list instead of using top 20 documents from the initial retrieval list for relevance feedback. Similar to Chapter 3, retrospective experiments are used to validate our retrieval models because: (a) the experiments can reveal the

potential of the models; (b) they can isolate the problems of the models from those of the parameter estimation; and (c) they can provide information about the major factors affecting the retrieval effectiveness of the models. Table 5.3 shows the results of the retrospective experiments using our model, SVM and the modified MRF model.

From the results, we can see that SVM on average outperforms our model and the MRF model in retrospective experiments for all the 4 collections tested. SVM performs statistically significantly better than MRF in all collections tested with 90% C.I. When compared with our model, only TREC-7 is statistically significantly better for SVM. Good SVM performance is probably due to the fact that SVM optimizes its performance for each query in each of the collections whereas our model and the MRF model are calibrated using TREC-2005 and are tested on the 4 collections using the same parameter values. Our model outperforms the highly effective MRF model statistically significantly in the 4 collections with a 90% C.I.

Table 5.3: Retrospective results from our proposed model, SVM and the modified MRF

TREC	P@10			MAP			R-Precision		
	Ours	SVM	MRF	Ours	SVM	MRF	Ours	SVM	MRF
6	.922	.930	.816	$.796^{\beta}$	$.813^{\beta}$	$.591^{\alpha}$	.846	.887	.549
7	.896	.986	.863	$.774^{\alpha\beta}$	$.806^{\beta}$	$.562^{\alpha}$	.816	.873	.494
8	.884	.992	.859	$.786^{\beta}$	$.793^{\beta}$	$.598^{\alpha}$	.834	.909	.527
2005	.912	.992	.875	$.793^{\beta}$	$.812^{\beta}$	$.621^{\alpha}$	.868	.924	.548

α - The result compared with SVM is statistically significantly different with a 90% C.I.

β - The result compared with RM3 is statistically significantly different with a 90% C.I.

5.4 Chapter Summary

In this chapter, we have showed the development of the probabilistic document-context based retrieval model and tested it with relevance feedback experiments and retrospective experiments. Our qualitative relevance decision model is developed into a probabilistic retrieval model

based on the log-odds ratio. For retrospective experiments using a variety of TREC English ad hoc retrieval test collections, the mean average precisions (MAPs) of these probabilistic models are between 70% and 80%. For relevance feedback using top 20 ranked, judged documents, our model is statistically significantly better than the highly effective state-of-the-art models at 90% confidence level. These provide empirical support for both our retrieval model and the proposed theory. In addition, this qualitative model is supported by the results in Chapter 3 that develops a hybrid retrieval model combining the log-odds, the extended/fuzzy Boolean model and the estimation methods in language models. In retrospective experiments, this hybrid model achieves similar MAPs as the new probabilistic retrieval model. This suggests that the qualitative model has general significance.

The main difference between the probabilistic document-context model used in this chapter and the hybrid document-context model used in Chapter 3 is that the probabilistic model assumes the generalized query-centric assumption while the hybrid model assumes the query-centric assumption. From the retrospective experimental results, the probabilistic model outperforms the hybrid model. This suggests that the generalized query-centric assumption is preferred over the query-centric assumption.

Chapter 6

A Split-List Approach for Relevance Feedback in Information Retrieval

In this chapter we present a new algorithm for relevance feedback in information retrieval. The algorithm uses document-contexts by splitting the retrieval list into sub-lists according to the query term patterns exist in the top ranked documents. Query term patterns include single query term, a pair of query terms occur in a phrase and in proximity. The document-contexts of a particular query term pattern are extracted from each of the ranked documents in the ranked retrieval list. Therefore, each sub-list contains the document-contexts having the same query term pattern. The document-contexts are then ranked in each of the sub-lists. The scores of the top ranked document-contexts for the same document are summed together to form the document score. The document with the highest score is used for feedback. The algorithm is an iterative algorithm which takes one document for feedback in each of the iterations. We experiment the algorithm using the TREC-6, -7, -8 and -2005 data collections and we simulate user feedback by the TREC relevance judgements. From the experimental results, we show that our proposed split-list algorithm is reliably better than a similar algorithm using maximal marginal relevance but without document-contexts.

6.1 Introduction

Relevance feedback is known to be effective for improving retrieval effectiveness [Rocchio, 1971; Salton and Buckley, 1990; Harman, 1992]. Relevance feedback requires user's efforts and time to judge whether a document is relevant to the user's information need. When a user judges a particular document to be irrelevant, in the user's point of view, some

efforts and time is wasted because the document provides no relevant information to the user. As a result, users are more willing to judge relevant documents than non-relevant document. In the relevance feedback process, it is better to have more relevant documents to be judged by the user. However, on the other hand, judging two very similar relevant documents also wastes user's effort and time because the information contained in the two documents is nearly the same. Judging two very similar relevant documents provides no additional relevant information to the user. Therefore, two main factors would affect the user's satisfaction in the relevance feedback process:

(a) the number of relevant documents (the more the better), and,

(b) the diversity of the documents (the more diverse the better).

In standard relevance feedback process, the user would judge documents from the top ranked ones in the initial ranked list by assuming that the top ranked documents contain more relevant information. In some cases, the top ranked documents are very similar to each other or even identical. Judging the relevance of the nearly identical documents provides no additional useful information to both the user and the retrieval system. Therefore, the set of documents used for relevance feedback may not be the top ranked ones. This is called active feedback [Shen and Zhai, 2005] in which the retrieval system actively chooses suitable documents for the user to judge for relevance.

Our proposed algorithm uses document-contexts by splitting the retrieval list into sub-lists according to the query term patterns exist in the top ranked documents. Figure 6.1 shows an example of the lists of document-contexts for the query "Hubble Telescope Achievements". Note that only the lists of document-context for single query terms are shown. By splitting the retrieval list into sub-lists, we hope that the proportion of relevant documents in a particular sub-list will be higher than that of the others. Therefore the scores of document-contexts in the particular sub-list will be higher. By that we can increase the number of relevant documents judged by the user in the relevance feedback process. Also, the set of documents being

judged by the user in the split-list approach is different from the set of top ranked documents. Hence it can increase the diversity of the documents being judged.

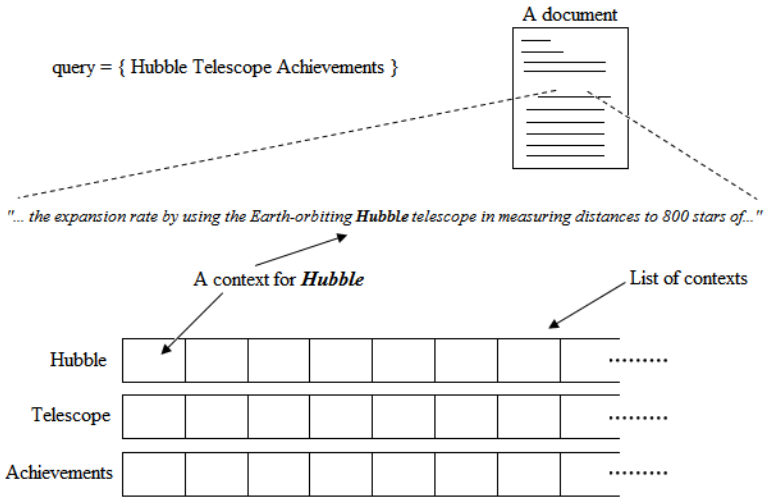

Figure 6.1: Illustration of the lists of document-contexts.

The rest of the chapter is organized as follows. In section 6.2, we outline the standard relevance feedback used in our experiments. In section 6.3, we describe some active feedback algorithms including the gapped method and cluster method in [Shen and Zhai, 2005] and the maximal marginal relevance method [Carbonell and Goldstein, 1998]. Section 6.4 describes our split-list approach using document-contexts. We show the experiment results in Section 6.5 and Section 6.6 concludes the chapter.

6.2 Standard Relevance Feedback

In this section, we outline the baseline relevance feedback algorithm used in the experiments. We use the BM25 model [Robertson and Walker, 1994] as the retrieval model throughout the relevance feedback process. Equation 2.16 shows the ranking equation of the BM25 model with k_1 and b being the model parameters. For a query q, an initial retrieval is performed using the BM25 model. In order to increase the number of top ranked relevant documents for relevance feedback, pseudo-relevance feedback (PRF) is performed after the initial retrieval.

In the PRF, the top N_{prf} documents from the initial ranked list are assumed to be relevant, denote this set of top ranked documents to be d_{prf} (i.e., $| d_{prf} | = N_{prf}$). Terms are extracted from d_{prf} for query expansion. The terms in the d_{prf} are ranked using the following formula:

$$Score_{prf}(t) = \frac{f(t, d_{prf})}{f(t, d_{prf}) + 1} \times \log \frac{card(D) - df(t) + 0.5}{df(t) + 0.5} \qquad (6.1)$$

where t is a particular term occurs in d_{prf}, $f(t, d_{prf})$ is the occurrence frequency of t in the set of assumed relevant documents d_{prf}, D is the collection of documents, $card(D)$ is the cardinality of D which is the number of documents in D and $df(t)$ is the number of documents in D containing t. The is actually the TF-IDF weight of term t.

After all the terms in d_{prf} are ranked using Equation 6.1, top K_{prf} terms are extracted. Denote the set of extracted terms to be q_{e_prf} such that $| q_{e\_prf} | = K_{prf}$. The scores of the K_{prf} terms are normalized so that they can combine with the original query q to prevent shifting the topic. The combined query q_{prf} is the union of q and q_{e_prf} with the weight of each term being:

$$w(t, q_{prf}) = \alpha_{prf} \frac{f(t, q)}{\sum_{u \in q} f(u, q)} + (1 - \alpha_{prf}) \frac{Score_{prf}(t)}{\sum_{u \in q_{e\_prf}} Score_{prf}(u)} \qquad (6.2)$$

where $w(t, q_{prf})$ is the weight of the term t in the expanded query q_{prf}, $f(t, q)$ is the occurrence frequency of t in the original query q and $\alpha_{prf} \in [0, 1]$is a mixture parameter controlling the weights of q and q_{e_prf}. When $\alpha_{prf} = 1$, it is the same as the initial retrieval using the original query.

A second retrieval is performed using the BM25 model with the expanded query q_{prf}. Top N_{rf} documents are judged by the user. In our experiments, we simulate user relevance feedback using the TREC relevance judgement files. Denote d_{rf_rel} to be the set of judged relevant documents with size N_{rel} and

`d_{rf_irl} to be the set of judged non-relevant documents with size N_{irl} such that $N_{rel} + N_{irl} = N_{rf}$. Similar to PRF, terms in d_{rf_rel} and d_{rf_irl} are scored using

$$Score_{rf\_rel}(t) = \frac{f(t, d_{rf\_rel})}{f(t, d_{rf\_rel}) + 1} \times \log \frac{\left(\frac{r(t) + 0.5}{N_{rel} - r(t) + 0.5} \right)}{\left(\frac{df(t) - r(t) + 0.5}{card(D) - N_{rel} - df(t) + r(t) + 0.5} \right)} \quad (6.3)$$

$$Score_{rf\_irl}(t) = \frac{f(t, d_{rf\_irl})}{f(t, d_{rf\_irl}) + 1} \times \log \frac{\left(\frac{i(t) + 0.5}{N_{irl} - i(t) + 0.5} \right)}{\left(\frac{df(t) - i(t) + 0.5}{card(D) - N_{irl} - df(t) + i(t) + 0.5} \right)} \quad (6.4)$$

respectively where $f(t, d_{rf\_rel})$ is the occurrence frequency of term t in the set of judged relevant documents d_{rf_rel}, $r(t)$ is the number of documents in d_{rf_rel} containing t, $f(t, d_{rf\_irl})$ is the occurrence frequency of term t in the set of judged non-relevant documents d_{rf_irl} and $i(t)$ is the number of documents in d_{rf_irl} containing t. The second term in Equation 6.3 is the w_4 weight [Robertson and Sparck Jones, 1976] in Equation 2.11. IDF is not used here because we now have some relevance information.

