



FEZA BASKAYA

Simulating Search Sessions  
in  
Interactive Information Retrieval  
Evaluation



ACADEMIC DISSERTATION

To be presented, with the permission of  
the Board of the School of Information Sciences  
of the University of Tampere,  
for public discussion in the Auditorium Pinni B 3116,  
Kanslerinrinne 1, Tampere, on June 13th, 2014, at 12 o'clock.

UNIVERSITY OF TAMPERE

FEZA BASKAYA

Simulating Search Sessions  
in  
Interactive Information Retrieval  
Evaluation

*Acta Universitatis Tamperensis 1949*  
*Tampere University Press*  
*Tampere 2014*

ACADEMIC DISSERTATION  
University of Tampere  
School of Information Sciences  
Finland

The originality of this thesis has been checked using the Turnitin OriginalityCheck service in accordance with the quality management system of the University of Tampere.

Copyright ©2014 Tampere University Press and the author

Cover design by  
Mikko Reinikka

Distributor:  
kirjamyynänti@juvenes.fi  
<http://granum.uta.fi>

Acta Universitatis Tamperensis 1949  
ISBN 978-951-44-9498-7 (print)  
ISSN-L 1455-1616  
ISSN 1455-1616

Acta Electronica Universitatis Tamperensis 1434  
ISBN 978-951-44-9499-4 (pdf)  
ISSN 1456-954X  
<http://tampub.uta.fi>

Suomen Yliopistopaino Oy – Juvenes Print  
Tampere 2014



# Acknowledgments

Sitting in a time-machine called Earth, one does not even notice that time flies like an arrow. At the School of Information Sciences (SIS) which by the way means “fog” in Turkish, I started as a software developer 10 years ago. Having written Information Retrieval related software for some time I became a university student again, this time, a Ph.D. student. After trying to find the correct path in a foggy environment, I have seen the lights of a FIRE (Finnish Information Retrieval Experts), around which master chefs have gathered. Enthralled by the flickering fire, I got thirsty for more knowledge. Countless discussions around fire ushered the way to cooking this thesis. Finland’s wonderful nature, with lakes, forests, and beauties gave inspiration for a myriad of recipes. In addition, the ingredients for the recipe are collected from the endless World Wide Web. Finally, the dish is now cooked with the kind advice of the chefs around the FIRE; one last challenge to stand is opponent’s wisdom. Then, my precious Ph.D. thesis is ready to be served to the wide world. Hopefully, it instigates further recipes in this specific domain pursuant to Zeitgeist. Bon appetite!

First of all, I would like to thank my supervisors and co-authors Prof. Kalervo Järvelin and Dr. Heikki Keskustalo. It was a pleasure to work with you. Without your positive attitude, this piece of art would not have been created at all. I learned a lot during our weekly discussions. As Kal always said, I have a Bosphorus bull attitude. I would like to have simple straightforward novel and palpable methods and the best results. However, I learnt how everything is relative, how differently one could interpret the facts, among other things. Moreover, Heikki’s strictness about every detail just contributed to the perfection of the papers and this thesis. I am delighted in having both of you as my supervisors.

Further, I would like to thank my colleagues Dr. Sanna Kumpulainen, Dr. Paavo Arvola, Dr. Sami Serola, and Dr. Eija Airio for their nice discussion about moving heaven and earth as well as other Ph.D. students and FIRE group experts for supportive comments during FIRE seminars. I am grateful to Professors Jaana

Kekäläinen, Eero Sormunen, Reijo Savolainen, and Pertti Vakkari for providing knowledge and wisdom.

Furthermore, I am grateful to the pre-reviewers of my thesis, Dr. Leif Azzopardi, University of Glasgow, UK, and Assistant Professor Mark D. Smucker, University of Waterloo, Canada, for their assessments.

Finally, I want to thank all my friends and my family. I thank my parents, Sevim and Kemal, for raising us, and my brothers, Dr. Zafer Baskaya and candidate Dr. Alparslan Baskaya, for creating a competitive environment. Last, I owe my wife, Katri, a debt of gratitude for her constructive positive outlook on life, and our children, Sema Sofia and Martti Fatih for their very being and sweetness.

In case I forgot your name, I thank all of you who feel thanks are deserved.  
Love, Peace, and Happiness!

Tampere May 9<sup>th</sup> 2014

Feza Baskaya

# Abstract

Modern knowledge society would not be possible without Information Retrieval (IR), because of the ever-growing amount of information available on the Internet. Information Retrieval provides crucial ways of finding the proverbial needle in a haystack. While computing technology is nowadays ubiquitous, users interact with various computer interfaces with varying goals and time constraints in order to complete their tasks, which may be initiated by their work or leisure-related activity. Thereby, Interactive Information Retrieval (IIR), which is the subject of this thesis, constitutes an important part of task performance. Users' interaction is shaped by users' personal and search characteristics, such as query formulation strategy, strategies for scanning and assessing of the search results, as well as users' feedback behavior.

Experimental evaluation is essential to the assessment of the effectiveness of IR systems. The traditional approach to measuring the effectiveness of diverse IR systems goes back to the Cranfield tests in the 1960s. However, neither user characteristics nor time are considered in the traditional evaluation process. In the Cranfield-type tests, still popular today, users are taken into account only marginally and their interests are represented in relevance assessments, evaluation metrics and topics to some extent. However, interaction with an IR system can be dissected more precisely and users' interaction during a search session can be divided further into subtasks. This in turn affects the evaluation process of IR systems. Moreover, users' feedback during a search session, which may be of high or poor quality, can be exploited to improve the search results. This again influences the effectiveness of search systems. In the present thesis, we examine the effects of users' characteristics and the relevance feedback behavior on search effectiveness. While conducting interactive experiments with test persons is costly in terms of time and resources, our experiments are based on user behavior simulations, which can be conducted within a short time, even though a vast number of sessions representing various user characteristics are reproduced.

Our study suggests that relevance feedback can be utilized in conjunction with classification algorithms to improve search results. Further, a realistic level of fallibility in the feedback process does not deteriorate the search outcomes significantly. When time is taken into account, it plays a major role in the evaluation process. Comparing the different search environments and strategies may be considered in respect to time expended during the search session. In that case, traditional evaluation metrics may deliver misleading conclusions in experiments. Further, we examined all possible query formulation strategies. Our experiments indicate that there is no single winning strategy that performs best across all topical search tasks. Moreover, we show that conventional IR experiments are not aware of user-dependent search variables such as query formulation, result scanning and assessing behavior, which govern the subtasks of the search process and the effectiveness of IIR. Therefore, the effect of these variables should be taken into account in the IIR evaluation process.

Finally, this thesis contributes to the methods of interactive information retrieval by better regarding the real life context and by simulating users' characteristics in information retrieval test environments. Consequently, the more users' behavior is perceived, recognized and understood, the more user friendly and effective information retrieval systems may be constructed.

## Table\_of\_Contents

Acknowledgments.....	3
Abstract .....	5
List of original publications .....	9
Research contribution of the author .....	10
1. Introduction .....	11
2. Information Retrieval .....	16
2.1 Traditional Information Retrieval .....	16
2.2 Interactive Information Retrieval .....	20
2.3 Relevance Feedback .....	25
2.3.1 Explicit Relevance Feedback.....	26
2.3.2 Implicit Relevance Feedback.....	27
2.3.3 Pseudo-Relevance Feedback .....	28
2.4 Applying Classification Methods for Relevance Feedback .....	29
2.4.1 Classification and Clustering Methods Used in the Present Thesis .....	30
2.4.2 Term Space Reduction Algorithms .....	32
2.4.3 Learning to Rank vs. Relevance Feedback Classification Approach.....	33
3. Simulation of Interactive Information Retrieval .....	36
3.1 Introduction to Modeling and Simulation .....	36
3.2 Modeling Behavioral Factors in Simulation .....	39
3.2.1 Fallible User Modeling for Relevance Feedback Simulation.....	40
3.2.2 Query Modification Strategies.....	43
3.2.3 Scanning and Assessment Behavior .....	44
3.2.4 Modeling Frustration .....	48
3.3 Session Simulation .....	48
3.3.1 Search Environments .....	48
3.3.2 Cost Aspects .....	51
4. Evaluation of Interactive Information Retrieval .....	53
4.1 Rank-Based Evaluation .....	53
4.2 Time-Based Evaluation .....	54
4.3 Statistical Methods .....	55
5. Summary of Contributed Studies .....	57



5.1 Study I: Effectiveness of Search Result Classification based on Relevance Feedback .....	57
5.2 Study II: Simulating Simple and Fallible Relevance Feedback.....	59
5.3 Study III: Time Drives Interaction: Simulating Sessions in Diverse Searching Environments .....	62
5.4 Study IV: Modeling Behavioral Factors in Interactive Information Retrieval .....	65
5.5 Summary of Findings.....	68
6. Discussion and Conclusions .....	69
References.....	78
Appendix.....	83

# List of original publications

This thesis consists of a summary and the following original research publications, reprinted here by permission of the publishers.

- I. Baskaya, F., Keskustalo, H., & Järvelin, K. (2013a). Effectiveness of search result classification based on relevance feedback. *Journal of Information Science*, 39(6), 764-772. doi:10.1177/0165551513488317
- II. Baskaya, F., Keskustalo, H., & Järvelin, K. (2011). Simulating simple and fallible relevance feedback. In: *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, (pp. 593-604), ECIR 2011, Springer. Berlin Heidelberg. doi:10.1007/978-3-642-20161-5\_59
- III. Baskaya, F., Keskustalo, H., & Järvelin, K. (2012). Time drives interaction: Simulating sessions in diverse searching environments. In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 105-114), SIGIR 2012, ACM. Portland, Oregon, USA. 105-114. doi:10.1145/2348283.2348301
- IV. Baskaya, F., Keskustalo, H., & Järvelin, K. (2013b). Modeling behavioral factors in interactive information retrieval. In: *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*, (pp. 2297-2302), CIKM 2013, ACM. San Francisco, California, USA. doi:10.1145/2505515.2505660

These publications will be referred to as Studies I-IV in the summary part of the thesis.

# Research contribution of the author

First of all, this work and the present author stand on the shoulders of giants, who pioneered the way by creating the wonderful domain of Information Retrieval. Without the invaluable contributions of the supervisors, this thesis would not have seen the light of day.

In Study I, the present author created the research questions; collected and analyzed the data, designed and developed the necessary programs, implemented them, and then analyzed, compared and evaluated the research results, prepared the results for the publications, and wrote the articles with the help of co-authors.

In Study II, III, and IV, the present author contributed to the creation of research questions, collected and analyzed the data, designed and developed the necessary programs, implemented them, and then analyzed, compared, and evaluated the research results, prepared the results for the publications, and wrote the articles with the help of co-authors.

Prof. K. Järvelin and Dr. H. Keskustalo acted as the supervisors of the present author.

# 1. Introduction

Information Retrieval (IR) (Manning et al., 2008; Ricardo, 1999) is indispensable to our modern knowledge-based society. Modern information environments are becoming large and complex as well as ubiquitous, because the amount of available heterogeneous information grows exponentially each year (Alpert & Hajaj, 2008). Almost every aspect of our lives and every profession are affected by the information available on the Internet.

In order to gain knowledge, information should be obtained and analyzed by information users. In the first place, users should explicate their information need in a predefined way for a certain information retrieval system. Many different information objects such as documents, news, tweets, pictures, videos, audios, maps and 3D structures require various information need representations, not to mention the representation of those information objects in computer systems. However, one well-established communication medium is based on natural languages, or more precisely on the representation of words. Not only can documents, news, tweets, etc. be represented as text, but also other types of information objects such as audio-visual elements like pictures, videos and music can be described with words, which alleviate the possible problems of representation and access of those information objects. Therefore, information retrieval based on textual documents plays a major role in the research community and in real life. Consequently, the present research focuses on text-based document retrieval.

The history of document retrieval goes back to library science, where the documents were cataloged and accessed via catalog cards (Ruthven & Kelly, 2011, pp. 1-14). The categorization of documents was carried out according to salient features like title, author, publishing date and limited number of content keywords. However, emerging computer systems paved the way for automatic indexing of the full content of documents. Having all the content indexed, users were able to access and search appropriate documents according to their information need through search engine user interfaces. At first, Cleverdon et al. (1966) set up an

experimental environment for IR experiments, in which documents were indexed by content features and retrieved via queries, and then evaluated in a batch mode. This experimental setup is better known as Cranfield IR evaluation, sometimes also called the laboratory IR, which can also be described as system-centered/oriented IR. However, while system-oriented IR focuses on performance and effectiveness, designing a good IR system depends not only on system-oriented performance issues but also on understanding the users who interact with the system (Ingwersen & Järvelin, 2005, pp. 111-258).

Research and industry efforts in IR bifurcate into two areas; the first being system-oriented research and development, and the second a user-oriented, academic research field. Most of the effort in research and development is spent on system design, development and evaluation. Even though these evaluation efforts take the user into consideration by including predefined relevance judgments and diverse evaluation methods, they are limited in nature. As humans are diverse, so are IR system users. Accordingly, the interaction of the user with IR systems exhibits miscellaneous behavior, which is lacking in the design and implementation of the pertinent systems. On the other hand, conducting comprehensive user studies is not only intricate but also prohibitively expensive. Unsurprisingly, academic studies on user-oriented IR usually employ a small number of users in their studies. This in turn confines the expressive power of those studies in terms of the generalization of hypotheses claimed. To bridge between system-oriented and user-oriented IR, in the present thesis we simulate the user characteristics in respect of information retrieval interaction. Thus, we not only circumvent the peculiarities of the individual user characteristics, but also enable the system-oriented IR to respect the user behavior and improve the capability of IR systems to utilize an enormous number of simulated users in laboratory experiments.

Real life information retrieval takes place in sessions, where users search by iterating between different subtasks through an interactive interface (Marchionini, 1995, pp. 27-60). As an overly simplified view, after examining results, users either modify the initial query or supply relevance feedback (RF) (Ruthven & Lalmas, 2003), which means users give feedback to the search system by indicating the relevant documents from the result list and continue the session until the information goal is achieved or the session is abandoned because of frustration or

lack of time. Thereby, questions arise such as how relevance feedback can be utilized in the search system, how RF affects the IR performance when fallible feedback is provided, where the limits of effectiveness of diverse interactive searching strategies in different searching environments under overall cost constraints are, what kind and how effective the optimal sessions are under varying goals and constraints, and human stochastic behavior.

Psychological and/or social aspects of user behavior can be simulated in experimental designs according to experiment design (Ruthven, Lalmas, & Van Rijsbergen, 2003). Germane aspects of users should first be characterized to be exploited in the simulation. Among others, user's relevance feedback, fallibility in user's feedback, user's behavior under time pressure, endurance and scanning strategies in result scanning, and query modifications strategies in sessions are some of the simulated aspects in the present thesis.

The present thesis focuses on the simulation of user behavior and its effects on IR evaluation, and aims to answer research questions related to relevance feedback and multi-query session evaluation.

In previous RF studies, RF has been used to learn better queries in order to improve search result rankings after user's feedback. Those studies utilize query expansion methods to create better queries, which are consequently executed by the retrieval system. Instead of query expansion methods, we are interested in applying classification algorithms to improve result rankings without executing any further expanded queries. Accordingly, in Study I the main research question is: given RF on the first result page, assuming ten document surrogates are shown to a simulated user, is it possible to learn effective classifiers for the following result pages? Furthermore, we query issues such as how this novel classification approach depends on initial query length and how the effectiveness of this approach depends on diverse classification methods and term space reduction algorithms, which attempt to sort out the insignificant document terms.

Traditional RF studies assume perfectly correct RF, which means users are required to identify relevant documents in the initial results. In Study II we challenged that point and exercised progressively less perfect RF. This was motivated by the user studies, which expose fallible user behavior during RF (Vakkari & Hakala, 2000). Consequently, in Study II the overall research question

is: how does RF affect information retrieval performance when short initial queries, which are one to three words long suggested by real persons, are employed and fallible feedback, assuming that users may err when they indicate the relevant documents, is provided? Further, we are also interested in finding out whether mistakes in RF affect the quality (relevance level) of the documents found.

In real life, users interact with retrieval systems on different devices such as smartphones, desktop computers or tablets. Again, these devices lend themselves differently regarding user interaction. This in turn affects the time users spend in order to achieve search goals. However, time aspects of retrieval results have not been considered in commonly applied evaluation methods. Besides, some users are fond of having only highly relevant documents, while others would be perfectly satisfied even with marginally relevant documents. Consequently, varying search goals and time constraints encourage us to find out their effects on IR evaluation. Hence, in Study III we explore how various devices affect information retrieval sessions under overall time constraints and what the proper evaluation methodology is when time is taken into account. Moreover, we also explore all the search strategies which are query sequences applied during a search session, in order to find the best and worst sessions and compare them to query patterns frequently observed in real life.