Terms in the set of judged relevant documents (d_{rf_rel}) and judged non-relevant documents (d_{rf_irl}) are ranked by Equations 6.3 and 6.4 respectively. Top K_{rf_rel} terms are extracted from the judged relevant documents ($q_{e\_rf\_rel}$) while top K_{rf_irl} terms are extracted from the judged non-relevant documents ($q_{e\_rf\_irl}$). Denote q_{e_rf} to be the union of the two sets of terms with the weights of terms being:

$$w(t, q_{e\_rf}) = \beta_{rf} \frac{Score_{rf\_rel}(t)}{\sum_{u \in q_{e\_rf\_rel}} Score_{rf\_rel}(u)} - (1 - \beta_{rf}) \frac{Score_{rf\_irl}(t)}{\sum_{u \in q_{e\_rf\_irl}} Score_{rf\_irl}(u)} \quad (6.5)$$

where $\beta_{rf} \in [0, 1]$ is the mixture parameter controlling the weights of extracted relevant terms and non-relevant terms. Note that the range of $w(t, q_{r\_rf})$ is between -1 and 1. If a term has a weight less than 0, it is used to decrease the document score in the final retrieval.

Finally, q_{rf} is the union of the terms from the original query q and the extracted terms q_{e_rf} with the weights of the terms:

$$w(t, q_{rf}) = \alpha_{rf} \frac{f(t,q)}{\sum_{u \in q} f(u,q)} + (1 - \alpha_{rf}) w(t, q_{e\_rf}) \tag{6.6}$$

where $\alpha_{rf} \in [0, 1]$. A final retrieval is performed using the BM25 model with q_{rf}. Figure 6.2 describes the flow in our standard relevance feedback experiment. In active feedback experiments, changes are made in the RF block in Figure 6.2 such that the judged documents are not the top ranked ones.

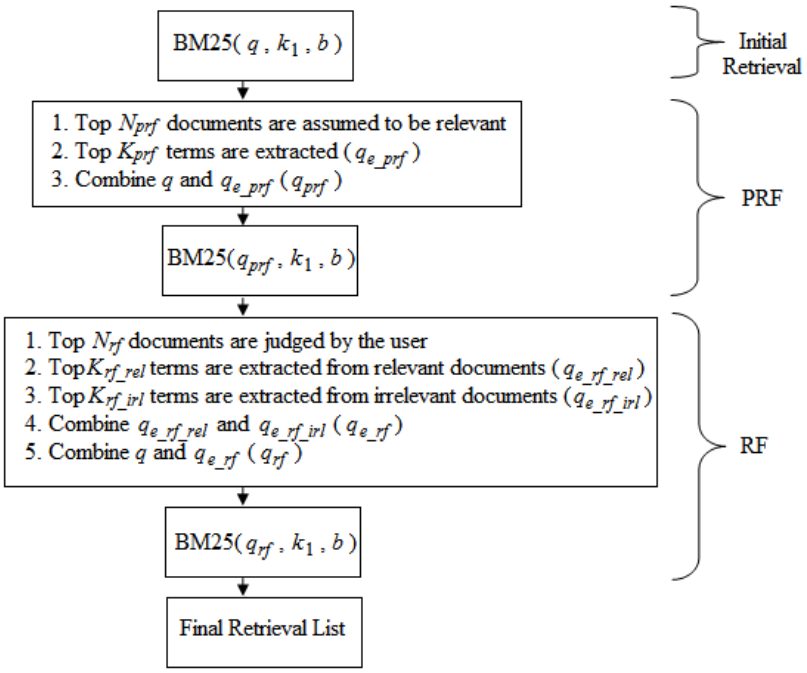

Figure 6.2: The flow of standard relevance feedback in our experiments.

6.3 Related Work

In this section we review some active feedback algorithms which do not use top N_{rf} ranked documents for relevance feedback. By modelling active feedback using the risk minimization framework for retrieval [Lafferty and Zhai, 2001], Shen and Zhai [2005] formalized the active feedback problem

as a decision making problem and experimented two active feedback methods. One is Gapped-Top-N_{rf} and the other one is N_{rf}-Cluster-Centroid method. Maximal marginal relevance (MMR) [Carbonell and Goldstein, 1998] is also described in this section.

Gapped-Top-N_{rf}

In the Gapped-Top-N_{rf} method, instead of judging the top N_{rf} ranked documents, a gap of G documents is introduced between two judged documents. As a result, the i-th judged document is ranked at $i+(i-1)G$. For example, if $G = 2$, the set of judged documents will have rank numbers 1, 4, 7,…, $N_{rf}+(N_{rf}-1)2$ in the retrieval list. With $G = 0$, the Gapped-Top-N_{rf} is essentially the standard method using top N_{rf} ranked documents (Figure 6.2). The Gapped-Top-N_{rf} method can be thought of clustering the top $(G+1)N_{rf}$ ranked documents in the retrieval list based on their relevance scores such that the first cluster contains the first $G+1$ documents, the second cluster contains the next $G+1$ documents, etc. The document with the highest relevance score in each of the clusters is used for relevance feedback. It tries to capture diversity of documents by skipping documents with little difference in their relevance scores. Figure 6.3 shows the flow of the Gapped-Top-N_{rf} method used in our experiments.

N_{rf}-Cluster-Centroid

In order to directly capture diversity, explicit clustering is performed among the top N_{cc} documents. The top N_{cc} ranked documents are clustered into N_{rf} clusters and a representative document in each of the clusters is selected to be judged by the user. In [Shen and Zhai, 2005], the K-Medoid clustering algorithm [Kaufman and Rousseeuw, 1990] is used to cluster the top N_{cc} documents and the distance function used is the J-Divergence [Lin, 1991]. The clustering algorithm tries to group the documents into clusters such that documents within a cluster are similar to each other while documents belong to different clusters are dissimilar to each other. Similar to K-Means clustering algorithm [MacQueen, 1967], the K-Medoid clustering is a non-

hierarchical clustering algorithm which minimizes the distance between documents in the clusters. The K-Medoid clustering algorithm is less sensitive to outliers to K-Means clustering algorithm. Figure 6.4 shows the details of the K-Medoid clustering algorithm. Note that when $N_{cc} = N_{rf}$, the K-Medoid clustering is the same as the standard relevance feedback algorithm (Figure 6.2).

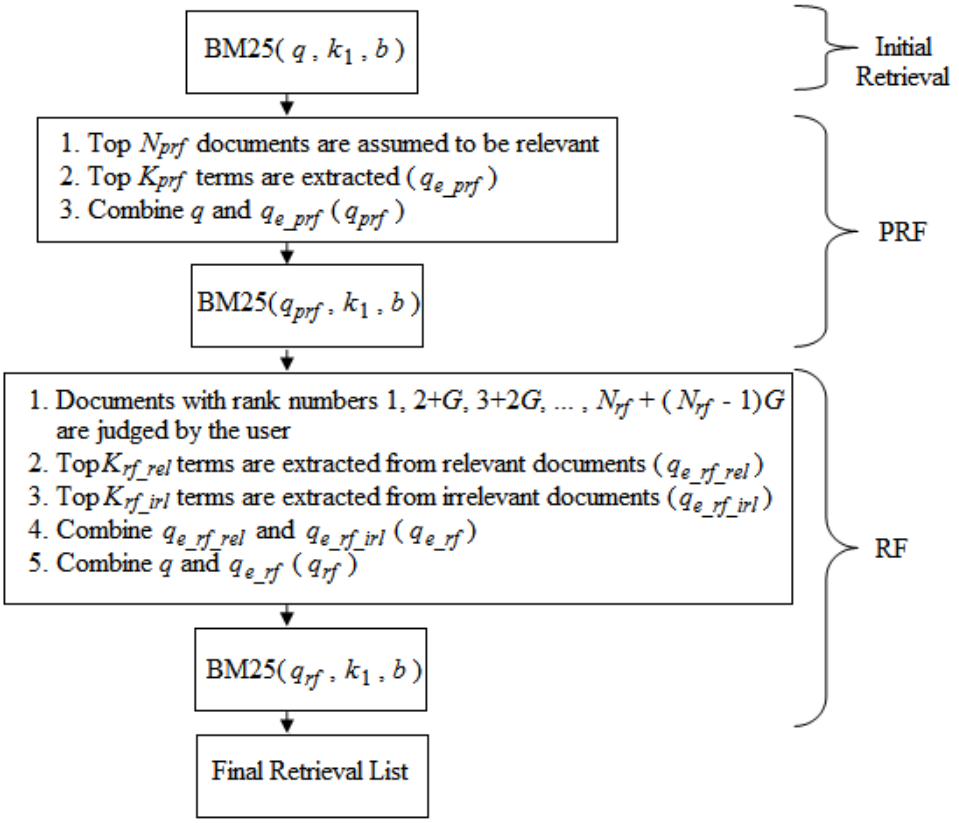

Figure 6.3: The flow of Gapped-Top-N_{rf} method in our experiments.

Algorithm K-Medoid Clustering

1. Randomly select N_{rf} of the N_{cc} documents as the medoids
2. Associate each non-medoid document to its closest medoid
3. Compute total distance which is the sum of distances from all documents to their medoids
4. For each medoid document d_m
5. For each non-medoid document d_o
6. Swap d_o and d_m and compute the new total distance
7. Select the medoids with the smallest total distance
8. Repeat Steps 4 to 8 until there is no change in the medoids

Figure 6.4: Algorithm for K-Medoid clustering

For a pair of documents d_i and d_j, the KL-Divergence [Kullback, 1968] measures the difference between the two documents by considering their underlying probability distributions:

$$D_{KL}(d_i \| d_j) = \sum_t P(t \mid \theta_i) \log \frac{P(t \mid \theta_i)}{P(t \mid \theta_j)} \tag{6.7}$$

where θ_i is the underlying language model for the document d_i which is a probability distribution defining the probability of seeing a certain term t:

$$P(t \mid \theta_i) = \alpha_{LM,d_i} \frac{f(t, d_i)}{|d_i|} + (1 - \alpha_{LM,d_i}) \frac{f(t, D)}{|D|} \tag{6.8}$$

where $f(t, d_i)$ is the occurrence frequency of term t in the document d_i, $|d_i|$ is the length of d_i, $f(t, D)$ is the occurrence frequency of t in the collection D, $|D|$ is the collection length which is the sum of all document lengths and $\alpha_{LM,d_i} \in [0,1]$ is the smoothing parameter used for mixing the document frequency with the collection frequency in order to avoid zero probability. Dirichlet smoothing is a common method for smoothing:

$$\alpha_{LM,d_i} = \frac{|d_i|}{|d_i| + \mu} \tag{6.9}$$

where $\mu > 0$ is a constant.