Classical studies assume an average user, who interacts with a retrieval system in a predictable and regular way. However, users are diverse and not always predictable. Moreover, because of numerous reasons, they can make mistakes such as skipping relevant documents when examining a result list. Thus, in Study IV we analyze what kind and how effective the optimal search sessions are under varying search goals and time constraints, provided that both ideal and stochastic human behavior is regarded. In addition to the simulation variant in Study III, we further elaborate the search process with more detailed subtasks. With ideal human behavior we mean that users make no errors during the search process, or to be more precise, users scan all documents one after another, click every relevant document without making any judgment errors, read them and judge their relevance correctly. In contrast, fallible human behavior means that users may well err during the search process, in other words they may skip some relevant documents, read non-relevant ones, judge them as relevant or judge the relevant ones as non-relevant by mistake.

Besides, we are methodologically interested in the simulation of a behavioral model based on comprehensive session subtasks and fallible human behavior.

The rest of this thesis is organized as follows. Chapter 2 briefly introduces Information Retrieval (IR), while Chapter 3 addresses the simulation of Interactive IR (IIR). In Chapter 4 the evaluation issues in IIR are handled. The summaries of the contributed studies are presented in Chapter 5. Chapter 6 discusses the results, draws conclusions and proposes future research.



## 2. Information Retrieval

Human information behavior consists of phenomena such as information needs, information seeking, searching, browsing, finding, judging, usage, communication, sharing, transfer, management, information habit, and information style, which in brief means any information-related human behavior (Ruthven, 2008).

Human information behavior can be modeled in numerous ways with a focus on different aspects. The history of IR has witnessed many such models, which are still valid and consider various aspects from diverse point of views. Ruthven (2008) and Toms (2013) discuss some of those models. In general, these models lay out the information landscape which characterizes human information behavior.

On the other hand, interactive information retrieval models are formed to describe information retrieval interaction, which is the focus of this thesis. Among others, we can list some significant and salient ones like Belkin's anomalous state of knowledge (ASK) (Belkin, 1980), Ingwersen's cognitive model (Ingwersen & Järvelin, 2005), Saracevic's stratified model (Saracevic, 1997), and Bates' berry-picking model (Bates, 1989).

In this chapter, we first introduce traditional information retrieval, and then describe interactive information retrieval. The third section discusses relevance feedback, which can be categorized into explicit, implicit and pseudo-relevance feedback. Finally, we discuss some common classification methods that we applied for RF in the present thesis.

### 2.1 Traditional Information Retrieval

Information retrieval systems store and manage information items, e.g., text documents, as well as enable users to access them efficiently. With traditional Information Retrieval (IR) we mean system-oriented IR, which focuses on documents and document collections, matching algorithm(s) to retrieve relevant

information items to stated queries, and relevance judgments about documents in relation to queries.

Figure 1 depicts the traditional IR process, which is also called the laboratory model of IR. Figure 1 is adapted from Ingwersen and Järvelin's (2005, p. 5) schematized system-oriented IR Model. The main focus of the system-oriented approach is the representation of documents and search requests as well as their matching process. The user's involvement is confined to relevance and possible feedback judgments. Moreover, the relevance judgments of documents were created once by persons who may be developers of the experimental environment. In this view of IR, documents are represented and stored in a database corresponding to the applied retrieval model. Thereafter, the user's information need is translated into a search request, which is in turn represented as a query for the matching process. However, neither the task, which causes the user's information need, nor the user's real information context is taken into account in any way. Nevertheless, the matching algorithms deliver more or less relevant documents according to the match between the presentations of documents and query. At this stage it is possible to exercise feedback and modify the query. Results can now be evaluated by comparing the output documents against the recall base via diverse evaluation measures (e.g., Demartini & Mizzaro, 2006; Su, 1992).

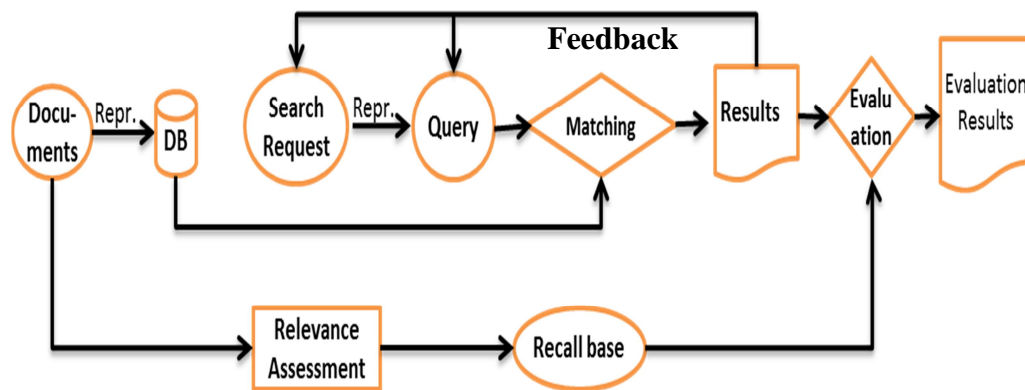


Figure 1. System-oriented view on IR (adapted from Ingwersen and Järvelin 2005, p. 5)

Retrieval effectiveness is dependent on the selected retrieval model. Belkin and Croft (1987) classify retrieval models into two main branches, namely exact and partial matching models. Exact matching models are developed on the basis of Boolean algebra. For this type of model, queries are meticulously constructed with

the help of Boolean operators. Boolean logic-based retrieval models only deem those documents which exactly match the Boolean query as relevant, for example for the query “Information AND Retrieval”, both keywords must appear in the relevant documents. Consequently, this model is robust but very strict, i.e., there is no possibility to obtain partially matching documents. Moreover, the delivered results list is not ordered according to relevance, though some ordering criteria like date or author’s name can be enforced. Even if Boolean systems seem to be obsolete nowadays, still some specific domains like legal domain can require recall-oriented retrieval, which can be provided by Boolean systems.

On the other hand, in order to allow partially matching documents to be listed on the results list, partial matching models such as vector space models (VSM), probabilistic retrieval models and more recently probabilistic language models have been developed (Croft et al., 2010, pp. 233-296; Manning et al., 2008, pp. 109-134 & 219-252).

VSM was first realized in Salton’s Smart information retrieval system (Salton, 1970). VSM represents both documents and queries as vectors in multidimensional space, whose dimensions consist of keywords. Every vector representing documents and queries can be built up with term weights like *tf.idf* (term frequency multiplied by **i**nverse **d**ocument **f**requency) (Belew, 2000, pp. 96-97) in respective documents and queries as axis values in each pertinent dimension. The similarity between a document and a query is calculated, for example, with the cosine similarity measure, which gauges the angle between two vectors. Then the documents can be ranked according to the cosine values in descending order. VSM is based on vector algebra, and is therefore mathematically founded, whereas its applicability in IR may be arguable from the justification point of view.

Probabilistic retrieval models (PRM) (Croft et al., 2010, pp. 233-296; Manning et al., 2008, pp. 219-236) are based on probability theories, especially the probability ranking principle, which means ranking by the decreasing probability of relevance of documents to a query. Documents can be ranked by the proportion of the probability of relevance and the probability of non-relevance ( $\frac{P(R|D)}{P(NR|D)}$ ). PRM utilizes Bayes’ rule for replacing the posterior probability  $P(R|D)$ , the probability of relevance given to a certain document in the context of a current query, with the prior probability  $P(R)$  and the likelihood  $P(D|R)$ . Applying Bayes’ rule for both

probabilities (e.g.,  $P(R|D)$  and  $P(NR|D)$ ) transforms the above proportion to  $\frac{P(D|R) \cdot P(R)}{P(D|NR) \cdot P(NR)}$ . Because the prior probability of relevance and non-relevance  $P(R)$  and  $P(NR)$  are the same for all documents, they are just playing a scaling factor for document scores, they can be removed from the formula.

Because the real probabilities are unknown, the probabilities in the formula above are estimated by diverse probability estimation methods in many different PRMs. For example, in the binary independence model (BIM), independence of the terms is assumed and term frequency in documents is taken into account simply as a binary feature. Hence,  $P(D|R)$  is estimated as a product of the probability of presence of a term and the probability of absence of a term in relevant documents. Given a query, the score for a document is the proportion (likelihood ratio) of the products of the term probabilities for all matching terms for relevance and non-relevance, which is usually converted to the sum of logarithms of term weights, because of mathematical precision concerns in computer memory systems. Because initially no relevant set is known, the pertinent probabilities are often set to a constant. Finally, when the proportions of the probabilities represent the term weights in a document, the similarity to the VSM model will be obvious.

Yet another probabilistic model introduced to the IR community is borrowed from language technologies. Language models (Manning et al., 2008, pp. 237-252) are applied in speech recognition, machine translation, spelling correction and other domains. Language modeling is based on probabilistic language models, which are estimated for every document in a collection. In order to rank the documents, document models are utilized to calculate the probability of generating the query. In language modeling, finite automata are exploited, for example, to generate the probabilities instead of generating strings for a language.

The probability of a query can be decomposed into the probability of each successive keyword conditioned on earlier keywords. Language models in IR are usually built from a single document. Therefore, there is not enough data to model complex conditional probabilities. The simplest possible language modeling, namely unigram language modeling (Manning et al., 2008, pp. 237-252), assumes the independence of terms; hence the probability formula is reduced to a probability calculation without a conditioning context. Assuming unigram language modeling, the probability of a query, especially in the query likelihood model, can simply be

generated by multiplication of the probability of each query term with the help of the language modeling of the pertinent document. In case the query word is missing from the document language model, the probability of that query word will be zero, which requires special handling, or ‘smoothing’ (Zhai & Lafferty, 2001). Smoothing not only adds a fraction of probability to every word, but also discounts the non-zero probabilities. Having calculated these probabilities, documents can be ranked accordingly in descending order. As mentioned above, during implementation the multiplication of small numbers is replaced with a summation of logarithmic values; because the logarithmic function is a monotonic function, the ultimate order does not change, which is important for ranking the documents correctly.

In the current thesis we employed the Lemur/Indri search engine (Strohman et al., 2005) to conduct the experiments, because it is one of the very effective retrieval systems available as open source software. The Lemur/Indri search engine allows searching based on language modeling. Moreover, smoothing via Dirichlet priors (Zhai & Lafferty, 2001) is applied in order to avoid mathematical conundrums caused by the terms that do not exist in documents.

## 2.2 Interactive Information Retrieval

Ingwersen and Järvelin (2005, pp. 313-357) depicted an evaluation framework for interactive information retrieval. They tried to set IIR into various self-contained contexts, which resemble a Russian matryoshka doll. The framework starts from the socio-organizational and cultural context and ends at the IR context. Thereby, IR effectiveness can be evaluated in the socio-organizational context with socio-cognitive relevance, in other words, the quality of work task results. If the socio-organizational context is opened, the work task context appears. Again, IR effectiveness can be evaluated by the quality of information and work process or results. Under the work task context, the seeking context is located, in which IR effectiveness can be evaluated by the quality of the seeking process. The final context represents the IR context, in which the effectiveness of IR systems is traditionally evaluated by measures based on recall, precision, and efficiency, for example. Moreover, search engines, which are the core of the IR context, can be evaluated among others by the effort in using criteria such as comprehensibility of

interface and query language, support in query formulation, and results presentation. In other words, these evaluation criteria need a user's subjective opinion. However, existing methods and measures for the evaluation of systems are user-agnostic. In the present thesis we try to close the gap between traditional IR and user studies by respecting the user characteristics such as his/her fallibility of feedback (i.e., pinpointing relevant information), frustration and context in time, space, and user's search goals (see Figure 2. Dimensions for extending IR ) (Kamps et al., 2009).

In real life, users do not usually have ready-made topics or queries at their disposal. Instead, they are confronted with a problem or task, which is the source of the users' information need. This motivates them to seek information about the task or problem at hand. Thereby, different users cope with diverse information sources and with their access methods differently. Moreover, users' problems or tasks can be either work or leisure-related (Vakkari, 2003). This often sets some time and /or goal constraints. In addition, users have miscellaneous traits and backgrounds such as education, gender, domain knowledge, and perseverance which affect how they conduct the search process with an information retrieval system. In addition, the effect of numerous search interfaces of ubiquitous computer systems such as desktop computers, mobile phones, tablets and even smart television sets also have an influence on the search process. However, all these variables are not, at least directly, taken into account in the design and evaluation of interactive information retrieval systems. For example, system-oriented IR experiments, which hardly represent IIR experiments and systems, assume a user model representing ideal users, single query sessions, well-defined topics and queries, topical relevance, and document independence in order to design, develop and evaluate a user's information retrieval process.

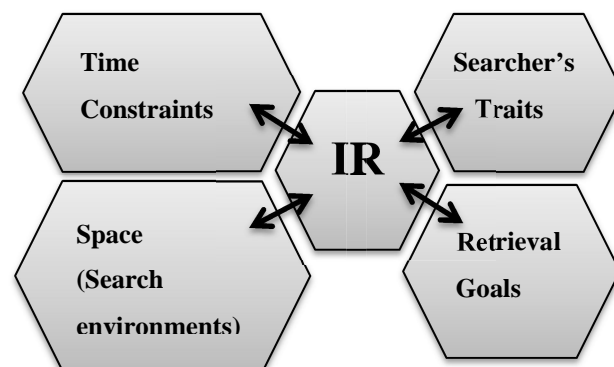


Figure 2. Dimensions for extending IR evaluation experiments

Keskustalo (2010) described six major limitations of traditional IR in his thesis:

1. No explicit user modeling
2. Single query sessions
3. Well-defined topics and queries
4. Topical relevance
5. Document independence
6. Challenge of traditional evaluation

The first limitation was about explicit user modeling, which means that traditional IR lacks an explicit user modeling which represents an individual user with a certain mental state, learning capability, educational background, and gender, inability in many respects, and a dynamic view of relevance (Kelly, 2009). However, taking all these user attributes into account would also complicate the evaluation process and even make it difficult to compare the results of different systems. Nevertheless, users in real life are different and exhibit different searching behaviors, and the systems should be evaluated accordingly. Therefore, we suggest explicit user modeling, which takes the personal traits and backgrounds of the users into account. In our studies we have incorporated users' search and query formulation strategy, recognizing the relevant snippets and documents, perseverance as frustration, and time and goal dependent behaviors, among other things.

The second limitation of traditional IR was about the single query sessions, which is justified from the batch processing point of view of system-oriented IR. However, users do not have all the query words initially at hand when they start to pose the query to a search system, let alone the search topic (Marchionini, 1993). Users learn while searching. Consequently, users often pose multiple queries during a search session. Another reason that users modify the query is the ambiguity of the query, e.g., in cases where homonyms are used. A search session ends when either the user's information need or goal is satisfied, or users give up because of frustration due to unsatisfactory results or a lack of time. Accordingly, multiple query sessions with several query modification or formulation strategies are simulated in our studies to explore the effects of multiple query sessions.

Third, well-defined topics and queries were a limitation mentioned by Keskustalo (2010). As a relic of Cranfield experimental design, topics and verbose queries usually constructed from topic description are quite common in IR experiments. Fixed information needs cast as topics and corresponding relevance judgments constitute a typical experimental setup for IR experiments. However, a user's

information need is fuzzy at the beginning of a search session, and the information need may crystallize itself during the search process. Moreover, a user's relevance perception changes as they learn from inspected snippets and documents.

In addition to that, users in real life often prefer short queries (Jansen et al., 2000). In this thesis we also employ realistically short queries produced by real people for topics, which are predefined for certain information needs with corresponding relevance judgments in standard test collections. Thus, even though we do not circumvent all the limitations about topics, queries and recall bases, nevertheless we bring human-generated and realistically short queries into research settings.

The fourth limitation was about topical relevance, which can be described by the relationship between a topic expressed in a query and a topic covered by an information object (Saracevic, 2006). According to Saracevic's (1997) relevance system, relevance can be motivational, situational, cognitive, topical, and algorithmic. For example, situational relevance affected by e.g., time pressure, and motivational relevance affected by e.g., commitment to task, or cognitive relevance affected by e.g., domain knowledge and expertise can easily affect the information retrieval performance. Consequently, there is a need to deal with all these types of relevance in IR evaluation. Therefore, we simulate not only the topical relevance but also some aspects of situational, motivational and cognitive relevance through constraints, goals, and fallibility.

Document independence was the fifth limitation in Keskustalo's thesis (2010), which means relevance judgments in a recall base are based on individual documents and their mutual effects are omitted. However, a user's relevance perception changes with every inspected document or snippet. Moreover, recurrent information in a result list can affect users' relevance judgment (Kekäläinen & Järvelin, 2002). In spite of this we do not abandon document independence assumption, especially since the recall base which we utilize in our simulations is constructed under this premise. Still, we applied result list freezing (Keskustalo et al., 2008; Ruthven & Lalmas, 2003) in the evaluation process in order to alleviate the effect of recurrent documents. Besides, setting an appropriate goal like "find one relevant document" (Sakai, 2006) also contributes to the elusion of the independence limitation.



The challenges of traditional evaluation are presented as the last limitation in IR in Keskustalo's thesis (2010), which means inadequate evaluation metrics from the user's point of view are applied to measure the outcome of IR experiments. However, user's costs for example in terms of time expended and the frustration of the user with a futile system should be taken into account. Indeed, we will discuss the risks of traditional metrics when time is considered as a component in evaluation. Apart from this, we will present a formula to describe the perseverance behavior of searchers in our simulations.

In addition to the six limitations described by Keskustalo (2010), we discuss the following novel issues in the present thesis. We show the effects of various search interfaces during the search process (Kamvar et al., 2009). While traditional IR usually assume a typical search interface for experiments, we considered different types of search interfaces and their effects on the search process in respect of the costs involved and the utility gained. Thereby, instead of the utility a search session produces, we set time constraints and gain goals and investigated the best-performing patterns which are governed by user habits and behavior. Not only are the prototypical patterns investigated, which are prevalent for typical users as earlier research (Vakkari, 2000) showed; we also investigated a comprehensive set of search patterns in order to find any better strategies, which can surpass the common prototypical strategy outcomes.

Even though users are tacitly integrated into traditional IR research with relevance judgments, users' personal differences are not really regarded. For example, users' understanding of snippets (Turpin et al., 2009) or users' relevance perception can vary. This can lead to accepting various less relevant or even non-relevant documents as relevant. In our studies we examine various behavior-related variables such as clicking a snippet and judging a document, which are then represented by various probabilities. Another traditional assumption about users is that they are perfect. However, as we know, to err is human. We examine the fallibility of searchers<sup>1</sup> during RF and scanning search results. Yet another characteristic assumption about searchers in traditional IR is their robust perseverance during result inspection. In fact, users may get frustrated, especially when they encounter unproductive search results. Thereafter, users either abandon

---

<sup>1</sup> The terms *searcher* and *user* are interchangeably used in the present thesis.

the search session altogether or formulate another query. Indeed, this is an issue where we look forward to formulate frustration as a skipping probability. It represents a user's perseverance during a search session.

## 2.3 Relevance Feedback

Before even trying to explain relevance feedback (Harman, 1992; Ruthven & Lalmas, 2003), the following question should be answered: What is relevance? Let us first describe relevance in the context of IR. First of all, relevance can be classified into five sub-categories. The first of these is system relevance, which is related to the order of documents produced by the retrieval system and users' request; the second is topical relevance, which represents the topical relationship between documents and queries; the third is cognitive relevance, which is related to the mental level of receiving pertinent information; the fourth is situational relevance, which takes into account the situations e.g., the time pressure the user is subjected to and the effort they expend to carry out their tasks; and the fifth motivational relevance includes the user's frustration and lack of accomplishments (Saracevic, 1996). Under this classification of relevance we can now utilize some aspects of these relevance types that are appropriate to the present thesis.

Furthermore, relevance is not a constant – it changes during an information retrieval session, because the better the information need is understood by the user, the more precisely the relevance of documents can be judged. It is also reasonable to see this the other way round, in that the information needs of a user changes during a search session, and this in turn affects the relevance judgments (Saracevic, 2007; Vakkari & Hakala, 2000).

In order to improve information retrieval effectiveness, relevance feedback (RF) can be utilized (Ruthven & Lalmas, 2003). RF means that an information retrieval system utilizes feedback supplied by users either explicitly or implicitly inferred from user's behavior on a search result page in order to improve subsequent search results and their ranking. IR systems can exploit the given relevance feedback in various ways, for example a frequently employed method is the use of query modification (QM) (Carpineto & Romano, 2012; Efthimiadis, 1996). Certainly, RF implementation depends on the information models used, as different models

require the integration of the RF in different ways into the model. However, a novel approach to applying RF is suggested in this thesis, which leverages the classification methods for RF by classifying the as-yet unseen results with the help of a classifier, which is trained on documents from RF.

There are two types of relevance feedback: explicit and implicit. Explicit RF requires explicit input from the retrieval system user, whereas implicit RF discovers RF information from the user's behavior and actions on the result page (White, 2011).

### 2.3.1 Explicit Relevance Feedback

Explicit RF requires explicit user feedback, which means users of IR systems should contribute the feedback information to the system in some form. However, there are many challenges to relevance feedback, starting with the cognitive load of the user. Depending on the user's mental capacity and current task difficulty, supplying RF can provide an extra burden on the users. Moreover, RF requires additional effort both from users and system developers. In spite of this, additional effort can be compensated with better ranking of the search results. Although RF is not usually realized as an essential part of a routine search process, additional effort can be justified with the reward of better search results. For example, Google offers a "similar" or "related articles" link, which appears from time to time either in the results snippet or in the results preview window, depending on the presented information object, for requesting relevant pages. Yet another difficulty could emerge with peculiar complex document types which users have to cope with. Complex documents, such as multi-topic or partially relevant documents, are those which can influence the outcome of the RF process. Such documents can accommodate both relevant and non-relevant keywords, so that they may cause query drifting in query modification realization of RF (White, 2011).

As mentioned briefly in the previous section, the nature of relevance judgments makes the relevance feedback harder. Another aspect of relevance is its partiality. As the majority of relevance assessments are based on binary decisions, a document is either relevant or non-relevant (Voorhees & Harman, 2000). However, if the complexity of the documents and information needs are considered, it can be very

quickly understood that the binary approach is not sufficient for judging the relevance of documents.

Instead of binary judgments, Sormunen (2002) created a graded relevance-based collection, in order to study the effect of graded relevance on IR effectiveness. His graded relevance scale had four levels, namely highly relevant, fairly relevant, marginally relevant, and non-relevant. For instance, this graded relevance scale is employed in Study II to scrutinize the effects of fallibility at different relevance levels. In addition to the challenges mentioned above, relevance feedback depends on initial result ranking. This can be problematic because a user's information need or knowledge state changes, since users usually learn even from the snippets of the search results presented with every document surrogate. Moreover, users can only indicate relevant documents, if such documents appear on the result page. On the other hand, if the user's information need is already fulfilled through the first result page, the necessity of applying RF is dissolved (by itself). In Study I we analyzed when to apply RF based on the precision of first result page. Apart from these, users have to assess every document individually. If user's fuzzy information need, imperfect knowledge state, and difficulties ascribed by diverse user interface issues, e.g., misleading snippets, are taken into account, it is not a big surprise that user can err (Vakkari & Sormunen, 2004) during relevance feedback. Consequently, the human fallibility in providing RF is simulated in Study II to measure the effect of fallibility on RF effectiveness.

However, users are not always ready to supply feedback information to the IR system explicitly even if they have a positive attitude towards giving relevance feedback (White, 2011). In such cases, IR systems may exploit implicit relevance feedback, which can be beneficial for current web search engines (Joachims & Radlinski, 2007; White et al. 2002).

### 2.3.2 Implicit Relevance Feedback

Because explicit RF has time and effort implications for users of an IR system, they may be reluctant to provide RF explicitly. A remedial action lies in user behavior during interaction with the IR system. Implicit RF does not require explicit feedback action from users, so in lieu thereof user's interaction with the search system will be

observed and the user's information need will be prognosticated from user behavior. For example, user's dwell time on search result page and especially on some particular documents, saving or printing any result documents for further reuse, selecting, referencing or commenting any document could bring the required evidence for RF (Kelly, 2005; White, 2011).

Implicit RF is classified according to the user's intent by Oard and Kim (2001) into four categories: first, "examine" behaviors, where users read, listen, view or select a document; second, "retain" behaviors, where users bookmark, save, or print a document; third, "reference" behaviors, where users give a reference to a document or its parts by replying, linking, or citing; finally fourth, "annotate" behaviors, where users annotate the documents by marking up, rating, or liking. However, not all categories can be leveraged during an online search, because IR search systems have limited access to users' actions on document pages. Such limited access to user's actions on various pages can be gained by search systems via "like" buttons, advertisement banners or similar constructs to follow the user click behaviors. Moreover, implicit relevance can be combined with explicit relevance to increase the effectiveness of IR systems and to corroborate the understanding of users' needs by IR systems.

Furthermore, user profiles, which are based on users' search history and preferences, could also be created by the search system, and can be incorporated into the search result building process as a way of implicit RF.

Although explicit relevance in our experiments is assumed and simulated, relevance information could be harnessed by diverse user behaviors in practical operational environments. Nevertheless, certain aspects of user behaviors are brought into simulation settings. Those aspects of user behavior are discussed in more detail in Section 3.2.

### 2.3.3 Pseudo-Relevance Feedback

Even though the usefulness of explicit relevance feedback accomplished with the query expansion technique was reported by Ruthven and Lalmas (2003), user's reluctance to provide feedback remains a thorn in one's side. Consequently, implicit RF can alleviate this burden to some extent. Yet another approach to utilizing RF,

pseudo-relevance feedback (PRF) (Ruthven & Lalmas, 2003), assumes the top-ranked documents produced by an initial query to be relevant. PRF also avoids the user's explicit feedback. Before the first result page is presented to the user, which in case of explicit feedback is compulsory, PRF can hopefully be applied to improve search results before presenting them to the user. However, if the top-ranked documents are non-relevant, they cause query drifting, and can decrease the effectiveness of the IR system instead of increasing it. Nevertheless, IR experiments with PRF have shown that it improves the system outcome slightly. Moreover, to counteract the problem of query-drift, Järvelin (2009) and Lam-Adesina et al. (2001) introduced the query-based summarization of top-ranked documents in PRF and demonstrated the advantages of their approach. In addition to that, PRF can be combined with RF, at least after the first page is presented to the user to obtain implicit and/or explicit relevance feedback.

In Study I, we simulated a combination of PRF and explicit RF, in other words we applied our novel approach to the RF process on top of PRF results, and showed how this classification method can improve search effectiveness.

## 2.4 Applying Classification Methods for Relevance Feedback

Instead of selecting a conventional path to apply RF by expanding the initial query with additional keywords extracted from the relevant and non-relevant documents, or by modifying query terms weight (Ruthven & Lalmas, 2003), we opted for using classification and clustering methods (Sebastiani, 2002) to distinguish the relevant documents from non-relevant ones in subsequent results after RF provided by the user on the initial results. In order to train a classification or clustering algorithm, we assumed that a simulated user indicates the relevant and non-relevant documents on the first result page. The first result page was constructed by applying PRF to the initial query results, using the Lemur/Indri PRF algorithm (Strohman et al., 2005; Lavrenko & Croft, 2001); before the first result page is presented to the simulated user. Then the simulated user supplies the simulation program with relevance information based on the recall base of the collection used. Having trained the

algorithms with two sets of documents representing relevant or positive ones, and non-relevant or negative ones, and built the classification model, the important question is whether the search program is able to discern the documents as positive or negative ones further down in the result list. Thereafter, the negative documents (or non-relevant ones) from the result list are removed and the relevant ones are shifted forwards to vacant positions in the result list.

However, before even starting with the application of the classification or clustering the subsequent documents after the first page results, one should ask the question: When should be the RF applied? Are we able to improve the results with the help of classification? Is it necessary or possible to improve the second and third page results at all, when the first page results either already satisfy the user's information needs or have no relevant documents? In order to answer these questions, it is necessary to analyze the precision of the first and subsequent result pages.

The next section briefly summarizes the main ideas of classification and clustering methods which are used in the present thesis. Further, the term space reduction algorithms, which are utilized during the classification process, are described. At end of the section, Learning to Rank is compared to the proposed RF classification approach, because both approaches apply machine learning methods and are therefore related.

#### 2.4.1 Classification and Clustering Methods Used in the Present Thesis

The following classification and clustering methods are applied in the present thesis: Naïve Bayes classification method, K nearest neighbor (KNN) algorithm, KMeans algorithm and Support vector machines (SVM) classifier (Joachims, 1999; Manning et al., 2008, pp. 253-376; Sebastiani, 2002). These methods are selected because they are the most common and state-of-the-art methods in research and practice.

The Naïve Bayes classification method is based on Bayesian theory. This method tries to express posterior probabilities with prior probability and likelihood. Thereby, the probability of a document belonging to either the positive set or the

negative set is defined by the highest posterior probability, which any of the sets produces with respect to the document being classified.

Again, the naïve assumption is also made here and the independence of words from each other in a document is assumed. Even though the words in a topical text depend on each other, obviously this assumption does not harm the outcome of classification and/or retrieval process severely. The probability of a document belonging to a class can now be easily determined by multiplication of the individual probabilities with which each word occurs in the documents of each class. In addition, the prior probabilities can be estimated according to the proportion of documents in each class, namely positive and negative. Still, the probabilities of the words in pertinent classes should be estimated and they can be computed from the frequencies in the training sets. However, test documents can contain words unseen during the training phase, which results in zero probabilities. Therefore, smoothing functions are applied, which assumes small probabilities for every word. Laplace correction is one of the smoothing functions employed in the present thesis. In our experiments we preferred the multinomial model, where the frequency of the words in documents is taken into account. An alternative, the multivariate Bernoulli model, regards word frequencies as binary features, which means that the duplicate words are eliminated from document representations. The multivariate Bernoulli model produces competitive results where very short documents like tweets are classified (e.g., neglecting the word frequencies does not change much) (Manning et al., 2008, pp. 253-288).

The K nearest neighbor (KNN) algorithm first calculates distances between the test document and all training documents. Then KNN selects the  $k$  nearest neighbors, which are the  $k$  closest documents to the test document according to whatever distance metric is used. Then the class of test document will be decided by a majority vote of the selected documents. There are several distance metrics in the literature such as Euclidian distance, Minkowsky distance, Manhattan or City-block distance, and Canberra distance (Losee, 1998, pp. 43-75). These were also employed in our experiments. Finally, we measured the similarity of documents to one another with the Euclidian distance metric, because it achieved the best effectiveness in our experiments. Even though KNN is quite a successful classification algorithm, it requires comparison with every other document in the



collection of negative and positive documents, which happens on the fly after the test document is submitted. In comparison to Naïve Bayes, KNN does not build a model beforehand.

The clustering algorithm KMeans is also exploited for distinguishing between the relevant and non-relevant documents in our experiments. The documents on subsequent result pages are clustered with the help of the KMeans algorithm, which starts with the centroids of the clusters of the relevant and non-relevant documents pointed out by simulated user and tries to assign the documents to pertinent clusters by selecting the nearest cluster centroid. Similarly to KNN, nearest cluster centroids are determined by comparing the distances between the document and the centroid. After every document is assigned to any one of the either clusters, cluster centroids are recalculated with those documents in each cluster. This process is repeated until either the cluster centroid positions do not change anymore or the preset maximum number of iterations is attained.

A support vector machine (SVM) is the state-of-the-art classification algorithm, which separates data points, or the result page documents in our study, by means of a hyper-plane in a multidimensional space. First, SVM tries to find a hyper-plane which maximizes the distances to the nearest training data points, or rather documents, of two classes. These nearest data points to the hyper-plane are named support vectors, which play a major role in developing the training model; all the rest of the training documents will be discarded after determining the hyper-plane. After building the training model, or defining the hyper-plane, test documents can be readily projected to either side of the hyper-plane, which determines the appropriate class (Joachims, 1999; Manning et al., 2008, pp. 319-348).

## 2.4.2 Term Space Reduction Algorithms

The purpose of applying reduction is either performance improvement or the reduction of noise introduced by high dimensionality. Reducing the number of features also reduces the number of calculations, which would otherwise be computed for the disregarded features. This in turn contributes to the performance of the classification algorithm. On the other hand, one should also notice that term space reduction methods consume processing power. The benefit of term space

reduction may be the avoidance of noise introduced by high dimensional feature space. However, some of the classification methods like SVM can cope with the high dimensions very efficiently by regarding only the decisive features.

In order to reduce the number of dimensions in our experiments, we experimented with the following methods: mutual information gain, Kendall-Tau rank correlation coefficient, Pearson's chi-squared test, odds ratio, Spearman rank correlation coefficient, and Fisher's exact test (Banerjee & Pedersen, 2003; Sebastiani, 2002).

The term space reduction methods employed in the present thesis are defined in the appendix.