The KL-Divergence measure is non-symmetric (i.e., $D_{KL}(d_i \| d_j) \neq D_{KL}(d_j \| d_i)$). To obtain a symmetric measure, the J-Divergence [Lin, 1991] is defined as follows:

$$\begin{aligned} D_J(d_i \| d_j) &= D_{KL}(d_i \| d_j) + D_{KL}(d_j \| d_i) \\ &= \sum_t P(t \mid \theta_i) \log \frac{P(t \mid \theta_i)}{P(t \mid \theta_j)} + \sum_t P(t \mid \theta_j) \log \frac{P(t \mid \theta_j)}{P(t \mid \theta_i)} \\ &= \sum_t \left(P(t \mid \theta_i) - P(t \mid \theta_j)\right) \times \log \frac{P(t \mid \theta_i)}{P(t \mid \theta_j)} \end{aligned} \tag{6.10}$$

The J-Divergence is used as the distance function in the K-Medoid clustering algorithm. Figure 6.5 shows the flow of the N_{rf}-Cluster Centroid method used in our experiments.

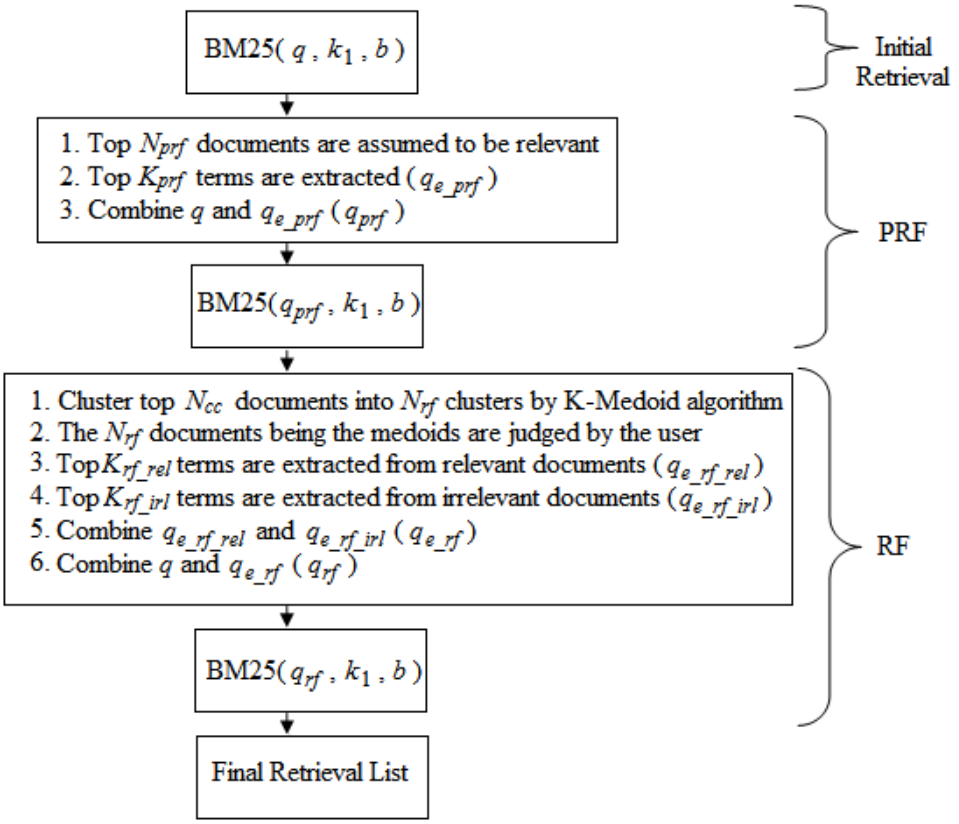

Figure 6.5: The flow of N_{rf}-Cluster-Centroid method in our experiments.

Maximal Marginal Relevance (MMR)

In 1998, Carbonell and Goldstein [1998] introduced the maximal marginal relevance. It is an iterative algorithm which selects a document d_i in each of the iterations by optimizing:

$$MMR(J,q) = \arg\max_{d_i \notin J}\left[\lambda_{MMR} Sim_1(d_i,q) - (1-\lambda_{MMR}) \max_{d_j \in J} Sim_2(d_i,d_j)\right] \quad (6.11)$$

where J is the set of currently judged documents in the relevance feedback process, q is a query, $Sim_1(d_i, q)$ is a similarity measure (relevance score)

given by a retrieval model, $Sim_2(d_i, d_j)$ is a similarity measure between two documents d_i and d_j, finally, $\lambda_{MMR} \in [0, 1]$ controls the weights of Sim_1 and Sim_2. The document selected by the MMR is said to have high "marginal relevance" which means it is both relevant to the query q (Sim_1 is high) and contains the minimal similarity to previously judged documents ($\max\{Sim_2\}$ is low). In our experiments, as we are using BM25 model for ranking the documents, $Sim_1(d_i, q)$ is the score returned by the BM25 model (Equation 2.16):

$$Sim_1(d_i, q) = \sum_{t \in d_i \cap q} \frac{f(t,q) \times f(t,d_i) \times (k_1 + 1)}{f(t,d_i) + k_1 \times \left(1 - b + b \times \frac{|d_i|}{\Delta}\right)} \log \frac{card(D) - df(t) + 0.5}{df(t) + 0.5} \quad (6.12)$$

For $Sim_2(d_i, d_j)$, in our experiments, we use the cosine similarity between the two documents d_i and d_j:

$$Sim_2(d_i, d_j) = \frac{\vec{d}_i \bullet \vec{d}_j}{\left|\vec{d}_i\right|_2 \times \left|\vec{d}_j\right|_2} \quad (6.13)$$

where $\vec{d}_i$ is the vector representation of d_i, $\vec{d}_i \bullet \vec{d}_j$ is the dot-product of the two document vectors, $\left|\vec{d}_i\right|_2$ is the Euclidean length of the document vector $\vec{d}_i$, similarly for d_j. The weight of a term t in the document vector $\vec{d}_i$ is given by:

$$w(t, \vec{d}_i) = \frac{f(t,d_i)}{f(t,d_i) + 1} \times \sqrt{\log \frac{card(D) - df(t) + 0.5}{df(t) + 0.5}} \quad (6.14)$$

This is similar to the standard TF-IDF term weight but using the square root of the IDF factor. The square root of the IDF factor is used because it is found to perform better [Dang et al., 2006] as it will multiply itself in the cosine similarity in Equation 6.13.

Figure 6.6 shows the flow of the MMR algorithm used in our experiments. The MMR algorithm is an iterative algorithm which takes one document for relevance judgement in each of the iterations. The first document to be judged is always the one ranked the first in the retrieval list. Note that the values of Sim_1 in Equation 6.11 are unchanged for all documents throughout the iteration process. They are actually the scores from the BM25 model using in the PRF process using the query q_{prf}. Therefore, when $\lambda_{MMR} = 1$, the MMR algorithm is essentially the standard relevance feedback which N_{rf} top ranked documents are judged by the user (Figure 6.2).

We can also re-rank the top N_{mmr} documents during each of the iterations using the available relevance information. The re-rank is done using query expansion from the judged relevant and non-relevant documents similar to the Equations 6.3 – 6.6. Instead of using the parameters K_{rf_rel}, K_{rf_irl}, α_{rf} and β_{rf} (see Section 6.2), a different set of parameters (K_{mmr_rel}, K_{mmr_irl}, α_{mmr} and β_{mmr}) is used. That is, a retrieval using the BM25 model is performed in each of the iterations with a modified query and the retrieval is done on the top N_{mmr} documents only (i.e., re-ranking top N_{mmr} documents). After the re-rank of the top N_{mmr} documents in each of the iterations, the Sim_1 scores of the documents in Equation 6.11 are changed. This is because we have more relevance information (i.e., one judged document) after each of the iterations such that the modified queries for each of iterations are different. As a result, $\lambda_{MMR} = 1$ will not produce the same result as the standard relevance feedback (Figure 6.2). Figure 6.7 shows the flow of the MMR-Rerank algorithm which is very similar to Figure 6.6 but with a re-ranking step in each of the iterations. Since our split-list approach is also an algorithm with a re-ranking step, we mainly compare our results with those from the MMR-Rerank algorithm.

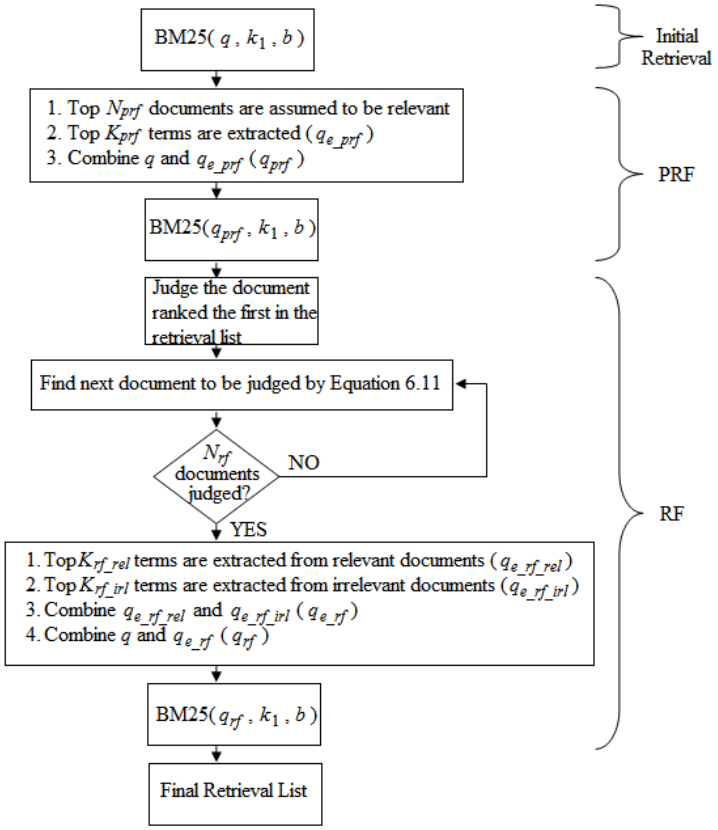

Figure 6.6: The flow of the MMR method in our experiments.

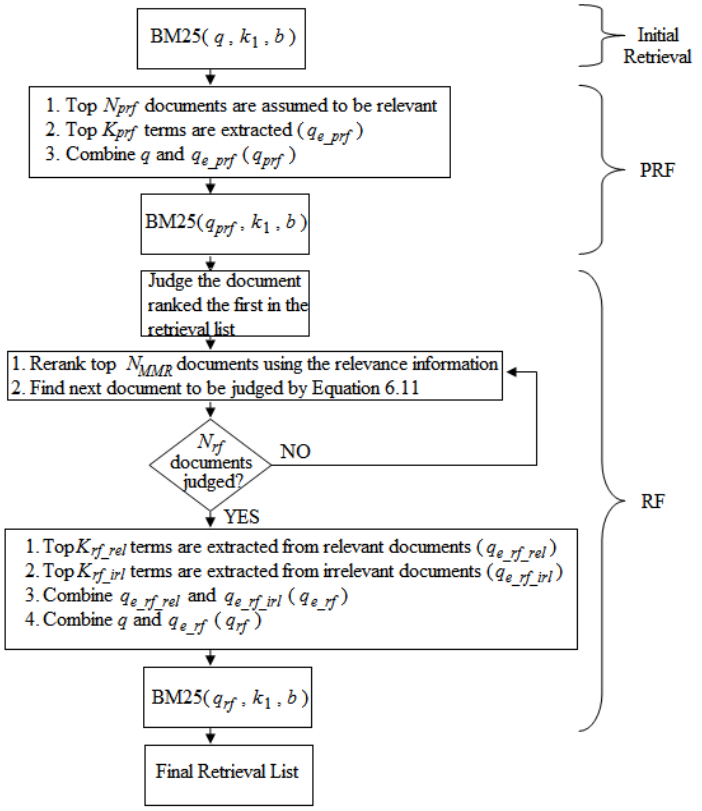

Figure 6.7: The flow of the MMR-Rerank method in our experiments.

6.4 Split-List Approach to Relevance Feedback

In this section we describe the algorithm of the proposed split-list approach to relevance feedback. The algorithm uses document-contexts by splitting the retrieval list into sub-lists according to the query term patterns exist in the top ranked documents. Query term patterns include single query term, a pair of query terms occur in a phrase and in proximity. The document-contexts of a particular query term pattern are extracted from each of the ranked documents in the ranked retrieval list. Therefore, each sub-list contains the document-contexts having the same query term pattern. The document-contexts are then ranked in each of the sub-lists. Figure 6.1 shows an example of the lists of document-contexts for the query "Hubble Telescope Achievement", only the lists of single query term are shown. The scores of the top ranked document-contexts for the same document are summed together to form the document score. The document with the highest score is used for feedback. Similar to the MMR-Rerank algorithm discussed in the previous section, our split-list algorithm is an iterative one which takes one document for relevance judgement in each of the iterations until N_{rf} documents are judged. Unlike MMR-Rerank, in each of the iterations, we re-rank the document-contexts in each of the sub-lists instead of re-ranking the documents.

Similar to previous chapters, define $d[k]$ to be the term occurs at the k-th position in the document d such that $k \in [1, |d|]$, $c(d, k)$ is the context of $d[k]$ such that it contains $2n+1$ terms which are the terms surrounding $d[k]$:

$$c(d,k) \equiv \{d[k-n],...,d[k-1],d[k],d[k+1],...,d[k+n]\} \qquad (6.15)$$

In the case where $k \leq n$ (i.e., at the beginning of a document), we do not have enough terms on the left hand side of $d[k]$. We then take $(n-k+1)$ more terms on the right hand side to make sure $2n+1$ terms are considered in a context. A similar trick is done when $k > (|d|-n)$ (i.e., at the end of a

document) which we take (n+k-|d|) more terms on the left hand side. Stop words are removed from all the documents in our experiments.

For a query q with s distinct terms $\{q_1, q_2, \ldots, q_s\}$, we have s lists of document-contexts which consider single query term occurrence, (s-1) lists of document-contexts which consider a pair of query terms occur in a phrase in the same order as the query and ${}_sC_2$ lists of document-contexts which consider a pair of query terms occur in proximity with a window size w. As a result, there is a total number of ($2s$+${}_sC_2$-1) lists of document-contexts for a query having s distinct terms (Figure 6.8).

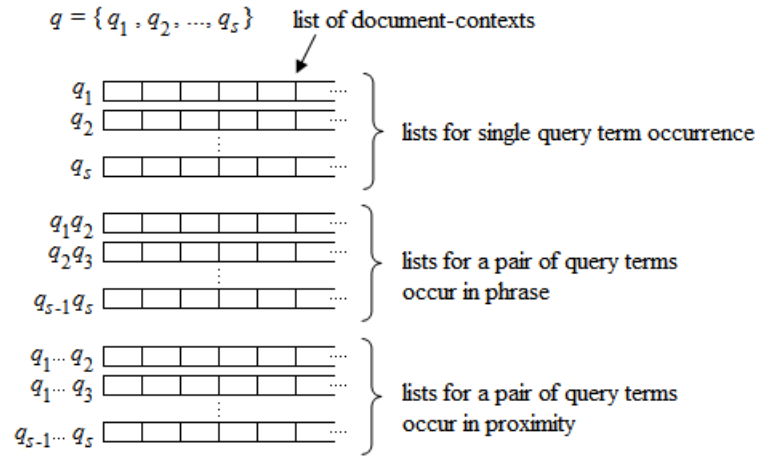

Figure 6.8: The lists of document-contexts for a query with s terms.

We extract document-contexts from the top N_{split} ranked document in the retrieval list. These N_{split} documents are scanned using a sliding window with size $2n$+1. The set of contexts $\{c(d, k) : d[k] = q_i, 1 \le i \le s\}$ are inserted to the list of q_i for single term occurrence. The set of contexts $\{c(d, k) : d[k] = q_i, d[k+1] = q_{i+1}, 1 \le i < s, \}$ are inserted to the list of q_i q_{i+1} for query terms occur in a phrase. The set of contexts $\{c(d, k) : d[k] = q_i, d[k+p] = q_j, 1 \le i \le s, 1 \le j \le s, i \ne j, p \le w\}$ are inserted to the list of $q_i \ldots q_j$ for query terms occur in proximity with distance less than w.

After all the document-contexts are inserted to the corresponding lists, we re-rank the contexts in each of the lists using the available relevance information. We only discuss the ranking for the list of q_1 because the

ranking for other lists are done similarly. We define the score of a context $c(d, k)$ in the list of q_1 being the probability of relevance of the context using the log-odds:

$$\begin{aligned} P(R=1 \mid q_1, c(d,k)) &\overset{rank}{=} \frac{P(R=1 \mid q_1, c(d,k))}{P(R=0 \mid q_1, c(d,k))} \\ &\overset{rank}{=} \frac{P(c(d,k) \mid R=1, q_1)}{P(c(d,k) \mid R=0, q_1)} \\ &\overset{rank}{=} \sum_{p=1}^{2n+1} \log \frac{P(c(d,k)[p] \mid R=1, q_1)}{P(c(d,k)[p] \mid R=0, q_1)} \end{aligned} \quad (6.16)$$

where R is the binary relevance variable which R=1 means relevant and R=0 means non-relevant, $c(d, k)[p]$ is the term at the p-th position in the context $c(d, k)$. Denote d_{split_rel} to be the set of judged relevant documents and d_{split_irl} to be the set of judged irrelevant documents. For a term t:

$$P(t \mid R=1, q_1) = \alpha_{s\_rel} \frac{\sum_{d \in d_{split\_rel}} \sum_{k:d[k]=q_1} f(t, c(d,k))}{\sum_u \sum_{d \in d_{split\_rel}} \sum_{k:d[k]=q_1} f(u, c(d,k))} + (1-\alpha_{s\_rel}) \frac{f(t,D)}{|D|} \quad (6.17)$$

$$P(t \mid R=0, q_1) = \alpha_{s\_irl} \frac{\sum_{d \in d_{split\_irl}} \sum_{k:d[k]=q_1} f(t, c(d,k))}{\sum_u \sum_{d \in d_{split\_irl}} \sum_{k:d[k]=q_1} f(u, c(d,k))} + (1-\alpha_{s\_irl}) \frac{f(t,D)}{|D|} \quad (6.18)$$

where $f(t, c(d,k))$ is the occurrence frequency of the term t in the context $c(d,k)$, d_{split_rel} is the set of judged relevant documents, d_{split_irl} is the set of judged non-relevant documents, $\alpha_{s\_rel} \in [0, 1]$ and $\alpha_{s\_irl} \in [0, 1]$ are smoothing parameters similar to $\alpha_{LM,\ di}$ in Equation 6.8. After the scoring of the contexts using Equation 6.16, they are ranked by the descending order of the scores in the list. We rank the contexts in each of the lists.

After the contexts are ranked, the top N_c ranked contexts in each of the lists are used to form the document scores. It is intuitive to assign weights to different lists, for example, the weight $w_l(q_1)$ of the list of q_1 is:

$$w_l(q_1) = \frac{\sum_{d \in d_{split\_rel}} f(q_1, d) + \alpha_l}{\sum_{d \in d_{split\_rel}} f(q_1, d) + \alpha_l + \delta_l} \tag{6.19}$$

where $\alpha_l > 0$ and $\delta_l > 0$ are constants. The weights for other lists are computed similarly. Equations 6.16 and 6.19 are multiplied together when combining context scores.

The top N_c ranked contexts in each of the lists are extracted and scores of contexts are summed together for contexts belonging to the same document. The document with the highest score is used for relevance feedback. If N_{rf} documents are judged, the iterative process ends. Figure 6.9 shows the flow of the split-list approach algorithm in our experiments.

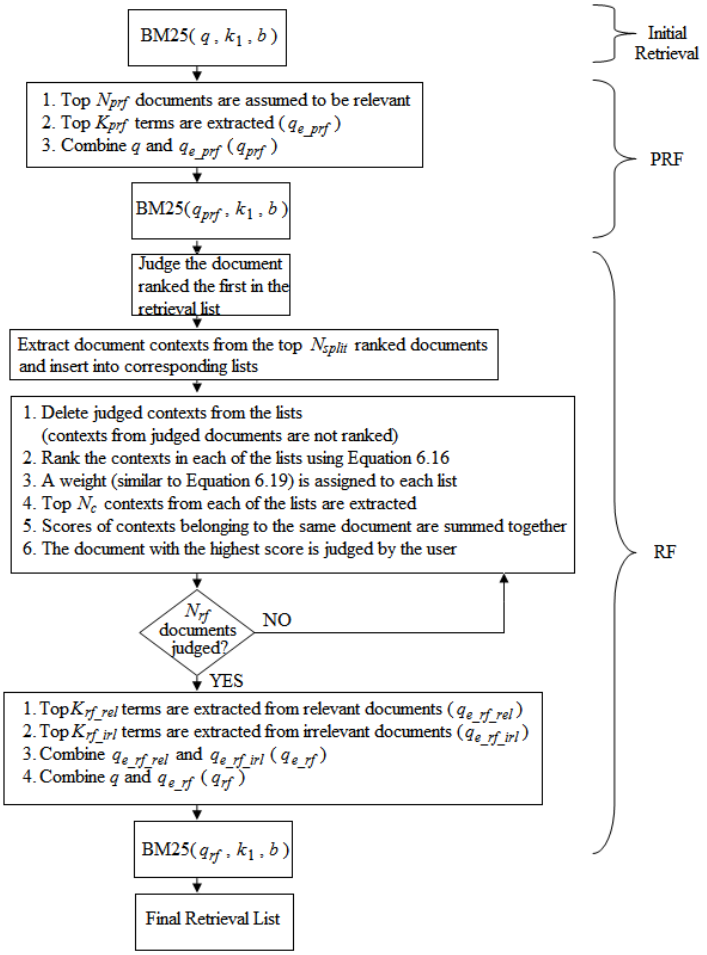

Figure 6.9: The flow of the split-list algorithm in our experiments.