### 2.4.3 Learning to Rank vs. Relevance Feedback Classification Approach

Web search phenomena triggered the ranking studies in order for the search engines to better serve search results for web search engine users (Li, 2011). However, ranking is not only confined to document ranking in web search, but is also applied in collaborative filtering like product recommendation, machine translation, and meta-search, which aggregates search results from several search systems. Nevertheless, we focus on the document ranking creation in this thesis. Initially unsupervised ranking models or rather ranking formulas like BM25, Language Model of IR and PageRank formula (Manning et al., 2008, pp. 219-252 & 461-482) and their combinations have been exploited in the evolution of search systems. These algorithms are unsupervised because they do not require any training phase with labeled data, even though some collection-dependent parameters should be gleaned first. These algorithms extract some features from queries and documents in the collection in order to calculate a score for the documents for a given query. There is a number of features, among others term frequency, BM25 scores, and edit distance for various parts of the documents like title, anchor text, URL, and body, as well as the number of incoming links and PageRank score of the document or page (Qin et al., 2010). Moreover, implicit RF based on click-through data or explicit RF features can also be collected in order to be employed in these algorithms.

However, the static way of determining the document or page score by these methods or their combinations can be improved by machine learning approaches. These machine learning approaches include classification and regression. In a broad sense, any machine learning algorithm applied to ranking can be called a Learning to Rank algorithm. On the contrary, a narrower definition of Learning to Rank is associated with the machine learning methods for constructing ranking models in ranking creation and ranking aggregation. The former creates rankings; the latter aggregates the rankings of different systems. Learning to Rank aims to create better rankings for search results. The most relevant documents will be placed on the top positions in the result list. Learning to Rank methods first create a ranking model out of training data, which is a collection of queries and respective documents labeled as relevant. Then the most relevant documents for future queries are retrieved by engaging that particular ranking model (Li, 2011).

The studies on Learning to Rank have produced plenty of different methods. In the main, there are three major approaches to learning a ranking model: *pointwise*, *pairwise* and *listwise* approaches. The *pointwise* approach considers the request and the respective documents individually. Naturally, both requests and documents are represented by feature vectors, which are involved in building the ranking model. The group structure between the request and relevant documents is omitted. This approach transforms the ranking problem into a classification, regression, or ordinal classification problem. For example, a learned ranking model can produce scores, e.g., real numbers in case of regression, for documents with respect to a query; thereafter documents can be ranked according to scores. The *pairwise* approach converts the ranking problem to pairwise classification or regression. Likewise, the pairwise approach ignores the group structure between request and documents. A classifier decides the ranking order of document pairs. On the other hand, the *listwise* approach takes the group structure into account, in other words, the ranking lists as whole are utilized both in the training and prediction phase. A ranking model ranks the documents according to scores which are reckoned by the ranking model. The *listwise* approach requires new methods because the existing machine learning techniques cannot be directly employed (Li, 2011).

Besides these major approaches, there are query-dependent and multiple nested ranking approaches. Furthermore, there are many diverse implementations of

Learning to Rank methods, some of which are also employed by commercial search engine companies. For example, the pairwise approach LambdaMART performed best in the Yahoo Learning to Rank Challenge (Chapelle & Chang, 2011).

In a broad sense, our classification approach for applying relevance feedback may be seen as a Learning to Rank method, because we employ a machine learning technique for re-ranking the search results and we have the same goal as Learning to Rank methods. However, Learning to Rank methods principally try to build a single model from training data, which will be exploited for future similar queries. In contrast, our approach builds a classification model based on either implicit or explicit relevance feedback after user's individual query and examining the very first result page and applies this particular model only for the rest of the results, which have already been collected for this specific query.

## 3. Simulation of Interactive Information Retrieval

Simulations are based on models. A model represents the phenomena that will be replicated in simulations and further captures the essential components and interactions. In this chapter, we first give an introduction to modeling and simulation, and then describe modeling behavior factors in simulations. Finally, we discuss search environments and cost aspects in session simulation.

### 3.1 Introduction to Modeling and Simulation

Before IIR simulation, which is the main focus of this thesis, is described, modeling and simulation are introduced generally in this section. Simulations add one more knowledge building tool in addition to the theoretical and experimental tools. Simulations can be exploited to gain insight, validate models and experiments, predict the potential outcomes of system changes, test and evaluate systems, among others (Sokolowski & Banks, 2011, pp. 25-43). In other words, simulations are executed to conduct *what-if* analyses. These *what-if* analyses in turn can contribute to gaining insight on the one hand and solving problems on the other. Problem-solving simulations incorporate less uncertainty, whereas gaining insight simulations are naturally plagued with more uncertainty, because the models used in these kinds of simulations are neither complete nor even accurate with respect to reality, which is simulated and investigated. The more insight is gained, the more accurate the models, and consequently the simulations, become. As a consequence the gaining insight simulation models are ephemeral by nature. However, problem-solving simulations can serve for longer periods, because these simulations are usually parameterized and have a stable model. The following type of questions, which are given in the simulation book (Sokolowski & Banks, 2011, pp. 25-43), can be answered by problem-solving simulations: *What would happen if...?*, *How will*

*a...?, Why would a...?, Can a...?, Does the...?, Should we...?* On the other hand, the gaining insight type of simulations can answer the following questions: *What has the greatest influence? How will X and Y interact? Is there a way to make X happen? Why has unexpected behavior X occurred? What new behaviors might emerge?*

By nature, the simulations in the present thesis are of the gaining insight type, and those general questions can be specified such as: What has the greatest influence on cumulated gain in a session? What is the influence of user interface on cumulated gain in a session? How will gain and time in an information retrieval session interact? Is there a way to achieve the best gain? Why do the traditional evaluation metrics deliver unexpected results when time is taken in to account? What kind of new behaviors for search process might emerge?

First, a simulation model represents a real event, phenomenon or system, which will be simulated and analyzed, and is expressed in a formal way, usually mathematically or as a computer program. Therefore, a model should approximate the real event or system closely, and reflect the important features of the real world from the pertinent aspects of the simulation. However, incorporating every feature of the real world into a model not only increases model complexity but also makes the simulations infeasible. Consequently, some of the salient features are selected for model building and the rest are ignored. The balance between realism and the simplicity of the model is one of the critical decisions the model builder has to face. Too much simplicity diverts the simulation from reality, which may cause drawing the wrong conclusions. On the other hand, too much realism may cause computational difficulties, e.g., either excessive requirements for memory and/or processing time, which in turn makes simulations prohibitively expensive.

Models can be classified on the one hand as *static* or *dynamic* with respect to time, and on the other hand as *deterministic* or *stochastic* with respect to input or output variables. While dynamic models take time into account, static ones do not. Similarly, the stochastic ones model the probabilistic values for input or output variables, whereas the deterministic ones regard those variables as fixed. In Studies II, III and IV, we utilized the following model types, namely static, dynamic, deterministic, and stochastic models (Maria, 1997).

Before a simulation based on a model is further run, it should be verified and validated. While verification ensures that the model complies with its specification, validation enforces the validity of the model, which means that the model imitates the real event or system genuinely (Altiok & Melamed, 2007, pp. 1-10).

After a part of reality is modeled and formally expressed as a model, the model can be executed or, in other words, simulated in an environment, usually on a computer. Simulation allows input variables of the model to be changed and the execution of the model to be repeated, and then the outcome of the simulation experiment to be analyzed. In this way, simulations can shorten or extend the real time of a real event into a simulated time, which can be much shorter or longer. Hence, simulations empower us to test and analyze hypotheses about a real system in a timely manner, which can save enormous costs and efforts in comparison to real setup. In some situations real events cannot even be repeated easily or are almost impossible to repeat, for example because of side effects, which in turn affect the outcome of the real experiment. For instance, the information retrieval system users learn during searching in IR experiments, therefore the same task cannot be assigned to the same user for analysis of variations of the various task variables.

The simulations can be realized either as stand-alone programs, which run independently, or as integrated simulations, which are embedded into the real system. The stand-alone simulations can be classified according to application areas, for example: Training, decision support, understanding, education & learning, and entertainment. Further, the simulations can be classified according to the user point of view, namely users or the researcher. While plain simulation users are more interested in problem-solving issues which they encounter day by day, for example during training, decision-making or entertainment, researchers rather try to gain insight into peculiar problems (Sokolowski & Banks, 2011, pp. 25-43). The simulations performed and analyzed in the present thesis can be regarded as stand-alone and simulations for understanding, because they facilitate hypothesis testing about the user behavior in IIR.

The following two sections address human factors in simulations, especially IIR simulations, and IIR session simulation. The first section models the fallible user during RF, further defines query modification strategies, and describes the scanning and assessment behavior, which are employed in the experiments in the present

thesis. Finally, the scanning strategy of search results and the frustration of the user during search result scrutinizing are delineated. The second section specifies the simulation environments, and describes the cost factors in simulations.

## 3.2 Modeling Behavioral Factors in Simulation

One of the main focuses of this thesis is the modeling of human behavioral factors for IIR simulations (Azzopardi et al., 2011; Clarke et al., 2013). In order to achieve a realistic behavior representation, the observation of human subjects during information retrieval interaction is indispensable. An information need causes searchers to initiate a search process, which often consists of multiple queries in a session depending on the type of the search and the availability of documents. While traditional IR typically assumes a long query with persistent scanning of a long list of search results, we simulate more complex sessions based on user interaction with an IR system. Thereby, sessions may consist of multiple queries and/or user feedback. For the former, the searcher poses multiple queries one after another, and for the latter the searcher may give some feedback to the IR system, which utilizes the feedback to improve the search results before presenting them as an enhanced result list to the searcher. As one might expect, the searcher typically scans the search results and assesses the quality of the snippets before clicking the document links. After inspecting the respective document, the searcher then assesses the relevance of the document and judges whether the document fulfills their information need at least partly. This complicated process can be dissected into subtasks like scanning a snippet, clicking a document link, assessing the relevance of a document, and reformulating a query. These subtasks represent certain actions users apply during a search process. Each subtask is associated with certain effort that users expend. This effort may have many aspects and depends on the strategic decisions of the searchers. One way of measuring the effort can be realized in terms of time, which users spend performing the particular subtasks (Azzopardi, 2011).

However, searchers are human, and as we know, humans are fallible. We continuously make mistakes in every phase of the search process. Starting with misunderstanding the (work) task, searchers may type in misspelled search keys, or judge the relevance of the snippet or document incorrectly and consequently may



give ineffective feedback. In spite of human fallibility in IIR, there is little research that takes the consequences of fallibility into account.

In this section, we first define the fallible user models for RF simulations to evaluate the effects of fallibility in RF experiments, whereby fallible humans are simulated according to some probability distributions. Second, we investigate the effectiveness of query modification strategies observed in real life. Then, we analyze the scanning and assessment behavior and their characteristics in terms of deterministic and stochastic ways. Finally, humans have limited scanning endurance, especially when the search results are of low quality; they either reformulate their query or give up the search session altogether. In order to model this human aspect, we discuss a novel frustration formula in the last subsection, which models the user's dedication to a search session.

### 3.2.1 Fallible User Modeling for Relevance Feedback Simulation

Modeling users for RF simulation requires several considerations regarding users' readiness to browse the initial search results and to give feedback, the level of relevance of the RF documents and users' fallibility during relevance judgments about documents. The first three points are addressed by Keskustalo and colleagues (Keskustalo, 2010; Keskustalo et al., 2006) by defining a user model. However, the last point, fallibility during relevance judgments, is a novel approach to user modeling in RF simulation. The motivation for this point comes from the literature, where for example Turpin and colleagues (2009) and Vakkari and Sormunen (2004) discovered erroneous relevance judgments of searchers.

Because we simulate the relevance judgments of users without resorting to their real judgments, we look for an alternative source of relevance judgments. Since the recall base of the test collection indicates the topical relevance level of the documents for respective query, we exploit the recall base as the source of relevance judgments for RF simulation. The simulation is conducted in the following way: Initial query results are scanned; each document on the ranked result list is checked against the recall base of the respective topic to obtain the relevance judgments.

When the recall base is applied as it is, i.e., relevance judgments are obtained directly from the recall base, this represents the deterministic case. We applied the deterministic case in Study I, where we accepted the topical relevance judgments about documents as given by the recall base. Thereafter, classifiers are trained to separate the relevant and non-relevant documents. However, does accepting relevance judgments of the recall base as gold standard reflect the real behavior of information system users? First of all, the recall bases which are utilized in experiments are generated by IR experts or by various persons who may have a dedicated task to develop test collections (Voorhees & Harman, 2000). On the contrary, we assume a typical information searcher. Actually, even assuming that users will agree with the expert's opinion about document relevance would be naïve, yet we also accept this assumption in our experiments; otherwise it would not have been possible to conduct the experiments and compare them with past ones.

Despite the fact that users usually judge correctly the relevance of a document, they can make mistakes between adjoining relevance levels (Vakkari & Sormunen, 2004). A probability distribution of making mistakes during feedback can be constructed. For example, according to such a distribution, a user may assess a document that is assessed by experts as fairly relevant, e.g., with 10% probability as non-relevant, 20% probability as marginal, 50% probability as fair (correct) and 20% as highly relevant.

We defined fallibility scenarios, which are employed in Study II for RF to construct models for fallible users. Our fallibility scenarios range from 100% correct judgments to completely random judgments. Although the probability distribution values are arranged with more probability mass to the correct relevance level and its neighboring levels, only some of them obey the normal distribution according to Shapiro-Wilk test<sup>2</sup>, because the rest are limited by the far edge of relevance levels.

In addition to the systematically varied distributions, we designed one based on empirical observations by Vakkari and Sormunen (Vakkari & Sormunen 2004; Sormunen, 2002). They discovered that searchers are capable of recognizing highly relevant documents quite correctly but tend to err when dealing with the marginal and non-relevant documents. Therefore, in this empirically grounded distribution,

---

<sup>2</sup> “R: A language and environment for statistical computing” retrieved May 8, 2014, from <http://www.r-project.org/>

presented in Table 1 and labeled “0.50-0.80”, the probabilities are more peaked – 80% correct – for fairly and highly relevant documents, and flatter – only 50% correct – for non-relevant and marginally relevant documents.

Table 1. Fallibility probability distributions

Fallibility		Human Judgment Probabilities			
Scenario		n	m	f	h
<b>0.50-0.80</b>	n	<b>0.5</b>	0.4	0.1	0.0
	m	0.4	<b>0.5</b>	0.1	0.0
	f	0.0	0.1	<b>0.8</b>	0.1
	h	0.0	0.0	0.2	<b>0.8</b>

In more detail, the sample fallibility scenario labeled “0.50-0.80” in Table 1 defines respective assessment probabilities across the relevance levels. The row and column headers represent the relevance levels of non-relevant (n), marginal (m), fair (f), and highly relevant (h) documents. The row labels n to h represent the true document relevance labels as given in the test collection. The column labels n to h represent the (simulated) fallible human relevance judgments. The human judgment probabilities are given in the respective cells. The gold standard for RF would always deliver correct judgments on document relevance during the feedback process – that is, probability 1.0 along the diagonal in the table and other probabilities equal to zero. In the empirically grounded scenario of Table 1, the judgment probabilities approach the correct judgment for fair and highly relevant documents but deviate more from correct judgments for non-relevant and marginally relevant documents.

Further, we defined three more fallibility scenarios (see Study II, not shown in Table 1) which are motivated by the exploration of the effects of progressively increasing fallibility. Accordingly, we decreased the probability values of the sets systematically from fairly consistent judgments towards entirely random ones. Obviously, simulated relevance judgments are affected by random decisions made according to probabilities. As a consequence, RF effectiveness likely fluctuates in

relation to random decisions. Therefore, we run each RF experiment multiple times in accordance with the Monte Carlo simulation approach (Altiok & Melamed, 2007, pp. 11-22) and report the average effectiveness.

### 3.2.2 Query Modification Strategies

Interactive search sessions can be characterized simply by querying and scanning iterations. While reformulating a query, various users prefer diverse strategies (Fidel, 1985). We examined some of these strategies, with which users achieve their goals under the constraint of the overall available session time. The procedure to reformulate a query can be defined in terms of query modification (QM) strategy.

First, we naïvely assume that a list of individual words  $\{w_1, w_2, w_3, w_4, w_5\}$  is available for each particular topic, even though real life searchers learn from snippets seen and documents inspected (Vakkari, 2000). This can be seen as part of the simplification of the model for the simulation purposes. Nevertheless, to increase the realism of the experiments, we let two groups of test persons, students and researchers, suggest the search keywords for a set of topics. QM strategies determine how elements from this list are selected to form either an initial query or subsequent queries. In other words, the QM strategy defines how to form a sequence of queries (Keskustalo et al., 2009).