6.5 Experiments

In this section we report the experimental results of using various active feedback algorithms described in previous sections. We use the TREC-2005 data collection for parameters calibration and we perform tests on TREC-6, -7, -8 and -2005 data collections. N_{rf} is set to 20 for all the active feedback algorithms. That is 20 documents are judged by the user. The relevance judgements are given by the TREC judgement files for the corresponding collections. We report the precision at 20 documents (P@20) and the MAP measures. Randomization test is used for testing statistical significance between the MAP measures for different algorithms.

Unlike Chapter 5 (Probabilistic Document-Context Based Retrieval Model) which residual collection is used for evaluation, we do not use residual collection in this chapter because different active feedback algorithms have a different set of judged documents. Therefore the residual collections are different for different algorithms which make comparisons difficult. For example, if a query has a small number of relevant documents in a collection and an active feedback algorithm successfully chooses most of the relevant documents for the user to judge, as a result the number of residual relevant documents for the query is very limited and thus it is more difficult for the query to perform better. Instead of residual collection, we use the rank freezing technique (see Section 2.4.1 (Evaluation in Relevance Feedback)) which the ranks of the judged documents in the final retrieval list are assigned according to the order of judging the documents. For example, the first judged document is assigned the rank number 1 in the final retrieval list. An active feedback algorithm finding more relevant documents for the user to judge will have a higher P@20.

Table 6.1 shows the values of the parameters used in the various algorithms. The parameters are calibrated on TREC-2005 using grid search and the same parameter values are used in all other collections. Since all collections use the same parameter values, we are not picking the best results for each

of the collections. Therefore, no sensitivity studies are performed. In practice, the parameter values can be found using cross-fold validation. We optimize the P@20 for the PRF in order to have more relevant documents for the subsequent feedback experiments. The results of the initial retrieval (BM25) and PRF in TREC-2005 are shown in Table 6.2. From the results, the performance of BM25 can be greatly improved using the PRF.

Table 6.1: Parameter values in our experiments

Algorithm	Parameters	Values
BM25	k_1, b	1.4, 0.5
PRF	N_{prf}, K_{prf}, α_{prf}	40, 40, 0.1
Standard RF	N_{rf}, K_{rf_rel}, K_{rf_irl}, α_{rf}, β_{rf}	20, 100, 40, 0.2, 0.8
Gapped-Top-N_{rf}	G	1
N_{rf}-Cluster-Centroid	N_{cc}, μ	30, 2500
MMR	λ_{MMR}	0.9
MMR-Rerank	N_{mmr}, K_{mmr_rel}, K_{mmr_irl}, α_{mmr}, β_{mmr}, λ_{MMR}	70, 30, 10, 0.2, 0.8, 0.2
Split-List	N_{split}, N_c, n, w, α_{s_rel}, α_{s_irl}, α_l, δ_l	100, 50, 25, 8, 0.9, 0.9, 0.1, 0.1

Table 6.2: Results of Initial Retrieval and PRF in TREC2005

	P@20	MAP
Initial Retrieval	.3890	.2050
PRF	.4640	.2762

Table 6.3 shows the results of different algorithms using rank freezing in the four tested TREC collections. For completeness and reference, the results without rank freezing are also shown in Table 6.4. Note that we do not perform direct comparisons on Table 6.4 because the results do not reflect the real utility perceived by the user in the relevance feedback process (i.e., the ranking of the judged documents may not be the same as the order when they are judged). In Tables 6.3 and 6.4, RF is the standard relevance feedback algorithm (Figure 6.2), GAPPED is the Gapped-Top-N_{rf} algorithm

(Figure 6.3), CLUSTER is the N_{rf}-Cluster-Centroid algorithm (Figure 6.5), MMR is the maximal marginal relevance algorithm (Figure 6.6), MMR-Rerank is the maximal marginal relevance algorithm with re-ranking of documents (Figure 6.7) and SPLIT-LIST is our proposed split-list approach (Figure 6.9) for relevance feedback. Since rank freezing is used in Table 6.3, the P@20 of RF is the same as the P@20 of PRF in Table 6.2 for TREC-2005. The best MAP values obtained for each of the TREC collections are bolded. From the results in Table 6.3, SPLIT-LIST obtained the best MAP in all the tested collections. MMR-Rerank and SPLIT-LIST are both having a re-rank step in the relevance feedback process. We can see that the results of the two "with re-rank" algorithms are better than those without re-ranking. Particularly, in TREC-2005, MMR-Rerank on average chooses 7% more relevant documents than RF and SPLIT-LIST on average chooses 13% more relevant documents than RF. Generally the P@20 of MMR-Rerank and SPLIT-LIST is also higher than that of RF on other collections. Therefore, if we want the user to have more relevant documents for judging during relevance feedback, an algorithm with a re-ranking step should be used. The difference in P@20 between SPLIT-LIST and RF shows that SPLIT-LIST can find documents that are different from the top N_{rf} ranked ones, hence increasing the diversity of the judged documents.

Note that the lower P@20 of GAPPED and CLUSTER in Table 6.3 is due to the smaller number of relevant documents chosen by the algorithms during relevance feedback. When looking at the results without rank freezing (Table 6.4), GAPPED and CLUSTER actually perform better than RF on different collections. This shows that GAPPED and CLUSTER can find some relevant documents with high diversity although the number of relevant documents is smaller.

We mainly compare our results (SPLIT-LIST) with those using MMR-Rerank since both of them have a re-rank step. MMR-Rerank performs significantly better than RF in TREC-6 and TREC-7 while SPLIT-LIST performs significantly better than RF in all the four tested collections. This shows that SPLIT-LIST is more reliable than MMR-Rerank on different sets

of queries and collections. SPLIT-LIST performs significantly better than MMR-Rerank in TREC-2005 with 90% C.I. using randomization test. Since we are using TREC-2005 for calibration of parameters, the results show that SPLIT-LIST can perform better than MMR-Rerank when the values of parameters are set properly.

Table 6.3: Results of various algorithms with rank freezing

Algorithm	TREC-6		TREC-7		TREC-8		TREC-2005	
	P@20	MAP	P@20	MAP	P@20	MAP	P@20	MAP
RF	.3370	.2580	.3660	.2606	.4240	.3032	.4640	.3257
GAPPED	.3200	.2508	.3030	.2375	.3660	.2581	.4230	.3226
CLUSTER	.3270	.2613	.3460	.2567	.3680	.2850	.4190	.3269
MMR	.3320	.2591	.3730	.2617	.4230	.2968	.4620	.3255
MMR-Rerank	.3870	.2976$^{\phi}$	.4290	.2796$^{\phi}$	.4530	.3162	.5410	.3410
SPLIT-LIST	.4310	**.2996**$^{\phi}$	.4610	**.2848**$^{\phi}$	.4880	**.3209**$^{\phi}$	.6010	**.3574**$^{\phi\gamma}$

ϕ means the MAP is statistically significantly different with the MAP in RF with 90% C.I. using randomization test..

γ means the MAP is statistically significantly different with the MAP in MMR-Rerank with 90% C.I. using randomization test.

Table 6.4: Results of various algorithms without rank freezing

Algorithm	TREC-6		TREC-7		TREC-8		TREC-2005	
	P@20	MAP	P@20	MAP	P@20	MAP	P@20	MAP
RF	.4870	.3474	.5340	.3230	.5630	.3658	.6560	.3847
GAPPED	.4840	.3751	.5420	.3307	.5880	.3728	.6790	.4026
CLUSTER	.5070	.3658	.5360	.3323	.5700	.3693	.7090	.4193
MMR	.4830	.3482	.5340	.3280	.5600	.3684	.6510	.3827
MMR-Rerank	.5040	.3836	.5490	.3381	.5580	.3767	.6980	.3894
SPLIT-LIST	.5280	.4099	.5710	.3506	.5890	.3853	.7200	.4055

Depending on the specific tasks, different active feedback algorithms may be used to increase user's satisfaction during relevance feedback. If the user has little concern of judging less relevant documents (e.g., in the case of paid judges), GAPPED or CLUSTER can be used to choose documents with a high diversity. On the other hand, if the user is more willing to judge relevant documents than non-relevant ones and at the same time does not

want to sacrifice the diversity of documents, SPLIT-LIST can be used to provide more relevant documents.

6.6 Chapter Summary

To conclude, we have applied the notion of document-context using an iterative process in relevance feedback. We split the retrieval list into sub-lists of document-contexts for different query term patterns including single query term occurrence, a pair of query terms occur in a phrase and a pair of query terms occur in proximity. Our objectives are (a) finding more relevant feedback documents and (b) increasing the diversity of the feedback documents in order to enhance user's satisfaction during relevance feedback. We also implemented different active feedback algorithms including the Gapped-Top-N_{rf}, N_{rf}-Cluster-Centroid and two versions of Maximal Marginal Relevance (MMR). One is having a re-ranking step (MMR-Rerank) and the other does not. From the experimental results, algorithms with a re-ranking step (MMR-Rerank and SPLIT-LIST) can improve performance by finding more relevant documents for the user to judge. We also show that some active feedback algorithms (Gapped-Top-N_{rf} and N_{rf}-Cluster-Centroid) can find documents with high diversity such that they can perform better than standard relevance feedback even with less relevant documents. The results also show that our proposed algorithm (SPLIT-LIST) can perform better than standard relevance feedback and more reliable than MMR-Rerank on different TREC collections.

For future studies, we can use other retrieval models such as the language modelling approach to information retrieval [Ponte and Croft, 1998] instead of using the BM25 model as the baseline model throughout the relevance feedback process. Also, instead of only using query term patterns, expansion term patterns can also be used since the contexts of expansion terms can also be relevant to the user's information need.

Chapter 7

Conclusion and Future Work

This section concludes the thesis and proposes some possible items for future studies.

7.1 Conclusion

A hybrid document-context based retrieval model has been investigated and extensively tested in retrospective experiments. We have tested the two assumptions which are Document-Training assumption and Context-Training assumption, and find that context-training performs better document-training. Different smoothing methods are also experimented and results show that different smoothing methods perform similarly when the parameters are set properly. For combining the context scores, we have used different operators including the extended Boolean operators, Dombi operators and the ordered weighted average (OWA) operators. Results show that operators following the Disjunctive Relevance Decision (DRD) principle and Aggregation Relevance Decision (ARD) principle generally performs better and operators following the Conjunctive Relevance Decision (CRD) principle. The results are consistent with the TREC ad hoc retrieval evaluation policy. We have also shown that the proposed model obey the Probability Ranking Principle (PRP).

We also have shown that TF-IDF term weights can be interpreted as making relevance decisions. Form the relevance decision-making perspective, TF-IDF term weights are the result of simplifying out probabilistic non-relevance decision model, when assuming the minimal context assumption. We have shown that the quantity $-\log P_{\hat{\partial},n}(R=0 \mid t \in d)$ to be IDF by assuming that (a) a new usage of a term arrives at a constant rate following a Poisson distribution and (b) the probability of non-relevance of term *t* is specified by our random match model of term usage. We have also proposed

a modified minimum spanning tree clustering algorithm to find the number of clusters of a term as the number of usages of the term.

By no longer making the minimal context assumption, we have developed a probabilistic document-context based model which is called the binary independence language model (BILM). We have experimented the model in relevance feedback and retrospective experiments and the results show that the proposed model is effective across different TREC collections.

Lastly, we have applied the notion of document-contexts to a split-list approach for relevance feedback. The algorithm aims to (a) find more relevance documents, and (b) increase the diversity of the documents in the relevance feedback process. Thereby enhances the user's satisfaction during relevance feedback. The results show that the algorithm is promising when compared with other similar relevance feedback algorithms.

7.2 Future Work

Context definition

In this thesis, a context is defined as the set of terms surrounding a query term within a given distance *n*. That is, a context only consists of terms which having distance from query terms less than *n*. For terms having distance from query terms greater than *n*, they are not considered to be part of the context. This can be thought of having a sharp boundary in which terms outside the boundary are not considered. On the other hand, we can have a soft boundary by introducing a weight depending on the distance from the query terms. The weight decreases as the distance from query terms increases. This is used in the positional language models approach for information retrieval [Lv and Zhi, 2009] which defines a language model for each position of a document, and score a document based on the scores of the individual positional language models. The positional language model is estimated based on propagated counts of words within a document through a

proximity-based density function. Experiment results show that the positional language model outperforms other proximity-based retrieval models.

N-gram models

Following the language modeling approach to information retrieval [Ponte and Croft, 1998], a general language model [Song and Croft, 1999] was developed based on a range of smoothing techniques and it can be extended to incorporate probabilities of phrases such as term pairs and term triples. The results in [Song and Croft, 1999] have shown that term pairs are useful in improving retrieval performance. The unigram model makes a strong assumption that each term occurs independently, while the bigram and trigram models take into account the local context. For a bigram, the probability of seeing a term depends on the probability of seeing the previous term. For a trigram, the probability of seeing a term depends on the probability of seeing the previous two terms. In this thesis, the unigram model is used for calculating the probabilities of the terms inside each of the document-contexts. As a result, the context score is the product of the probabilities of individual terms. Similar to the general language model [Song and Croft, 1999], n-gram models instead of the unigram model can be used in the proposed document-context based models. When considering the document context $c(d, k)$ at the k-th location in document d, instead of just using the $2n+1$ single terms for making local relevance decision (i.e., $\partial_{d,k}(c(d, k), q)$), n-gram models can be considered. For example, the probability of seeing the p-th term in the context $c(d, k)$ where $p \in [1, 2n+1]$ is equal to the probability of seeing the p-th term given the previous $(p-1)$-th term.

Part-of-Speech (POS) tagging

Besides lexical features such as single query terms, a pair of query terms occurred in a phrase and proximity, semantic features such as part-of-speech

tags can also be used to identify better patterns in the document-contexts. Accurate part-of-speech tagging of natural language data can improve the effectiveness of information retrieval models. Recently, Lioma and Blanco [2009] introduced a new type of term weight that is computed from part-of-speech (POS) n-gram statistics. The POS-based term weight represents how informative a term is in general, based on the 'POS contexts' in which it generally occurs in language. Five different computations of the POS-based term weights were proposed and experimental results shown that when conventional retrieval models (e.g., BM25 model) is integrated with the POS-based term weights, the effectiveness of the retrieval increases. Therefore, besides only using terms, we can also use part-of-speech tags for calculating the context scores.

Language modeling approach

The document-context based models in this thesis are mainly developed using the log-odds similar to the binary independence retrieval (BIR) model [Robertson and Sparck Jones, 1976]. Different document-context based models can also be developed by considering the probability of generating the query q by the document-context $c(d, k)$ which is similar to the language modeling approach to information retrieval [Ponte and Croft, 1998].

References

AIZAWA, A. 2003. An information-theoretic perspective of tf-idf measures. *Information Processing & Management*, 39, 1, 45-65.

AL-MASKARI, A., SANDERSON, M., AND CLOUGH, P. 2008. Relevance judgments between TREC and Non-TREC assessors. In *Proceedings of ACM SIGIR 08*, pp. 683-684.

AMATI, G. AND VAN RIJSBERGEN, C.J. 1998. Semantic information retrieval. In CRESTANI, F., LALMAS, M. AND VAN RIJSBEREGEN, C.J. (EDS). *Information Retrieval: Uncertainty and Logic*. Kluwer Academic Publisher, 357-389.

AMATI, G. AND VAN RIJSBERGEN, C.J. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20, 4, 357-389.

BAEZA-YATES, R. AND RIBEIRO-NETO, B. 1999. *Modern Information Retrieval*. Addison-Wesley: New York.

BAI, J., SONG, D., BRUZA, P., NEI, J.Y., AND CAO, G. 2005. Query expansion using term relationships in language models for information retrieval. In *Proceedings of ACM CIKM 05*, pp. 688-695.

BARON, J.R., LEWIS, D.D., AND OARD, D.W. 2006. *TREC 2006 legal track*. http://trec-legal.umiacs.umd.edu/.

BELKIN, J.N. 1980. Anomalous state of knowledge as a basis for information retrieval. *Canadian Journal of Information Science*, 5, 133-143.

BODOFF, D. AND ROBERTSON, S.E. 2004. A new unified probabilistic model. *Journal of the American Society for Information Science and Technology*, 55, 6, 471-487.

BOLLMANN, P. AND WONG, S.K.M. 1987. Adaptive linear information retrieval models. In *Proceedings of ACM SIGIR 87*, pp. 157-163.

BOOKSTEIN, A. AND SWANSON, D. 1974. Probabilistic models for automatic indexing. *Journal of the American Society for Information Science*, 25, 312-318.

BRUZA, P. AND SONG, D. 2003. A comparison of various approaches for using probabilistic dependencies in language modelling. In *Proceedings of ACM SIGIR 03*, pp. 219-420.

BRUZA, P.D. AND HUIBERS, T.W.C. 1994. Investigating aboutness axioms using information fields. In *Proceedings of ACM SIGIR 94*, pp. 112-121.

BRUZA, P.D. AND HUIBERS, T.W.C. 1996. A study of aboutness in information retrieval. *Artificial Intelligence Review*, 10, 5-6, 381-407.

BUCKLEY, C. AND HARMAN, D. 2003. Reliable information access final workshop report. In the *Workshop for Reliable Information Access 2003* (http://nrrc.mitre.org/NRRC/Docs_Data/RIA_2003/ria_final.pdf).

BURGESS, C., LIVESAY, K., AND LUND, K. 1998. *Explorations in context space: words, sentences, discourse. Discourse Processes*, 25, 2&3, 211-257.

BURGESS, C. AND LUND, K. 1997. Modelling parsing constraints with high-dimensional semantic space. *Language and Cognitive Processes*, 12, 2, 177-210.

CALADO, P., RIBEIRO-NETO, B., ZIVIANI, N., MOURA, E., AND SILVA, I. 2003. Local versus global link information in the web. *ACM Transactions on Information Systems (TOIS)*, 21, 1, 42-63.

CALLAN, J.P. 1994. Passage-level evidence in document retrieval. In *Proceedings of ACM SIGIR 94*, pp. 302-310.

CARBONELL, J. AND GOLDSTEIN, J. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of ACM SIGIR 98*, pp. 335-336.

CHANG, Y.K., CIRILLO, C., AND RAZON, J. 1971. Evaluation of feedback retrieval using modified freezing, residual collection & test and control groups. *The SMART retrieval system*, G. Salton (ed), Englewood Cliffs, N.J.: Prentice-Hall, Inc., pp. 355-370.

CHEN, S.F. AND GOODMAN, J. 1996. An Empirical Study of Smoothing Techniques for Language Modelling. In *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pp. 310-318.

COOK, K.H. 1975. A threshold model of relevance decisions. *Information Processing & Management*, 11, 124-135.

COOPER, W.S. 1971. A definition of relevance for information retrieval. *Information Storage & Retrieval*, 7, 19-37.

COOPER, W.S. 1981. Gedanken experimentation: an alternative to traditional system testing? In K. Sparck Jones (ed.), *Information Retrieval Experiment*, Butterworths.

COOPER, W.S. 1995. Some inconsistencies and misidentified modeling assumptions in probabilistic information retrieval. *ACM Transactions on Information Systems (TOIS)*, 13, 1, 100-111.

COOPER, W.S., CHEN, A., AND GEY, F.C. 1993. Full text retrieval based on probabilistic equations with coefficients fitted by logistic regression. In *Proceedings of TREC-2*, pp. 57-66.

COOPER, W.S., GEY, F.C., AND DABNEY, D.P. 1992. Probabilistic retrieval model based on staged logistic regression. In *Proceedings of ACM SIGIR 92*, pp. 198-210.

COOPER, W.S. AND MARON, M. 1978. Foundations of probabilistic and utility-theoretic indexing. *Journal of the ACM*, 25, 1, 67-80.

CRESTANI, F., LALMAS, M., VAN RIJSBERGEN, C.J., AND CAMPBELL, I. 1998. "Is this document relevant? …probably": a survey of probabilistic models in information retrieval. *ACM Computing Surveys*, 30, 4, 528-552.

CROFT, W. AND HARPER, D. 1979. Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35, 285-295.

CUADRA, C.A. AND KATTER, R.V. 1967. Experimental studies of relevance judgments. *Final Report, Project Summary, Vol. 1, System Development Corporation Reports* TM-3520, 001-002.

DAMERAU, F. 1965. An experiment in automatic indexing. *American Documentation*, 16, 283-289.

DANG, E.K.F., LUK, R.W.P., LEE, D.L., HO, K.S., AND CHAN, C.S. 2006. Query-specific clustering using document-context based similarity score. In *Proceedings of ACM CIKM 06*, pp. 886-887.

DANG, E.K.F., WU, H.C., LUK, R.W.P., AND WONG, K.F. 2009. Building a framework for the probability ranking principle by a family of expected weighted rank. *ACM Transactions on Information Systems (TOIS)*, 27, 4.

DE KRETSER, O. AND MOFFAT, A. 1999. Effective document presentation with a locality-based similarity heuristic. In *Proceedings of ACM SIGIR 99*, pp. 113-120.

DE MOURA, E.S., NAVARRO, G., ZIVIANI, N., AND BAEZA-YATES, R. 2000. Fast and flexible word searching on compressed text. *ACM Transactions on Information Systems (TOIS)*, 18, 2, 113-139.

DE VRIES, A.P. AND ROELLEKE, T. 2005. Relevance information: a loss of entropy but a gain for IDF. In *Proceedings of ACM SIGIR 05*, pp. 282-289.

DOMINICH, S. 2000. A unified mathematical definition of classical information retrieval. *Journal of the American Society for Information Science*, 51, 7, 614-625.

DOMBI, J. 1982. A general class of fuzzy operators, the DeMorgan class of fuzzy operators and fuzziness measures induced by fuzzy operators. *Fuzzy Sets and Systems*, 8, 149-163.

DOMINICH, S. 2000. A unified mathematical definition of classical information retrieval. *Journal of the American Society for Information Science*, 51, 7, 614-625.