In Study III, we generated all possible query sequences with the permutations of the available individual search keys, and scrutinized their effectiveness. One should also note that the number of possible QM strategies becomes very large even with five search keywords and a limited number of queries per session. Thus, we limited the number of queries to three, which reflects real life search behavior (Jansen et al., 2000; Kamvar & Baluja, 2007; Yi et al., 2008). Besides, the required computation time for large number of queries per session would not be viable because of our limited computing resources. Still, it would be quite interesting to look at the lengthy sessions and their characteristics.

On the other hand, we paid special attention to five idealized versions of QM strategies, which have been employed by real users and reported in the literature (Keskustalo et al., 2009). We call them prototypical QM strategies. They are based on term-level changes. Consequently, we can only observe a limited number of

queries per session by prototypical QM strategies, which also reflect real life behavior. According to a study by Jansen et al. (2000), the typical length of a search session is about three queries, and users employ 2.21 keywords per query on average. The prototypical strategies<sup>3</sup> are:

**S1:** an initial one-word query ( $w_1$ ) is followed by queries which replace the word with the next one in the available list.

$Q_1: w_1 \rightarrow Q_2: w_2 \rightarrow Q_3: w_3 \rightarrow Q_4: w_4 \rightarrow Q_5: w_5$

**S2:** an initial two-word query ( $w_1 w_2$ ) is followed by queries which replace the second word in the initial query with the next one from the available list.

$Q_1: w_1 w_2 \rightarrow Q_2: w_1 w_3 \rightarrow Q_3: w_1 w_4 \rightarrow Q_4: w_1 w_5$

**S3:** an initial three-word query ( $w_1 w_2 w_3$ ) is followed by queries which replace the third word in the initial query with the next one from the available list.

$Q_1: w_1 w_2 w_3 \rightarrow Q_2: w_1 w_2 w_4 \rightarrow Q_3: w_1 w_2 w_5$

**S4:** an initial one-word query ( $w_1$ ) is followed by queries which extend the previous query with the next search word from the available list.

$Q_1: w_1 \rightarrow Q_2: w_1 w_2 \rightarrow Q_3: w_1 w_2 w_3 \rightarrow Q_4: w_1 w_2 w_3 w_4 \rightarrow \dots$

**S5:** an initial two-word query ( $w_1 w_2$ ) is followed by queries which extend the previous query with the next search word from the available list.

$Q_1: w_1 w_2 \rightarrow Q_2: w_1 w_2 w_3 \rightarrow Q_3: w_1 w_2 w_3 w_4 \rightarrow \dots$

### 3.2.3 Scanning and Assessment Behavior

After posing a query to a search system, a user may scan one or more documents before formulating the next query or ending the search session. If the search process is simply split into scanning and querying, after a single query  $Q_i$  a sequence of one or more document snippets may be scanned ( $s_{ij}$ : scanning the  $j^{\text{th}}$  document for query  $Q_i$ ):

$Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow \dots$

---

<sup>3</sup> In Study IV we omitted strategy S4, therefore strategy S4 in the paper represents strategy S5 in this summary.

A user can scan a varying number of document snippets after posing any particular query to a search system during a search session. This results in a vast number of possible querying-scanning sessions, e.g.:

$$\begin{aligned}
 & Q_1 \rightarrow s_{11} \rightarrow Q_2 \rightarrow s_{21} \rightarrow Q_3 \rightarrow s_{31} \rightarrow \dots \text{ or} \\
 & Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow Q_2 \rightarrow s_{21} \rightarrow \dots \text{ or} \\
 & Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow Q_2 \rightarrow s_{21} \rightarrow s_{22} \rightarrow Q_3 \rightarrow s_{31} \rightarrow \dots \text{ etc.}
 \end{aligned}$$

A typical search session continues until the user's information need is at least partially satisfied and/or all the time allocated for the session is consumed or the user has no further ideas for a new query or is unwilling to produce new queries. The scanning lengths may fluctuate for many reasons depending on the user's belief in the success of the current query (Carterette et al., 2011) and the user's accumulated total gain during the whole session. Therefore, we analyzed the properties of optimal and suboptimal interactive search sessions for given time constraints. For the analysis, all possible sessions, which were formed by all combinations of scanning lengths using a sequence of available queries for each topic, were produced. In simulations we confined our focus on the first result page, assuming ten documents on a page, because only a few top documents are often inspected by users in real life (Jansen et al., 2000; Ruthven, 2008). Therefore, the top ten document snippets were taken into account when the scanning length combinations were built. In Study III, we utilized all five QM strategies (see Section 3.2.2) and varying scanning lengths in order to find the best-performing combination.

Furthermore, we can elaborate on the search process with more subtasks such as scanning the snippet, clicking the link, reading the linked document, and judging its relevance. Every subtask may be associated with a cost, e.g., in terms of time. Consider the handling of a single query  $Q_i$  again.

$$Q_1 \rightarrow s_{11} \rightarrow c_{11} \rightarrow r_{11} \rightarrow j_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow c_{13} \rightarrow r_{13} \rightarrow j_{13} \rightarrow \dots$$

Here  $s_{ij}$  stands for scanning  $j^{\text{th}}$  snippet for  $i^{\text{th}}$  query,  $c_{ij}$  clicking on the snippet,  $r_{ij}$  reading the linked document, and  $j_{ij}$  judging its relevance. If the searcher clicks on every snippet on the search result page, and reads and judges every document which is clicked, the cost of this session manifests as the sum of action costs (e.g., assuming the first and third documents are relevant):

$$qc_1 + sc_{11} + cc_{11} + rc_{11} + jc_{11} + sc_{12} + sc_{13} + cc_{13} + rc_{13} + jc_{13} + \dots$$

Assuming a particular number  $n$  of keywords is either available or searchers are ready to produce, it is possible to generate  $2^n - 1$  word combinations for distinct queries. Further, when a set of queries is available for each topic, the searcher can scan a varying number of document snippets after any query. Altogether, this leads to a great many possible querying-scanning-reading-judging sessions.

However, document snippets are not always informative and searchers may overlook them (Ruthven, 2008; Turpin et al., 2009). This may cause the searcher to skip some of the snippets and documents, which should be read and assessed otherwise. Furthermore, the relevance judgments of the searchers may be different from experts' opinions or topic relevance assessors. In order to simulate this behavior we selected the probabilities given in Table 1. Table 2 shows the clicking and assessment probabilities by the relevance degree of the documents. For instance, the simulated searcher will click the snippet of a non-relevant document (of relevance degree 0) with the probability of 27%. Top-ranked but still non-relevant documents may mislead the searcher to click the link because of the apparent snippet relevance (Ruthven, 2008). Further, a searcher may judge a non-relevant document with the probability of 20% as relevant. The probabilities increase toward highly relevant documents which are judged as relevant with the probability of 97% because searchers are able to readily recognize highly relevant documents (Vakkari & Sormunen, 2004).

Table 2. Action probabilities by document relevance degree

Feature of Behavior	Relevance Degree			
	0	1	2	3
Clicking Probability	0.27	0.27	0.34	0.61
Judgment as Relevant Prob.	0.20	0.88	0.95	0.97

We model two types of session behavior in respect of interaction with the result list for Studies III and IV, namely deterministic and stochastic behavior cases. In the deterministic behavior case, we assume that a searcher always decides to execute the

subtask and there is no probability to skip the subtask. For example, a searcher will always scan all ten documents in the result list confined by the scan length constraint and available time for searching, will click all relevant document snippets in the scanned result list, as well as will judge them correctly (see Figure 3).

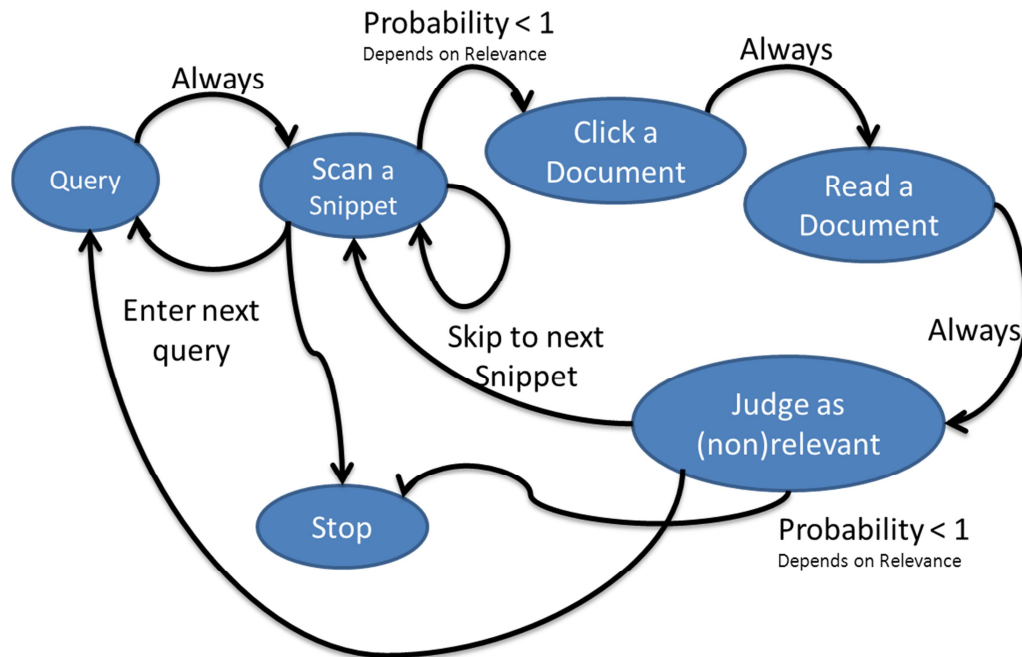


Figure 3. The simulation automaton depicts search session with subtasks

However, in the stochastic case, we assume more realistically that a searcher may err and sometimes may make the wrong decisions with some probabilities, as in real life. The simulation of stochastic behavior is established around scanning and assessment probabilities given in Table 2. For example, after posing a query to a search system, the searcher scans the result list and clicks some of those document links according to the selected probability values, as well as reads and judges them with some other probability (see Figure 3). We simulated sessions representing stochastic behavior with the help of the Monte Carlo approach (Altiok & Melamed, 2007, pp. 11-22). This means that we ran the experiments multiple times with random decisions according to the given probabilities, then averaged the outcomes of the experiments in order to achieve a more stable and robust analysis of the underlying phenomena.



### 3.2.4 Modeling Frustration

In real life, a searcher can scan one or more documents or links before frustration strikes, as a result of futile results, and then stops scanning further and reformulates the next query. Therefore, several factors affect the decision to continue scanning the results of a current query in a session: the search gain goal, the gain accumulated by the preceding queries in a session, the gain accumulated by a current query before the current scanning position, and the length of the current scan from rank one.

However, earlier studies (e.g., Carterette et. al. 2011), which model result scanning processes, assume single query sessions. This means that users are assumed to pose only one query and the proposed models predict at which rank users stop scanning the search results and quit the search process. In other words, they focus on the utility gained in a single query session. On the other hand, other researchers (e.g., Kanoulas et. al. 2011) modeled the multi-query sessions but did not consider either the search gain goals or the gain accumulated by previous queries in a search session. In summary, none of prior studies into scanning length modeling take multiple query sessions, varying gain goals, and simultaneously user's efforts as well as frustration into account.

Consequently, in the present thesis we modeled the user's frustration and contributed a novel formula for the scanning process regarding multiple query sessions in Study IV.

## 3.3 Session Simulation

### 3.3.1 Search Environments

First of all, Studies I-IV each had their own simulation environment, which emphasizes the particular aspects of that study and is restricted by the respective study objectives. In Study I we simulated relevance feedback on top results, and learned classifiers to classify the remaining results in order to improve IR effectiveness. Here, we resorted to the recall base of the collection (Voorhees & Harman, 2000) as the source employed to assess the topical relevance of documents

on the first result page. Of course, the evaluation procedure makes use of the recall base too. Further, in Study II we again utilized a recall base, but we also introduced some noise into the user's interpretation of the recall base to represent the real life user's struggle concerning document and snippet relevance judgment. In Study III we created a simulation environment in which we examine the effects of various QM strategies in two interface scenarios. Below we describe this environment more precisely. Finally, in Study IV we expand our simulation environment to conduct experiments to study the behavioral factors in the search process. In the last two studies, we produced all the possible variations of some independent variables, such as scanning the search results under several constraints to obtain statements about the effects of the independent variables on the dependent one such as effectiveness, while certain variables like search environment were held constant. These variables will be discussed in more detail later.

For the session simulation in Study III, we first formally generate all possible sessions under constraints. We represented sessions as sequences of actions with costs, because the core of this study was about time aspects of different subtasks. For example, the tuple  $\langle (a_1, c_1), (a_2, c_2), \dots, (a_n, c_n) \rangle$  is a session of  $n$  actions and each pair  $(a_i, c_i)$  in the session represents an action  $a_i$  and its cost  $c_i$  in seconds. The elementary action types and costs are:

- initial query, represented as ('iq', ic)
- query reformulation ('q', qc)
- document snippet scan ('s', sc)
- next page request ('n', nc)

The constraints are:

- MaxSLen, maximum session length in terms of elementary actions
- MaxSCost, maximum session cost (seconds)
- a session always begins with an initial query
- all queries (initial and reformulation) are followed by at least one snippet scan

Consequently, the shortest possible session can be formed by an initial action IA =  $\langle (iq, ic), (s, sc) \rangle$ , consisting of an initial query followed by the scan of one snippet (with costs). To generate longer sessions, the possible subsequent elementary actions with costs are defined as the set:

$$NA = \{ \langle (q, qc), (s, sc) \rangle, \langle (s, sc) \rangle, \langle (n, nc), (s, sc) \rangle \}$$

Note here that the next actions are tuples of one or two elementary actions; a scan may appear individually, while a reformulation/next page requires a scan to follow.

Sessions are generated by concatenating the actions subsequent to the initial action. This operation generalizes over a set of session tuples  $S_i$ , denoted as:

$$\times_{i=1\dots n} S_i = \langle \langle \dots \langle \langle S_1, S_2 \rangle, S_3 \rangle, \dots \rangle, S_n \rangle.$$

The cost of a session  $S$  can be determined, informally, by the sum of its action costs. More formally, we derive this cost by the function  $s\text{-cost}$  as follows:

$$s\text{-cost}(S) = \sum_{(a,c) \in S} c$$

Notably, the definition of the set membership operator was enhanced from sets to tuple components in an obvious way. For example, the cost of the session  $S1 = \langle ('iq', ic), ('s', sc), ('q', qc), ('s', sc) \rangle$  is  $s\text{-cost}(S1) = ic + sc + qc + sc$ .

To generate sessions, we first generate all sessions up to the maximum number of actions  $\text{MaxSLen}$ . This session set is  $\text{MLS}$ :

$$\text{MLS} = \bigcup_{i=1\dots\text{MaxSLen}} \{ \langle \text{IA}, \times_{j=1\dots i} ac_j \rangle \mid ac_j \in \text{NA} \}$$

We then select the subset of sessions fulfilling the time constraint  $\text{MaxSCost}$  and the scan length constraint. Note that this approach does not define the query contents or modifications in sessions (see Section 3.2.2). However, it keeps them within constraints and guarantees that the last action is a document snippet scan.

For session simulations in Study IV, we first generated all possible sessions under constraints as in the Study III. However, this time we refined the behavior aspect during search process further. Namely, we introduced more action types such as “click a link”, “read a document” and “judge document relevance”. Moreover, we integrated all query strategies into sessions, while paying special attention to prototypical QM strategies (Keskustalo et al., 2009). Thereby, all possible queries were constructed by a combination of keywords suggested by the real users for each topic (Keskustalo et al., 2009). Furthermore, sessions were executed many times in the stochastic case to encounter the randomness of the decisions concerning the actions. Again, we represented sessions as sequences of actions with costs because of time constraints. Nevertheless, the main focus of these simulations was the effectiveness of prototypical QM strategies employed by real users compared to all possible QM strategies under constraints. In the deterministic case, we ran more than a million sessions for each experiment, while for the stochastic case we ran more than a billion sessions for each experiment.

The next subsection explains and justifies the cost aspects used in the studies.

### 3.3.2 Cost Aspects

The effort to formulate a query, to scan the result list, to read documents and to judge the documents can be characterized by cost, or rather in terms of time expended (Azzopardi, 2011). Average subtask costs, which are utilized in our experiments, are given in Table 3. Thereby the scenario, which depends on the search environment such as the access device, determines the absolute cost. Empirical studies show that it takes significantly longer to enter queries using a small smartphone keypad than it does using an ordinary keyboard (Kamvar & Baluja, 2007). Two scenarios, i.e., a desktop PC scenario (PC) and a smartphone scenario (SP), are designed to study the effects of subtask costs under overall session cost constraints. These scenarios have different subtask costs, because the properties of the devices partially determine the user's effort to accomplish the subtasks (Kamvar et al., 2009; Smucker, 2009).

Table 3. Average subtask costs used in Study IV (in seconds)

<b>Session subtask</b>	<b>Costs</b>
Entering a query word	3.0
Scanning one document snippet	4.5
Reading and evaluating one document	30.0
Entering the relevance judgment	1.0

Obviously, forming queries under different QM strategies S1 – S5 (see Section 3.2.2) also leads to very different costs. All queries in strategies S1, S2, and S3 have a fixed query length in sessions (one, two or three words, correspondingly) while in strategies S4 and S5 the queries grow longer. In real life the typing speed is affected by, e.g., the experience and knowledge of the person, the size of the keyboard, the layout of the keyboard (e.g., nine-key multi-tap vs. QWERTY keyboard) (Kamvar & Baluja, 2007; Karat et al., 1999) and whether a predictive text feed is available and used. We derived the cost values in PC and SP scenarios regarding the initial query cost and the subsequent query cost from literature (see Table 1 in Study III) (Kamvar & Baluja, 2007).

The query costs in S1 – S5 in the desktop PC case are based on the typing costs of 3.0 seconds per word. The corresponding smartphone case costs are based on the article by Kamvar and Baluja (2007). The authors performed a large-scale log analysis of mobile phone usage and observed that an average smartphone query length was 2.56 words and the average query-entry time was 39.8 seconds (average typing cost of 15.5 seconds per word). We assumed in our simulations that the cost of adding one word to a query (that is, extending one-word query and extending two-word query strategies, S4 and S5) or replacing one word at the end of the previous query (that is, one, two, or three-word query strategies) is a constant depending on the scenario.

The information processing of humans can be approximately described by perceptual, motor, and cognitive systems (Card et al., 1983, pp. 23-100). The document snippet scanning costs in real life are affected by the costs accumulated by the above-mentioned systems. However, in Studies III and IV we assumed that the document snippet scanning cost is constant in both scenarios and across the QM strategies. Moreover, we excluded the eventual thinking time in producing query words, which can be interpreted as a modeling artifact because of simplification of the real world.

## 4. Evaluation of Interactive Information Retrieval

Without measuring the performance of systems and the outcomes of experiments, real progress in scientific pursuit cannot be achieved. Therefore, this chapter briefly introduces the evaluation methods (Catarci & Kimani, 2013), which are applied in our experiments. Further, the statistical methods, which are utilized to show the statistical significance of the experiment results, are shortly described.

### 4.1 Rank-Based Evaluation

Information retrieval has a long tradition in measuring IR system performance with respect to ranking of the search results. From the system-oriented view of IR, the most important aspect is the rank of documents which are returned as a query result. Precision, the proportion of retrieved relevant documents to the retrieved documents, and recall, the proportion of the retrieved relevant documents to all known relevant documents, are the very first measures which are applicable to the results of IR experiments. For instance, mean average precision (MAP) is a widely-used measure to compare systems with each other. MAP is calculated by averaging the precision values at the ranks of retrieved relevant documents of a query result, and thereafter the mean value of all query averages. Another recently popular measurement, cumulated gain (CG) (Järvelin & Kekäläinen, 2000), is based on the gain that every document contributes. The gain of a retrieved document is usually associated with the relevance level of the document. Further, ranks of the retrieved documents are taken into account by discounting the gain factor according to the position of the documents in the result list. This results in discounted cumulated gain (DCG) (Järvelin & Kekäläinen, 2002). However, in order to accomplish the comparability between different systems or experiments, the cumulated gains should be normalized; indeed the normalized discounted cumulated gain (NDCG)

introduced by Järvelin and Kekäläinen. (2002) has become one of the most widely-applied evaluation measures in IR domain. DCG is normalized with the help of ideal discounted cumulated gain, which can be calculated by summing the discounted gains of known relevant documents for each query in descending order up to the rank where NDCG value is required. Across the queries of an IR experiment, NDCG values are averaged by taking an arithmetic mean in order to obtain a final NDCG value.

## 4.2 Time-Based Evaluation

IR evaluation is traditionally considered a rank-based process. However, when the time a user expends during a search session is taken into account, traditional metrics are inadequate for evaluating the search results. Because traditional metrics are time agnostic, a user's effort in terms of time is entirely omitted in the evaluation process. However, there have also been research efforts which introduce the time dimension into the evaluation process. For example, Dunlop (1997) suggested "time-to-view" graphs, which incorporate user interface and system as well as the temporal issues into the same framework in order to evaluate search engine effectiveness. According to Dunlop, "time-to-view" graphs offer a single presentation, which enables researchers to compare the interface and effectiveness changes.

Another research effort to introduce the time factors into the traditional Cranfield setting was conducted by Smucker (2009). He tried to improve the traditional evaluation with the use of the GOMS model (Card et al., 1983, pp. 139-192), which stands for goals, operators, methods, and selections. Further, he proposes an IR user model, which incorporates the sequence of actions performed during the searching process, such as typing, clicking, evaluating a snippet summary, and waiting for the results to load. All these actions can be associated with times and probabilities, with which users perform the actions. For instance, whether a user will click on a relevant document surrogate is defined by a given probability. He studied simulations to show the impact of changes in the information retrieval interface on user performance, which was determined by the number of relevant documents read within a given time frame. Moreover, for IR evaluation he suggested a time-biased

gain metric, which captures some aspects of user behavior by regarding the search process (Smucker & Clarke, 2012a; Smucker & Clarke, 2012b). The suggested metric is calibrated through a user study for stochastic simulations. Furthermore, in a subsequent article (Smucker & Clarke, 2012c), the authors simulated different types of users by modeling user variance in time-biased gain in order to estimate the expected number of relevant documents that a user will collect while examining a single ranked result list. Still, their experiments were limited to single query sessions.

Azzopardi (2011) approached interactive IR as an economical problem and examined the trade-off between querying and browsing while holding search utility constant, computed in terms of normalized CG, at a certain level. He employed a user cost function in order to determine the search strategy, which keeps the minimum cost at the constant utility level to a user. The suggested user cost function takes the cost of querying and browsing into account, and is proportional to the number of queries issued. The time expended for querying and browsing is utilized to define the relative cost. Azzopardi (2011) claims that the user cost function estimates the relative cognitive effort of querying and browsing and his approach offers a reasonably fair comparison between strategies.

In our experiments, we take the user's effort as a variable represented by time into account. Consequently, we propose a new time-based evaluation approach. Nevertheless, we utilized the cumulated gain (CG) over time to compare the effectiveness of sessions as well as search strategies because of some very interesting peculiarities with traditional metrics, such as MAP and NDCG (see Chapter 6).

### 4.3 Statistical Methods

Parametric and non-parametric statistical methods are applied in order to identify statistically significant differences between the proposed and the state-of-the-art techniques in IR experiments. Thereby, parametric methods assume some statistical distributions to judge the differences between the examined algorithms, which are tested for significance in preset level of confidence. On the contrary, the non-parametric methods do not depend on such statistical distributions; they rather apply



the rank-based calculations for statistical tests. The following two statistical methods, namely the t-test and the Friedman test, are used in the present thesis to assess the significance of differences between the algorithms in our experiments. Therefore, the t-test and the Friedman test (Hill & Lewicki, 2007, pp. 15-40; Conover, 1999, pp. 367-372) will be briefly described below.

The t-Test, a parametric statistical test, analyzes two data samples and estimates whether the data samples are drawn from the same distribution. Student's t distribution is the underlying distribution for the t-test. In other words, the t-test examines the equality of the means of the two normally distributed samples with unknown variances. If the sample size is large enough, the normality assumption can be relaxed to some extent. Another requirement of the t-test is that the variances of the two data sets should not be too different.

However, when the data samples do not follow the normal distribution, the non-parametric alternatives are more proper than parametric ones. As a non-parametric alternative, the Friedman rank test is selected for the current study for the cases where the data do not follow normal distribution. The Friedman test executes two-way analyses of variance by ranks. Besides, the Friedman test can handle more than two data samples, which occurs in our studies. In brief, the Friedman test first checks whether a significant difference between data samples exists, then calculates pairwise comparison between the data samples, which are produced by the diverse methods which are under evaluation. In order to accomplish the test, the Friedman test determines whether the data samples originate from the same population or populations with the same median, by determining the probability of divergence of rank totals of the samples from the rank values obtained by chance. The Friedman test is explained in more detail in Conover (1999, pp. 367-372).

## 5. Summary of Contributed Studies

In this chapter we present the summaries of the four contributed studies. We briefly explicate motivation, problems, approach, and data for each respective study. Then we present the research questions in a succinct form. Finally, we describe the research results of each study.

Studies I and II handle RF simulations, while Studies III and IV simulate session behaviors without RF. While Studies I and II handle single query sessions, Studies III and IV utilize multiple query sessions.

### 5.1 Study I: Effectiveness of Search Result Classification based on Relevance Feedback

Relevance feedback has been one of the research areas of system-oriented IR for a long time. It has been studied by utilizing either test persons or simple simulations. RF has been conducted through query reformulations with the help of PRF and/or intellectual RF (Ruthven & Lalmas, 2003). In Study I we performed RF in a novel way through the classification of search results after users' initial intellectual feedback. We simulated users' initial intellectual RF for our experiments in a comprehensive collection, namely the TREC 1-2-3-7-8 ad-hoc collections with 250 topics (Voorhees & Harman, 2000). We tried several classifiers, which are explained in Section 2.4.1. We also studied the effects of diverse term space reduction techniques for the classification process. Experimental results were evaluated by user-oriented metrics, P@20, P@30, NDCG@20, and NDCG@30. The following research questions (RQ) were set for Study I:

RQ 1: Given RF on top results of pseudo-RF (PRF) query results, is it possible to learn effective classifiers for the following results? What is the effectiveness of various classification methods?

RQ 2: How does classification effectiveness in RQ 1 depend on term space reduction and classification methods?

RQ 3: When should RF and classification be employed regarding the availability of relevant results in the initial Top 10?

In Study I we propose a novel approach to applying RF. Our approach trains classifiers with the help of simulated user-RF on top of PRF results instead of reformulating the initial user query by expanding with keywords extracted from RF documents. These classifiers are then applied to identifying relevant documents among the subsequent search engine result documents, which have not yet been presented to the user as a result list.

For the first RQ, our results indicate that the proposed classification approach can be applied effectively on top of PRF results. In both cases of title only (T queries) and ‘title-and-description’ queries (T+D queries), the proposed classification approach improves both the initial query results and PRF results, while the improvements over PRF results are smaller than the ones over initial results. This suggests that even though state-of-the-art search engines have so much evidence from long initial queries, the classification approach can still improve the results by learning through top document RF. All in all, the effectiveness of both the short and the long queries can be improved with classification approach. Furthermore, all tested classification methods provide statistically significantly better results over PRF and initial query results.

For the second RQ, further results indicate that term space reduction is no more effective than using the full feature set in T+D queries but that it provides a marginal boost in the shorter T queries. Although the best results for short queries are achieved either by a classification method other than SVM with term space reduction or SVM (Joachims, 1999) with full feature set; the differences between classification methods were minor and statistically not significant. The best methods for long queries were KNN and SVM with a full feature set, while they had only an insignificant advantage over the other classification methods without reduction. However, one should also note that SVM with all features performs quite well, because term space reduction is an integral part of this method. Reduction in SVM

is applied by selecting the support vectors; in this vein, term space is implicitly reduced.

For the third RQ, we found that the classification approach should be applied when there is at least one relevant and non-relevant document in the initial result list. Regarding the searcher behavior, if the result list contains only relevant documents, searcher's information need is probably satisfied on the first page. On the other hand, learning a classifier with only positive or only negative documents complicates building classification models for document space. Moreover, our analysis points out the high correlation of P@10 with P@11-20/30 which means that if first result page has many relevant documents, the subsequent pages will have also many relevant documents, and vice versa, if first result page has no relevant documents, the subsequent result pages will have hardly any relevant documents. Consequently, high correlation between first and subsequent result pages supports the finding on where the classification effort should be focused.

Our findings are based on user simulation. We modeled searcher interaction during RF and assumed feedback on the Top 10 PRF search results. Realistically, we simulated that users browse the first page. However, the assumption of RF for all documents in the first page may be questionable regarding the observation in the IR literature on searcher behavior (Ruthven, 2008).

Finally, our findings indicate that this novel approach of applying RF is significantly more effective than PRF with short and long queries. This paper inspired us towards more elaborate models of user interaction in IR. Namely, we applied the ideas about user fallibility in RF in the next paper.

## 5.2 Study II: Simulating Simple and Fallible Relevance Feedback

In the previous study, relevance feedback was performed under laboratory conditions using test collections and a simulated deterministic searcher. In order to improve the realism of the experiments we designed a unique experimental setup in Study II. First of all, instead of title-and-description derived queries, we introduced realistically short queries that were suggested by real persons (Keskustalo et al., 2009). Second, we simulated human fallibility by providing RF, i.e., partially

incorrect judgments about the documents in the feedback process (see Section 3.2.1). Third, we performed a user simulation with several evaluation scenarios. Finally, we employed graded relevance assessments in the evaluation of retrieval results.

The research questions were:

RQ 1: How effective are PRF and RF when employed on the results of short initial queries and shallow browsing?

RQ 2: Does RF effectiveness seriously deteriorate when RF is of progressively lower quality?

RQ 3: How does RF effectiveness in RQ 2 depend on evaluation by liberal and fair vs. strict relevance criteria?

In order to study real world problems in a laboratory environment, we established a simulation environment, in which a simplified model of the real world is utilized to conduct the experiments. This motivates our present study in which we model user interaction features during RF and vary them systematically.

At first, the relevant features of real world searching were studied in order to fulfill the requirements for more realistic simulations. Day-to-day observations corroborate that interaction in real life IR is indispensable. Besides, individual users interact with information retrieval systems differently. However, a typical real life searcher interaction can be characterized as being simple and error prone, or more specifically, searchers try to achieve the best results with minimum effort, in other words with short queries as well as shallow browsing (for example at most the top 10 documents/snippets checked, rather than the top 1000) (Jansen & Spink, 2006; Jansen et al., 2000; Sakai, 2006). Because providing RF requires extra effort from searchers, they may be reluctant to give it (Ruthven & Lalmas, 2003). If they are ready to provide RF in order to achieve better results, they may make errors when judging the relevance of the feedback documents (Vakkari & Sormunen, 2004).

In our simulations for Study II, we employed (1) very short initial queries, namely one, two and three-word queries; (2) shallow browsing (assuming that at most the top 20 documents per query were inspected); and (3) we also defined the fallibility of the searcher during the providing of the RF. Fallibility was modeled according to several scenarios, assuming that searchers may err during the selection of feedback documents. These scenarios range from assuming perfect user

judgments to completely random judgments. In addition, we define a scenario (see Section 3.2.1) based on empirical findings on the level of fallibility when the user attempts to recognize relevant documents belonging to various relevance levels (Foley & Smeaton, 2009; Vakkari & Sormunen, 2004). A total of five different fallibility scenarios were analyzed. All experiments were run multiple times in line with the Monte Carlo approach (Altiok & Melamed, 2007, pp. 11-22) with random decisions which obey the defined fallibility probabilities, and the results of all runs were averaged to infer reliable statements about the subject matter.

The evaluation of the experiments was based on user-oriented measures, P@10/P@20 and a traditional system-oriented measure, MAP. In retrospect, it would have been interesting to employ cumulated gain-based metrics and to compare the results accordingly. Since in real life users differ in their preferences considering satisfaction levels, we applied three distinct relevance levels. In other words, some users prefer finding even marginally relevant documents, while others want to obtain only highly relevant documents because their expertise in topics varies. Moreover, we decided to exclude the seen documents from RF results, which means we applied full freezing (Keskustalo et al., 2008), because users would not gain any benefits from seeing the same documents in the improved result list after expending effort to inspect them, regardless of their relevance level, in the first result set.

Regarding the first RQ, our results suggest that both PRF and direct user-RF applied by using query-biased summaries<sup>4</sup> are promising methods when very short initial queries are used. For the second RQ, as we expected that although increasing error level in providing RF progressively decreases the performance compared to perfect RF, it is still slightly better than the best-performing PRF. Surprisingly, RF with the empirical level of fallibility yields results that are close to perfect RF results. Considering the third RQ, assuming empirical fallibility and using user-oriented measures such as P@10 and P@20, RF performance systematically exceeds the performance of all short-query types (one, two and three-word queries) at a liberal level (i.e., even marginal documents are accepted as relevant). However, RF does not improve the performance of all short queries against PRF, when strict

---

<sup>4</sup> Query-biased summary process is depicted in Study II in Figure 1.

evaluation is required (i.e., only highly relevant documents are accepted as relevant). This may be part of the reason why RF does not prevail in real life.