DYCKHOFF, H. AND PEDRYCZ, W. 1984. Generalized means as Model of Compensative Connectives. *Fuzzy Sets and Systems*, 14, 143-154.

FALOUTSOS, C. AND CHRISTODOULAKIS, S. 1987. Description and performance analysis of signature file methods for office filing. *ACM Transactions on Information Systems (TOIS)*, 5, 3, 237-257.

FELLER, W. 1968. *An Introduction to Probability Theory and Its Applications*, Vol. I, third ed. Wiley, New York.

FOX, E., BETRABET, S., KOUSHIK, M., AND LEE, W. 1992. Extended Boolean Models Information Retrieval: *Data Structures & Algorithms*, pp. 393-418.

FRAKES, W.B. AND BAEZA-YATES, R. 1992. *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall.

FRENCH, S. 1986. Decision Theory: *An Introduction to the Mathematics of Rationality*, Ellis Horwood: Chichester.

FUHR, N. 1989. Models for retrieval with probabilistic indexing. *Information Processing & Management*, 25, 1, 55–72.

FUHR, N. 1992. Probabilistic models in information retrieval. *The Computer Journal*, 35, 3, 243-255.

FUHR, N. 2008. A probability ranking principle for interactive information retrieval. *Information Retrieval*, 11, 3, 251-265.

FUJII, A., IWAYAMA, M., AND Kando, N. 2005. Overview of patent retrieval task as NTCIR-5. In *Proceedings of the NTCIR-5 Workshop*.

GAO, J., NIE, J.Y., WU, G., AND CAO, G. 2004. Dependence language model for information retrieval. In *Proceedings of ACM SIGIR 04*, pp. 170-177.

GALE, W.A., CHURCH, K.W., AND YAROWSKY, D. 1992. Work on Statistical Methods for Word Sense Disambiguation. In *Working Notes,*

AAAI Fall Symposium Series: Probabilistic Approaches to Natural Language, pp. 54-60.

GALE, W.A., CHURCH, K.W., AND YAROWSKY, D. 1993. A method for disambiguating word senses in a large corpus, *Computers and Humanities*, 26, 415-439.

GEBHARDT, F. 1975. A simple probabilistic model for the relevance assessment of documents. *Information Processing & Management*, 11, 59.

GORDON, M. AND LENK, P. 1991. A Utility Theoretic Examination of the Probability Ranking Principle in Information Retrieval, *Journal of the American Society for Information Science*, 42, 10, 703-714.

HARMAN, D. 1992. Relevance feedback revisited. In *Proceedings of ACM SIGIR 98*, pp. 1-10.

HARMAN, D. 2004. Personal communication at NTCIR-4.

HARTER, S.P. 1974. A probabilistic approach to automatic keyword indexing. PhD Thesis, Graduate Library, The University of Chicago, Thesis No. T25146.

HARTER, S.P. 1975a. A probabilistic approach to automatic keyword indexing. Part I: On the distribution of specialty words in a technical literature. *Journal of the American Society for Information Science*, 26, 4, 197–206.

HARTER, S.P. 1975b. A probabilistic approach to automatic keyword indexing. Part II: An algorithm for probabilistic indexing. *Journal of the American Society for Information Science*, 26, 4, 280–289.

HIEMSTRA, D. 1998. A linguistically motivated probabilistic model of information retrieval. In *Proceedings of the European Conference on Digital Library*, 569-584.

HIEMSTRA, D. AND ROBERTSON, S. E. 2001. Relevance Feedback for Best Match Term Weighting Algorithms in Information Retrieval. In *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*.

HUANG, X. PENG, F., SHUURMANS, D., CERCONE, N., AND ROBERTSON, S.E. 2003. Applying machine learning to text segmentation for information retrieval. *Information Retrieval*, 6, 4, 333-362.

HUNG, K.Y., LUK, R.W.P., YUENG, D.S., CHUNG, K.F.L., AND SHU, W.H. 2001. Determination of context window size. *International Journal of Computer Processing of Oriental Languages*, 14, 1, 71-80.

JEFFREYS, H. 1948. *Theory of Probability*. Clarendon Press, Oxford, second edition.

JELINEK, F. AND MERCER, R. 1980. Interpolated estimation of markov source parameters from sparse data. In *Proceedings of the Workshop on Pattern Recognition in Practice*, pp. 381-402.

JOACHIMS, T. 1997. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In *Proceedings of ICML-97*, pp. 143-151.

JOACHIMS, T. 1999. Making large-Scale SVM Learning Practical. *Advances in Kernel Methods, Support Vector Learning*, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press.

JOHNSON, W.E. 1932. Probability: deductive and inductive problems. *Mind*, 41, 421-423.

KASZKIEL, M. AND ZOBEL, J. 1997. Passage retrieval revisited. In *Proceedings of ACM SIGIR 97*, pp. 178-185.

KASZKIEL, M., ZOBEL, J., AND SACKS-DAVIS, R. 1999. Efficient passage ranking for document databases. *ACM Transactions on Information Systems (TOIS)*, 17, 4, 406-439.

KAUFMAN, L. AND ROUSSEEUW, P.J. 1990. *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley & Sons.

KISHIDA, K., CHEN, K-H., LEE, S., KURIYAMA, K., KANDO, N., CHEN, H-H., AND MYAENG, S.H. 2005. Overview of CLIR task at the fifth NTCIR workshop. In *Proceedings of the NTCIR-5 Workshop*, NII, Japan.

KLIR, G.J. 1992. *Fuzzy Sets, Uncertainty, and Information*. Prentice Hall, Inc.: New Jersey.

KOLMOGOROV, A.N. 1950. *Foundations of the Theory of Probability*. New York: Chelsea.

KONG, Y.K., LUK, R.W.P., LAM, W., HO, K.S., AND CHUNG, F.L. 2004. Passage-based retrieval based on parameterized fuzzy operators. In *ACM SIGIR Workshop on Mathematical/Formal Methods for Information Retrieval*.

KUPIEC, J., PEDERSEN, J., AND CHEN, F. 1995. A trainable document summarizer. In *Proceedings of ACM SIGIR 95*, pp. 68-73.

KULLBACK, S. 1968. *Information Theory and Stutistics*. New York: Dover Publications.

KUPIEC, J., PEDERSEN, J., AND CHEN, F. 1995. A trainable document summarizer. In *Proceedings of ACM SIGIR 95*, pp. 68-73.

KWOK, K.L. 1995. A network approach to probabilistic information retrieval. *ACM Transactions on Information Systems (TOIS)*, 13, 3, 324-353.

LAFFERTY, J. AND ZHAI, C. 2001. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of ACM SIGIR 01*, pp. 111-119.

LAU, K.Y.K. AND LUK, R.W.P. 1999. Word-sense classification by hierarchical clustering. *Communications of COLIPS*, 9, 1, 101-121.

LAVRENKO, V. AND CROFT, W.B. 2001. Relevance-based language model. In *Proceedings of ACM SIGIR 01*, pp. 120-127.

LAVRENKO, V. AND CROFT, W.B. 2003. Relevance models in information retrieval. W.B. Croft (ed.) *Language Modeling for Information Retrieval*, Academic Publishers.

LALMAS, M. 1997. Dempster-Shafer's Theory of Evidence applied to Structured Documents: modelling Uncertainty. In *Proceedings of ACM SIGIR 97*, pp. 110-118.

LEASE, M. 2008. Incorporating Relevance and Pseudo-relevance Feedback in the Markov Random Field Model. *TREC-2008*.

LEE, J.H. 1997. Analyses of multiple evidence combination. In *Proceedings of ACM SIGIR 97*, pp. 267-276.

LEE, K.S., CROFT, W.B., AND ALLAN, J. 2008. A cluster-based resampling method for pseudo-relevance feedback. In *Proceedings of ACM SIGIR 08*, pp. 235-242.

LIDSTONE, G.J. 1920. Note on the general case of the Bayes-Laplace formula for inductive or a posteriori probabilities. *Transactions of the Faculty of Actuaries*, 8, 182-192.

LIN, J. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37, 1, 145-151.

LIOMA, C. AND BLANCO, R. 2009. Part of Speech Based Term Weighting for Information Retrieval. In *Proceedings of ECIR 09*, pp. 412-423.

LIU, X. AND CROFT, W.B. 2002. Language models for information retrieval: passage retrieval based on language models. In *Proceedings of ACM CIKM 02*, pp. 375-382.

LOSEE, R.M. 1996. Evaluating Retrieval Performance Given Database and Query Characteristics: Analytic Determination of Performance Surfaces. *Journal of the American Society for Information Science,* 47, 1, 95-105.

LUCASSEN, J.M. AND MERCER, R.L. 1984. An information theoretic approach to the automatic determination of phonemic baseforms. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 84*, pp. 42.5.1-42.5.4.

LUHN, H. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development,* 2, 2, 159-165.

LUK, R.W.P. AND KWOK, K.L. 2002. A comparison of different Chinese document indexing strategies and retrieval models. *ACM Trans. on Asian Language Information Processing*, 1, 3, 207-224.

LUK, R.W.P., LEONG, H.V., DILLON, T.S., CHAN, A.T.S., CROFT W.B., AND ALLAN, J. 2002. A survey in indexing and searching XML documents. *Journal of the American Society for Information Science and Technology*, 53, 6, 415-437.

LUND, K. AND BURGESS, C. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. In *Behavior Research Methods, Instrumentation, and Computers*, pp. 203-208.

LV, Y. AND ZHAI, C.X. 2009. Positional language models for information retrieval. In *Proceedings of ACM SIGIR 09*, pp. 299-306.

MACQUEEN, J.B. 1967. Some Methods for classification and Analysis of Multivariate Observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281-297.

MALIK, S., KAZAI, G., LALMAS, M., AND FUHR, N. 2005. Overview of INEX 2005, Advances in XML Information Retrieval and Evaluation. In *Proceedingsof the Fourth Workshop of the INitiative for the Evaluation of XML Retrieval (INEX 2005).*

MARGULIS, E. 1992. N-Poisson document modelling. In *Proceedings of ACM SIGIR 92,* pp. 177-189.

MARON, M.E. AND KUHNS, J.L. 1960. On relevance, probabilistic indexing and information retrieval. *Journal of the ACM*, 25, 3, 216-244.

METZLER, D. AND CROFT, W.B. 2005. A Markov random field model for term dependencies. In *Proceedings of ACM SIGIR 05*, pp. 472-479.

MIYAMOTO, S. 1990. *Fuzzy Sets in Information Retrieval and Cluster Analysis*. Kluwer Academic Publishers, Dordrecht.

MOFFAT, A. AND ZOBEL, J. 1996. Self-indexing inverted files for fast text retrieval. *ACM Transactions on Information Systems (TOIS)*, 14, 4, 349-379.

NAVARRO, G. AND BAEZA-YATES, R. 1997. Proximal nodes: A model to query document databases by content and structure. *ACM Transactions on Information Systems (TOIS)*, 15, 4, 400–435.

NELSON, A.L. AND STENTON, S.P. 1992. Dialogue modeling for information access. In *Proceedings of ASLIB*, 44:7-87-8, 275-281.

NEY, H., ESSEN, U., AND KENSER, R. 1994. On structuring probabilistic dependencies in stochastic language modeling. *Computer Speech and Language*, 8, 1–38.

PAICE, C.P. 1984. Soft evaluation of boolean search queries in information retrieval systems. *Information Technology: Research and Development*, 3, 1, 33–42.

PAPINENI, K. 2001. Why inverse document frequency? In *Proceedings of the North American Association for Computational Linguistics*, pp. 25-32.

PICKENS, J. AND MACFARLANE, A. 2006. Term context models for information retrieval. In *Proceedings of ACM CIKM 06*, pp. 559-566.

PONTE, J. AND CROFT, W.B. 1998. A language modeling approach in information retrieval. In *Proceedings of ACM SIGIR 98*, pp. 275-281.

PORTER, M. 1980. An algorithm for suffix stripping. *Program*, 14, 3, 130-137.

REES, A.M., SCHULTZ, D.G., BAUMANIS, G., MARCUS, S., ROTHENBERG, L., SARACEVIC, T., STERN, M., AND ZULL, C. 1967. A field experimental approach to the study of relevance assessments in relation to document searching. Final Report, Case Western Reserve University, Cleveland.

ROBERTS, F.S. 1979. *Measurement Theory*, Addison-Wesley: Reading Massachusetts.

ROBERTSON, S.E. 1976. The probabilistic character of relevance. *Information Processing & Management*, 13, 247-251.

ROBERTSON, S.E. 1977. The probability ranking principle in IR. *Journal of Documentation*, 33, 106-119.

ROBERTSON, S.E. 1997. Overview of the Okapi projects. *Journal of Documentation*, 53, 1, 3-7.

ROBERTSON, S.E. 2004. Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation*, 60, 503-520.

ROBERTSON, S.E. 2005. On event spaces and probabilistic models in information retrieval. *Information Retrieval*, 8, 319-329.

ROBERTSON, S.E. AND SPARCK JONES, K. 1976. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27, 3, 129-146.

ROBERTSON, S.E., VAN RIJSBERGEN, C.J., AND PORTER, M.F. 1980. Probabilistic models of indexing and searching. In *Proceedings of ACM SIGIR 80*, pp. 35-56.

ROBERTSON, S.E. AND WALKER, S. 1994. Some simple approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of ACM SIGIR 94*, pp. 232-241.

ROBERTSON, S.E. AND WALKER, S. 1997. On relevance weights with little relevance information. In *Proceedings of ACM SIGIR 97*, pp 16-24.

ROBERTSON, S.E. AND WALKER, S. 1999. Okapi/Keenbow at TREC-8, automatic ad hoc, filtering, VLC and interactive. In *Proceedings of TREC-8 Conference*, pp. 151-162.

ROBERTSON, S.E., WALKER, S., AND HANCOCK-BEAULIEU, M.M. 1995. Large test collection experiments on an operational, interactive system: Okapi at TREC. *Information Processing & Management*, 31, 345-360.

ROCCHIO, J.J. 1971. Relevance feedback in information retrieval. *In The SMART Retrieval System*, G. Salton, Ed. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1971, pp. 313-323.

ROELLEKE, T. 2003. A frequency-based and a poisson-based definition of probability of being informative. In *Proceedings of ACM SIGIR 03*, pp. 227-234.

ROELLEKE, T. AND WANG, J. 2006. A parallel derivation of probabilistic information retrieval models. In *Proceedings of ACM SIGIR 06*, pp. 107-114.

SALTON, G., ALLAN, J., AND BUCKLEY, C. 1993. Approaches to passage retrieval in full text information systems. In *Proceedings of ACM SIGIR 93*, pp. 49-56.

SALTON, G., FOX, E.A., AND WU, H. 1983. Extended Boolean information retrieval. *Communications of the ACM*, 26, 11, 1022-1036.

SALTON, G. AND BUCKLEY, C. 1988. Term weighting approaches in automatic text retrieval. *Information Processing & Management*, 24, 5, 513-523.

SALTON, G. AND BUCKLEY, C. 1990. Improving retrieval performance by retrieval feedback. *Journal of the American Society for Information Science*, 41, 4, 288-297.

SALTON, G., WONG, A., AND YANG, C.S. 1975. A vector space model for information retrieval. *Journal of the American Society for Information Science*, 18, 11, 613-620.

SARACEVIC, T. 1975. Relevance: A Review of and a framework for the thinking on the notion in information science. *Journal of the American Society for Information Science*, 26, 6, 321-343.

SHAFER, G. 1976. *A Mathematical Theory of Evidence*, Princeton University Press, Princeton, New Jersey.

SHEN, X. AND ZHAI, C. 2005. Active feedback in ad hoc information retrieval. In *Proceedings of ACM SIGIR 05*, pp. 59-66.

SONG, D. AND BRUZA, P.D. 2003. Towards context-sensitive information inference. *Journal of the American Society for Information Science and Technology*, 54, 4, 321-334.

SONG, F. AND CROFT, W.B. 1999. A general language model for information retrieval. In *Proceedings of ACM CIKM 1999*, pp. 316-321.

SPARCK-JONES, K. 1972. Exhaustivity and specificity. *Journal of Documentation*, 28, 11-21.

SPARCK-JONES, K. 2004. IDF term weighting and IR research lessons. *Journal of Documentation*, 60, 521-523.

SPARCK-JONES, K., WALKER, S., AND ROBERTSON, S.E. 2000. A probabilistic model of information retrieval: development and comparative experiments. *Information Processing & Management*, 36, 6, 779-840.

STARK, H. AND WOODS, J.W. 1994. *Probability, random processes, and estimation theory for engineers*. Englewood Cliffs, N.J., Prentice-Hall International, Inc.

STOCKOLOVA, N.A. 1977. Elements of a semantic theory of information retrieval: I. The concepts of relevance and information language. *Information Processing & Management*, 13, 4, 227-234.

STROHMAN, T., METZLER, D., TURTLE, H., AND CROFT, W.B. 2004. Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligence Analysis*.

TANG, R. AND SOLOMON, P. 1998. Toward an understanding of the dynamics of relevance judgment. *Information Processing & Management*, 34, 2/3, 237-256.

TURTLE, H. AND CROFT, W.B. 1991. Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems (TOIS)*, 9, 3, 187-222.

TVERSKY, A. AND KRANTZ, D.H. 1970. The dimensional representation and the metric structure of similarity data. *Journal of Mathematical Psychology*, 7, 572-596.

VAN RIJSBERGEN, C.J. 1974. Foundation of evaluation. *Journal of Documentation*, 30, 4, 365-373.

VAN RIJSBERGEN, C.J. 1979. *Information Retrieval*. Butterworths.

VAN RIJSBERGEN, C.J. 1986. A non-classical logic for information retrieval. *The Computer Journal*, 29, 6, 481-485.

VECHTOMOVA, O., KARAMUFTUOGLU, M., AND ROBERTSON, S.E. 2006. On Document Relevance and Lexical Cohesion between Query Terms. *Information Processing & Management*, 24, 5, 1230-1247.

VECHTOMOVA, O. AND ROBERTSON, S.E. 2000. Integration of collocation statistics into the probabilistic retrieval model. In *Proceedings of the 22nd British Computer Society - Information Retrieval Specialist Group Conference*, pp. 165-177.

WALKER, S., ROBERTSON, S.E., BOUGHANEM, M., JONES, G.J.F., AND SPARCK-JONES, K. 1997. Okapi at TREC-6 Automatic ad hoc, VLC, routing, filtering and QSRD. In *Proceedings of TREC-6*, pp. 125-136.

WALLER, W.G. AND KRAFT, D.H. 1979. A mathematical model of a weighted Boolean retrieval system. *Information Processing & Management*, 15, 5, 235-245.

WANG, D.Y., LUK, R.W.P., WONG, K.F., AND KWOK K.L. 2006. An information retrieval based on discourse type. In *Proceedings of NLDB, LNCS 3999*, pp. 197-202.

WANG, Y.D. AND FORGIONNE, G. 2006. A decision-theoretic approach to the evaluation of information retrieval systems. *Information Processing & Management*, 42, 4, 863-874.

WANG, Z.W., WONG, S.K.M., AND YAO, Y.Y. 1992. An analysis of vector space models based on computational geometry. In P*roceedings of ACM SIGIR 92*, pp. 152-160.

WHITE, R.Y., KULES, B., DRUCKER, S.M., AND SCHRAEFEL, M.C. 2006. Supporting exploratory search: introduction. *Communications of the ACM*, 49, 4, 36-39.

WONG, A.K.C. AND GHARAMAN, D. 1975. A statistical analysis of interdependence in character sequences. *Information Sciences*, 8, 2, 173-188.

WONG, K.F., SONG, D., BRUZA, P.D., AND CHENG, C-H. 2001. Application of aboutness to functional benchmarking in information retrieval, *ACM Transactions on Information Systems (TOIS)*, 19, 4, 337-370.

WONG, S.K.M., ZIARKO, W., RAGHAVAN, V.V., AND WONG, P.C.N. 1986. On extending the vector space model for Boolean query processing. In *Proceedings of ACM SIGIR 86*, pp. 175-185.

WONG, S.K.M., ZIARKO, W., RAGHAVAN, V.V., AND WONG, P.C.N. 1989. Extended Boolean query processing in the generalized vector space model. *Information Systems*, 14, 1, 47-63.

WONG, S.K.M., ZIARKO, W., AND WONG, P.C.N. 1985. Generalized vector space model in information retrieval. In *Proceedings of ACM SIGIR 85*, pp. 18-25.

WONG, W.S., LUK, R.W.P., LEONG, H.V., HO E.K.S., AND LEE, D.L. 2008. Re-examining effects of adding relevance information in a relevance feedback environment. *Information Processing & Management*, 44, 3, 1086-1396.

WU, H.C., LUK, R.W.P., WONG, K.F., KWOK, K.L., AND LI, W.J. 2005. A retrospective study of probabilistic context-based retrieval. In *Proceedings of ACM SIGIR 05*, pp. 663-664.

XI, W., XU-RONG, R., KHOO, C.S.G., AND LIM, E.P. 2001. Incorporating window-based passage-level evidence in document retrieval. *Journal of Information Science*, 27, 2, 73-80.

XU, J. AND CROFT, W.B. 2000. Improving the effectiveness of information retrieval using local context analysis. *ACM Transactions on Information Systems (TOIS)*, 18, 1, 79-112.

YAGER, R.R. 1988. On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man and Cybernetics*, 18, 183-190.

YAO, Y.Y. AND WONG, S.K.M. 1991. Preference structure, inference and set-oriented retrieval. In *Proceedings of ACM SIGIR 91*, pp. 211-218.

YU, C.T. AND SALTON, G. 1976. Precision weighting – an effective automatic indexing method. *Journal of the ACM*, 23, 1, 76-88.

ZADEH, L.A. 1965. Fuzzy sets. *Information and Control*, 8, 3, 338-353.

ZAHN, C.T. 1971. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computers*, 20, 1, 68-86.

ZHAI, C.X. AND LAFFERTY, J. 2004. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22, 2, 179-214.

ZHAI, C.X. AND LAFFERTY, J. 2006. A risk minimization framework for information retrieval. In *Information Processing & Management*, 42, 1, 31-55.