Our findings suggest that completely random feedback is no different from pseudo-relevance feedback and is not effective in short initial queries. However, RF with empirically observed fallibility is as effective as correct RF and is able to improve the performance of short initial queries.

Next, we turned to focus on modeling the user characteristics during interaction with a search system. We also take user effort during interaction into account. We extended our experiments session dimension by undertaking multiple queries. In other words, we simulated direct query reformulation. Obviously, this strategy means that we do not study the RF process any more in the following studies.

### 5.3 Study III: Time Drives Interaction: Simulating Sessions in Diverse Searching Environments

In real life, users often conduct search activities by posing multiple queries during a search session (Jansen et al., 2009; White & Drucker, 2007), whereby searching consists of various cognitive, perceptual and motor subtasks (Smucker, 2009). During interaction with a search interface, users apply diverse strategies which affect their effort (cost), experience and session effectiveness. In Study III we suggest a pragmatic evaluation approach based on scenarios with explicit subtask costs. Furthermore, the effectiveness of diverse interactive strategies, namely query modification and scanning strategies, in two search environments, namely in desktop PC and smart phone search environments, was studied comprehensively. We simulated 20 million sessions in each environment to cover all possible interactive search scenarios that were possible within the study design. This in turn enabled us to analyze the effectiveness of the session strategies (see Section 3.2) and the properties of the best and worst performing ones in each environment.

We set the following three empirical and one methodological research questions (RQ):

RQ 1: How effective are the five QM strategies (S1 to S5, see Section 3.2.2) in terms of cumulated gain when we compare the Desktop PC and the Smart Phone (SP) scenarios under overall time constraint?

RQ 2: What are the characteristics of the best and the worst QM sessions?

RQ 3: How stable are the observed trends when the overall time constraint changes? Can we recommend QM strategies based on the PC and SP scenarios assuming a specific time constraint?

RQ 4: What is the proper evaluation methodology when time is part of the evaluation criteria?

In this study, we simulated various search scenarios on two different devices, a desktop PC and a smartphone, regarding diverse search subtask costs under an overall time constraint. Furthermore, the characteristics of the best and worst search sessions were explored. Because real life users have limited time to acquire the necessary information about their task and they use different devices for information access in different situations, our study has unquestionable user relevance, and consequently offers potential pragmatic value to the industry. Measuring the effectiveness of search systems from a user's point of view may reflect the user's interest more accurately, and thus increase the validity of the results achieved.

The first RQ was about the effectiveness of different QM strategies under time constraints. In the desktop PC scenario, when time is tight users cannot pose all possible queries, or utilize their entire search vocabulary. Instead, users may perform exhaustive scanning for a few queries posed. Short queries (strategy S1) perform worst in terms of session effectiveness, which is measured by cumulated gain metric. On the other hand, two or three-word queries clearly outperform the short queries. The same improvements in results can also be observed in strategies S4 and S5, when there is enough time to advance beyond the first query. When more time is available for searching, the initially weaker strategies catch up because users can scan more results, and the ranking of weaker strategies is not that critical. In the smartphone (SP) scenario, users have no time for long queries in a stringent time frame; therefore they must employ shorter queries and scan the weaker quality rankings. The effective strategies require a high query input cost; consequently they may not be applied at all. Again, the more time users have at their disposal, the smaller the gap between the effectiveness of best and worst sessions.

Regarding the second RQ, in both PC and SP scenarios and under stringent time constraints, the best sessions involved less queries and longer scans than the worst sessions. However, when more time is available, the differences between session



characteristics in the PC scenario disappear while in the SP scenario they remain. For the best strategies in each scenario, both the number of queries and the average scan lengths increase as time allowance grows. Respectively, in the worst sessions for the PC case, the number of queries stays the same but the scan lengths grow as more time for search is allocated. Because the worst sessions in the PC case consume all possible queries even under the shortest time frame and the number of queries is limited, the scan lengths grow but not the number of queries. However, in the case of SP for the worst sessions, the number of queries increases and the scan length remains low as time grows. Because input costs are higher than in the PC case, investing the effort for the costly query input defines the worst behavior.

Ultimately, if there is enough time for searching, posing two or more word queries followed by a longer scan seems to provide reasonable effectiveness no matter what the search strategy among S2 to S5 is.

The third RQ is about the stability of observations. When limited time is available for searching, there is a trade-off between two action types, namely posing queries and scanning the search results cost-consciously. Thereby, the overall cost levels related to the stringency of the time frame and the relations between cost types play a major role in selecting the action type. Search interfaces and devices on which searching takes place certainly affect these variables (Kamvar & Baluja, 2007). To sum up, expensive input costs cause lengthy scanning of the search results, whereas cheap input costs help to pose better or rather longer queries. Among the QM strategies S2-S5, there is no significant difference when enough time for the search process is allocated.

Regarding the methodological RQ about the proper evaluation of search sessions under time constraints, we can state that the typical IR evaluation metrics must be applied with great care because they may be insufficient or even misleading, because traditional rank-based IR metrics do not take the user's experience, observed costs and session gains into account. When search costs and time expended during a search session, are taken into consideration and metrics utilize normalization, i.e., scaling the value of measurement to a predefined range such as [0, 1], traditional metrics such as MAP and NDCG deliver deceptive results. Moreover, we pointed out the inappropriate use of all normalized rank-based measures.

## 5.4 Study IV: Modeling Behavioral Factors in Interactive Information Retrieval

In this study, we carry forward our simulation efforts with more fine-grained subtasks and more elaborate behavioral factors. As real life information access is session-based (Jansen et al., 2009), and every session consists of one or more query iterations, sessions are bound by several subtasks like query formulation<sup>5</sup>, result scanning, document link clicking, document reading and judgment, and stopping the session. As a result, the effects of behavioral factors associated with these subtasks are inevitable. These factors include search goals and cost constraints, query formulation strategies, scanning and stopping strategies, and relevance assessment behavior, among others. The purpose of Study IV is to assess the effects of these behavioral factors on retrieval effectiveness. Our research questions include:

RQ 1: How effective is ideal human behavior, i.e., persistent scanning and ideal assessments, employing prototypical query formulation strategies, compared to deterministic baselines under various CG goals and time constraints?

RQ 2: How effective is fallible human behavior, i.e., probabilistic scanning and fallible assessments, employing prototypical query formulation strategies, compared to stochastic baselines under various CG goals and time constraints?

RQ 3: How much does fallible behavior lose in session effectiveness compared to ideal human behavior?

RQ 4: When examining the best possible query formulation strategies, is there a winning query formulation strategy which delivers the best gain across topics?

RQ 5: Methodologically, how does one simulate a behavioral model based on comprehensive session subtasks, fallible human behavior and various query formulation strategies?

In this study, we simulated both ideal human search behavior and the more realistic fallible human search behavior in an environment based on a test collection with graded relevance assessments. Our session models allowed us to simulate multiple query sessions and several interactive subtasks. During simulation experiments, the interface properties, the test collection and the search engine were kept constant and fixed probability distributions for snippet and document relevance

---

<sup>5</sup> Query formulation (QF in Study IV) and Query modification (QM in Study III) are interchangeably used.

assessment and for snippet scanning behaviors were utilized. Then, the following behavior factors, the use of QF strategies, cost constraints and gain goals, were varied systematically. We compared the empirically grounded prototypical QF strategies to three baselines: one long query, which comprises all available query words, with a long scanning of search results, the best possible three query session, and randomly selected QF strategy with three queries.

The first RQ was about the effectiveness of ideal human behavior employing empirically grounded QF strategies in comparison to deterministic baselines under various CG goals and time constraints. Amongst others, we found that some of empirically grounded QF strategies, second word variations (S2) and third word variations (S3) with ideal behavior are the most effective under several time constraints. They are clearly more effective than the expected effectiveness of random query sessions with ideal behavior under open time constraints in binary and non-binary scoring schemes (assigning more weights to more relevant documents), but also perform poorly compared to “one long query” sessions.

Regarding the second RQ, instead of ideal human behavior we simulated fallible human behavior with probability distributions, which were motivated by the literature (Turpin et al., 2009; Vakkari & Sormunen, 2004). Because of the random decisions based on probability distributions, simulation experiments were repeated one thousand times in order to obtain stable statements about the underlying phenomena. This approach is obviously an example of the Monte Carlo simulation method (see section 3.2.1). Again, the third word variation strategy (S3) with fallible behavior was the most effective under time constraints and gain goals. This strategy exceeds the expected effectiveness of random query sessions with fallible behavior under open time constraints for both scoring schemes, but as in the ideal case it is inferior to “one long query” sessions.

The third RQ was about the difference in effectiveness between ideal and fallible human behavior. It is no surprise that fallibility in relevance assessment and scanning decisions affected the effectiveness of sessions negatively but less with regard to highly relevant documents, because of fewer errors in their assessments. This corresponds to human selective capability and effectiveness. All in all, the effectiveness of fallible behavior is 28% to 44% of the ideal behavior.

The fourth RQ was about identifying a winning query formulation strategy across all topics. Unfortunately, there was no optimal query formulation strategy among the almost 28 000 inspected for more than one topic. Furthermore, we analyzed the query formulation strategies across the topics which perform reasonably closely, that is, within 10% of the effectiveness of the optimal strategy that is obviously distinct for every topic. Study IV shows that good session effectiveness requires a topic-focused interaction. Therefore, topics play a major role in explaining IR effectiveness. If query words are available at the beginning of a search session, a long query with persistent scanning achieves quite competitive results. However, real life searchers learn keywords from snippets seen and documents found (Ruthven, 2008). Therefore, many query words are not necessarily available at the beginning of a search session.

Regarding the fifth RQ, we employed a comprehensive multiple query session model including several subtasks, behavioral factors, goals, and constraints. We employed multiple query sessions, whereas prior simulations of interactive IR have concentrated on single query sessions (Smucker, 2009). Further, we experimented with multiple gain goals and time constraints while other studies have had limited goal and time constraints in experiment settings (Azzopardi, 2011). Moreover, we defined a snippet scanning model, which takes not only the current session gain but also the session goal and frustration explicitly into account, whereas Carterette et. al. (2011), for example, only focus on the utility gained by a single query. While we performed an exhaustive search for the best-performing strategies among all possible query formulation strategies, which can be produced under the three query and 5-keyword constraints, we also paid special attention to query formulation strategies observed in real life and analyzed their effectiveness compared to the best-performing ones. These constraints were selected, both because they reflect the typical real life interactive IR sessions and because we were limited by our computing capacity. Having more queries per session and more keywords per query would increase the required computing power exponentially (by around several orders of magnitude).

## 5.5 Summary of Findings

The main research questions and main findings for respective Studies I-IV are summarized in Table 4.

Table 4. Summary of main research questions and main findings

Study	Main Research Questions	Main Findings
<b>Study I</b>	Given RF on the first result page of PRF, is it possible to learn effective classifiers for the subsequent results?	The proposed classification approach can be applied effectively on top of PRF results. Term space reduction is not necessary.
<b>Study II</b>	How does RF affect IR performance when short initial queries are employed and fallible feedback is provided?	Increased error level of providing RF decreases the performance compared to perfect RF. RF with a realistic level of fallibility is as effective as perfect RF and is able to improve the performance of short initial queries.
<b>Study III</b>	How do various interface devices and diverse query formulation strategies affect IR sessions under overall time constraints? What is the proper evaluation methodology when time is taken into account?	If there is enough time for searching, posing two-word or longer queries followed by a longer scan seems to provide reasonable effectiveness no matter what the search strategy is. Typical rank-based IR metrics such as MAP or NDCG should be applied with great care. These metrics evaluate rankings but not user effort or experience.
<b>Study IV</b>	What kind and how effective are the optimal sessions under varying goals and constraints provided that human stochastic behavior is regarded?	Empirically grounded query formulation strategies, second word variations and third word variations are the most effective under several time constraints but also perform poorly compared to “one long query” sessions. Fallible behavior affects IR effectiveness negatively but less when regarding highly relevant documents because of fewer errors in their assessments.

## 6. Discussion and Conclusions

With “Never stop questioning”<sup>6</sup> as our motto, we started to question the relevance feedback and user behavior-related issues in interactive information retrieval. First, we focused on the development of novel approaches for applying relevance feedback. Namely, we utilized various standard classification and term space reduction methods in order to classify retrieved documents according to simulated user relevance feedback. As a result, correct relevance feedback was taken for granted. However, in real search environments users may very well err when they make relevance decisions based on result lists. The fallible behavior of searchers has been observed in empirical studies (Turpin et al., 2009; Vakkari & Sormunen, 2004). Nevertheless, until now experiments have been conducted with the assumption of perfect relevance assessments. To address this, we introduced fallibility in relevance feedback by defining diverse fallibility levels according to which users supply relevance feedback information to a system. Thereafter, we concentrated our research efforts on user behavior aspects, such as search behavior on different devices, which lend themselves to situational requirements, under time and search goal constraints. In this vein we brought “time” into the evaluation, which unveils some very intricate problematic points in IIR evaluation. Thus, we discovered that highly popular rank-based evaluation metrics, such as MAP and NDCG, are inappropriate for the comparison of systems when a searcher’s time expenditure is taken into account. When the search time expended by users is part of the evaluation process, normalized rank-based metrics may provide misleading evaluation results. Therefore, non-normalized metrics should be employed. Finally, we elaborated our simulation experiments by defining fine-grained user behavior variables. For example, search strategies, search goals and cost constraints, scanning and assessment behavior, and relevance scoring were incorporated into the design of simulation experiments. We applied both deterministic and stochastic approaches

---

<sup>6</sup> Albert Einstein’s quote

during the simulation of searcher behavior and contrasted both approaches to narrow the gap between traditional Cranfield-type experiments and real life search behavior. Moreover, user behavior is always assumed correctly in Cranfield-type experiments. However, in our experiments, we adulterated the simulated user behavior with some probabilities which were set according to prior empirical studies (Turpin et al., 2009; Vakkari & Sormunen, 2004) to reflect the real user's interaction with a search system.

Table 5. Summary of themes and variables of the contributed and Cranfield-style studies. Variables are encoded as: fixed variables are lowercase, independent (varied) variables are uppercase and dependent variables are bold and uppercase.

Study	Theme	Subtheme	Signatures of Variables
Cranfield-style	Various	Various	V M q - - - - E
Study I	RF	Classification	V M q t s a c j E
Study II	RF	Fallibility	V m Q t s A c J E
Study III	Sessions	QF/Time/Interfaces	V m Q T S a c j E
Study IV	Sessions	QF/Frustration/Patterns	V m Q T S A C J E

In order to give an overview of the studies, the main and sub-themes together with the variables of Studies I to IV are summarized in Table 5. The relevant variables of each study are encoded in the table as follows: variables which are fixed during experiment execution are given in lowercase, independent variables which are varied, are denoted in uppercase, and dependent variables which are examined, are bolded. In Table 5, *v* stands for vocabulary, and consequently represents information need modeling, *m* stands for the retrieval method applied, *q* for querying, *t* for time consumed, *s* for scanning the search results, *a* for assessing the relevant snippets or documents, *c* for clicking the document links, *j* for judging the document relevance, and *E* for effectiveness of retrieval in the IR experiments. Accordingly, in order to analyze the new approach to RF, we fixed all interaction variables except the method (*M*) in Study I. In Study II we varied the user's judgment of document relevance, and explored its effects on performance. Further, in Study III not only the scanning patterns but also time is varied and effectiveness is analyzed. Finally, in Study IV we varied time and scanning patterns as well as assessing, clicking and judging behavior in order to examine the effectiveness. In comparison, Cranfield-style experiments do not take the variables into account at

all. These are depicted with a dash in Table 5. In Studies I to IV we progressively introduced and varied such variables, and furthermore we shed light on the effects of these variables.

In Study I, we applied standard classification algorithms to select the relevant documents from the result list, more precisely from the second page onwards, after relevance feedback was supplied on the first page of the search engine results. Thereby, we simulated the relevance feedback on the first page of results of the pseudo-relevance feedback run of topical queries. Thus, classifiers were able to learn relevant and non-relevant document classes, which are used to decide on the relevance of further documents on the result list. Moreover, we explored numerous term space reduction techniques (Sebastiani, 2002) for improving both effectiveness and efficiency of the classification process. Comprehensive experiments on TREC ad-hoc collections (Voorhees & Harman, 2000) indicated that this approach of applying RF with the help of classification methods is significantly more effective than PRF with title queries as well as title-and-description queries. However, the difference between classification approaches and the combination of classification methods with term space reduction methods entail no significant improvement from the statistical view point. In other words, any of the state-of-the-art classification methods can improve the PRF results of short queries. However, in order to learn a classifier, the first page of results should have at least one relevant and one non-relevant document, otherwise the necessity of inspecting the following pages may disappear because of the correlation of the first and subsequent result pages. When the first page has a lot of relevant documents, the user's information need may already be satisfied (Sakai, 2006), and vice versa, when the first page has no relevant document, the following pages are not likely to have enough relevant documents worth being classified or inspected either.

One could try to infer RF automatically e.g., from users' interaction with search interfaces (Ruthven, 2008), and thereafter apply the classification approach to amend the result lists. Overall, the parallelism between our approach and Learning to Rank algorithms (Li, 2011) should be emphasized once more while regarding how differently both approaches are applied to improving search results.

In Study II, we simulated the RF further but this time we suggested a novel approach to RF evaluation, i.e., we introduced and systematically studied the effects



of the searchers' fallibility in supplying RF. This can occur in real life search situations, because the snippets delivered by a search engine could mislead searchers into deciding incorrectly about the relevance of a document (Turpin et al., 2009). Further, the multi-grade relevance assessments (Sormunen, 2002) were employed in this study to improve the realism of simulations by regarding the expertise levels of searchers. Moreover, the experiments were carried out with the short initial queries generated by real searchers (Keskustalo et al., 2009) instead of the longer title-and-description queries, which are quite popular in Cranfield-style IR experiments. Our findings indicated that the results of very short initial queries can be improved by applying query-biased summaries (Bates, 1989; Tombros & Sanderson, 1998; Turpin et al., 2009) for both PRF and direct user-supplied RF. Furthermore, the experimental results showed that very fallible feedback is no different from pseudo-relevance feedback (PRF) and not effective in short initial queries. However, RF with empirical fallibility is practically as effective as correct RF and is able to improve the performance of short initial queries. RF systematically improves the performance of all short-query types when evaluation is liberal; but does not improve against PRF when evaluation is strict. Short initial queries obviously do not provide enough good documents for strict evaluation. It is not surprising that in real life users prefer to revise their queries instead.

In Study III our attention was drawn to session simulations away from the RF simulations. Real life search experience is characterized by time constraints, multiple short queries and a myriad of different search interfaces on different devices. Because busy life situations require prompt answers to ad-hoc emerging questions, ubiquitous computing lends itself to such kinds of needs very well. However, different devices demand different interaction styles; consequently time plays a major role during query input phase. First, we explored the space for all possible multi-query sessions within the limited framework for the best querying and scanning behavior under diverse time and goal constraints. Interestingly, there was no single winning query modification pattern which performed best across all the topic queries. Both the number of queries and the average scan lengths increase in the best strategies for both PC and SP scenarios analyzed when time allocation grows, whereas the number of queries does not change but scan length grows in the worst sessions for personal computer scenario, because the worst sessions consume

all the available queries as soon as possible. On the other hand, for the smartphone scenario, while the number of queries grows, the scan lengths remain low when more time is available for searching. Because the input costs are high, investing the effort in query typing is apparently the worst behavior.

The typing speed supported by a device or an interface determines a user's input effort. Where the input effort is low, for example in the PC case, better or rather longer queries are favorable. However, if the input effort is high, such as in the SP case, users are not able to type longer queries under time pressure. Instead they must perform lengthy scans of weak short-query results in order to achieve their goals. When the user has enough time at their disposal, the way in which searches are executed is not crucial, because there will be enough time to identify the relevant documents.

Finally we discovered the peculiarities of time-based evaluation with standard rank-based evaluation metrics. Typical IR evaluation metrics (Demartini & Mizzaro, 2006; Su, 1992) are based on the quality of ranking alone. When time is taken into account, normalized metrics may deliver misleading results, and therefore should be used with care. Furthermore, in session-based evaluation they must also be applied with great care because they may be insufficient or even misleading. They may be partially insensitive to the user's experience, e.g., because of recurrent documents, and observed costs and benefits. This is particularly critical when a user's costs (time expenditure) are taken into account and the metric employs normalization, i.e., scaling the measurements to a predefined range such as  $[0, 1]$ . For example, the popular NDCG metric (Järvelin & Kekäläinen, 2002) and its non-discounted counterpart NCG should be avoided in any comparisons between searching environments, and between strategies within a given searching environment when user effort is taken into account. This is because the ideal gain vector used for normalization is read to vastly different lengths between strategies or environments. For example, consider Figure 4, which plots NCG over time for QM strategy S2 in two scenarios, PC and SP, from Study III.

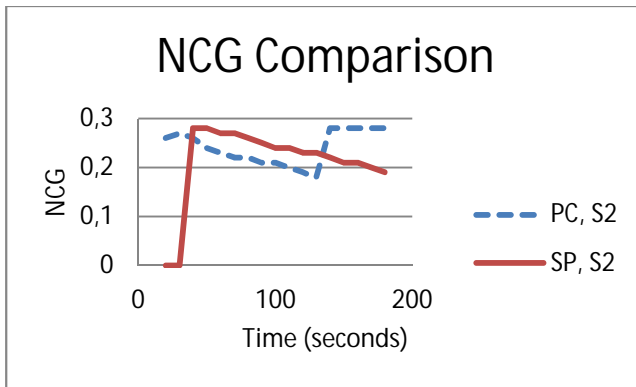


Figure 4. NCG vs. time comparison of PC and SP scenarios for QM strategy S2

Due to normalization (division by the ideal cumulated gain vector), the smartphone (SP) scenario seems to exhibit better performance in the time frame from 40 to 135 seconds. This is due to (a) ranking being somewhat effective, and (b) the number of documents seen in each session: in the PC case the user sees 15 to 35 documents, but in the SP case only 5 to 20 documents in the indicated time frame. Figure 5 plots CG with the corresponding data and makes the difference clear. Similar pitfalls also plague the most classic metric, MAP (see Chapter 6 in Study III).

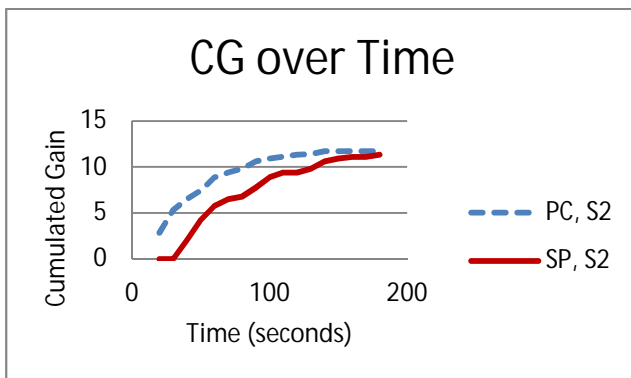


Figure 5. CG over time for QM strategy S2 in PC and SP scenarios

Even within a non-normalized metric like CG, incorporating time in session-based evaluation has profound effects. Consider figures 6 and 7. The former gives traditional cumulated gain over ranks for QM strategies S1 and S3 averaged over the topics. The latter gives CG over time for the same strategies in the two scenarios.

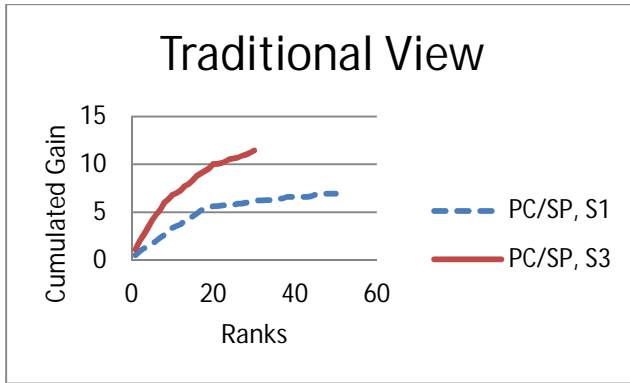


Figure 6. Traditional View, CGs over ranks, scenarios PC and SP for QM strategies S1 (allowing five queries) and S3 (allowing only three queries)

In Figure 6, both the PC and SP scenarios have the same observed effectiveness, because the evaluation focuses on the gain (CG) over the result ranks, regardless of how long it takes to retrieve the documents. The two strategies S1 and S3 differ in effectiveness, S3 providing far better effectiveness than S1. However, when time is taken into account (Fig. 7), the scenarios and strategies differ greatly from each other. Up to 60 seconds, S3 in the SP case is the worst strategy and this is entirely due to the high input cost of the long query. With enough time (180 sec.), S3 in SP catches up with S3 in the PC case. Also, PC and SP do not much differ for S1 due to the relatively low input cost and the weak result quality. Comparing figures 6 and 7, it is easy to see that time profoundly affects both user experience and effectiveness in sessions in different scenarios.

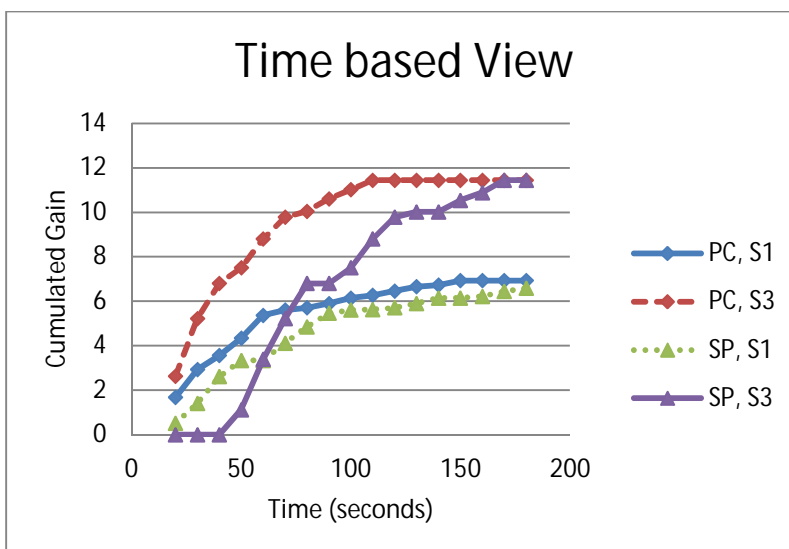


Figure 7. Time-based View, CGs over time, scenarios PC and SP for strategies S1 and S3

In terms of time, we employed cumulated gain in our experiments in Study III. But in order to compare various devices and search interfaces with respect to time-based evaluation, especially in different studies, the normalization issue remains to be addressed and is an interesting study subject.

In Study IV we delved into user behavior issues during query result inspection. We modeled full multi-query sessions with comprehensive subtasks. Thus we extended the previously defined user models (Keskustalo, 2010) with further details about scanning the snippets, deciding on clicking the links, reading the documents, and judging the relevance of documents. Moreover, we conducted experiments based on both deterministic and stochastic behavior. While the deterministic approach dictates certain predictable behavior, the stochastic case requires probabilities about the outcome of every decision. Therefore, we defined probability distributions beginning with the deterministic case and moving towards a totally random one. Then, we compared the prototypical query modification strategies with the best possible strategy. Hence, we can further suggest the prototypical strategies, especially second and third word variations (S2 and S3), for future research activities, because even they do not reach the level of best performer patterns; they still represent a regular pattern and are on par with best performers to some degree. In the ideal case, S3 achieves about three-quarters of the performance of the long query and of the by-topic optimized best session pattern, which means the strategy that is distinctly optimized for each topic. All prototypical QM strategies except the sequence of one-word queries (S1) are close to each other in terms of effectiveness, with both ideal and fallible behavior. Among all possible QM strategies inspected, around 28 000, there was not one strategy that was best across all the topics. Further analysis showed that the next to best-performing strategies, the effectiveness of which is at least 90% of the best strategy for each particular topic, across all the topics achieve good performance in only around one-third of the topics. This advocates the view that users apply topic-specific QM strategies in order to reach the highest possible effectiveness.

Because the results of stochastic experiments depend on the selected probability values, the experiments are repeated in order to get an average value over a wide range of possible values. In conclusion, stochastic behavior was obviously inadequate in comparison with deterministic one, which is not realistic although it is

superior. Probabilistic scanning and fallible relevance assessment limit the performance but cause considerably less damage regarding highly relevant documents. Another analysis showed that single long queries yield better performance levels than multiple query sessions. However, in real life searchers do learn during interaction with search results, frequently modify queries accordingly (Ruthven, 2008) and do not initially have a long query available.

Methodologically we extend the use of traditional test collections to include behavioral factors in a controlled experimental environment in order to study the effects of searcher-related factors in IIR. Furthermore, we simulated the behavioral factors using the Monte Carlo method (Altiok & Melamed, 2007, pp. 11-22) based on behaviors observed in real life studies (Vakkari & Sormunen, 2004). We also integrated the time variable into the evaluation process. Moreover, we simulated user interactive sessions on different interfaces. Besides, we examined all possible query formulation patterns created by a limited vocabulary in order to analyze their effects on IIR effectiveness.

In general, with every article the researchers learn new concepts and gain insight into diverse phenomena. In retrospect, so did we. Thus the recently gained knowledge can now be fed into the research settings of the previous studies as well as blended with new research topics; thereby very intricate research problems may arise, such as searcher behaviors in several stages of interaction with various search systems and environments as well as in the evaluation process.

However, one should bear in mind that the applied methods and findings are not limited to topical search, but they can be exploited across a wider field of information retrieval. When the importance of information retrieval in the current and future knowledge society is considered, the contribution of bringing the searcher's behavioral aspects into the fast-paced IR experimental world can be better appreciated.

With this thesis and its contribution to the IR research community, we hope we are able to instigate new approaches, methods, and metrics for interactive information retrieval.

# References

- Alpert, J. & Hajaj, N. (2008, July 25). Official Blog: We knew the web was big. [Web log post]. Retrieved from <http://googleblog.blogspot.fi/2008/07/we-knew-web-was-big.html>
- Altiok, T., & Melamed, B. (2007). *Simulation modeling and analysis with ARENA*. Burlington, MA, USA: Academic Press.
- Azzopardi, L. (2011). The economics in interactive information retrieval. *In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Beijing, China. 15-24. doi:10.1145/2009916.2009923.
- Azzopardi, L., Järvelin, K., Kamps, J., & Smucker, M. D. (2011). Report on the SIGIR 2010 workshop on the simulation of interaction. *SIGIR Forum*, 44(2), 35-47. doi:10.1145/1924475.1924484
- Banerjee, S., & Pedersen, T. (2003). The design, implementation, and use of the ngram statistics package. *Computational linguistics and intelligent text processing* (pp. 370-381) Springer.
- Bates, M. J. (1989). The design of browsing and berrypicking techniques for the online search interface. *Online Information Review*, 13(5), 407-424.
- Belew, R. K. (2000). *Finding out about: A cognitive perspective on search engine technology and the WWW*. Cambridge University Press.
- Belkin, N. J. (1980). Anomalous states of knowledge. *Canadian Journal of Information Science*, 5, 133-143.
- Belkin, N. J., & Croft, W. B. (1987). Retrieval techniques. *Annual Review of Information Science and Technology*, 22, 109-145.
- Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human computer interaction*. Routledge.
- Carpineto, C., & Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Comput.Surv.*, 44(1), 1-50. doi:10.1145/2071389.2071390
- Carterette, B., Kanoulas, E., & Yilmaz, E. (2011). Simulating simple user behavior for system effectiveness evaluation. *In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, Glasgow, Scotland, UK. 611-620. doi:10.1145/2063576.2063668.
- Catarci, T., & Kimani, S. (2013). Human-computer interaction view on information retrieval evaluation.7757, 48-75. doi:10.1007/978-3-642-36415-0\_3
- Chapelle, O., & Chang, Y. (2011). Yahoo! learning to rank challenge overview. *Journal of Machine Learning Research-Proceedings Track*, 14, 1-24.
- Clarke, C. L. A., Freund, L., Smucker, M. D., & Yilmaz, E. (2013). Report on the SIGIR 2013 workshop on modeling user behavior for information retrieval evaluation (MUBE 2013). *SIGIR Forum*, 47(2), 84-95. doi:10.1145/2568388.2568403
- Cleverdon, C. W., Mills, J., & Keen, M. (1966). *Factors determining the performance of indexing systems*. New York
- Conover, W. J. (1999). *Practical nonparametric statistics*. John Wiley & Sons.
- Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search engines: Information retrieval in practice*. Addison-Wesley Reading.
- Demartini, G., & Mizzaro, S. (2006). A classification of IR effectiveness metrics.3936, 488-491. doi:10.1007/11735106\_48
- Dunlop, M. D. (1997). Time, relevance and interaction modelling for information retrieval. *SIGIR Forum*, 31(SI), 206-213. doi:10.1145/278459.258569

- Efthimiadis, E. (1996). Query expansion. *Annual Review of Information Science and Technology*, 31, 121-187.
- Fidel, R. (1985). Moves in online searching. *Online Information Review*, 9(1), 61-74.
- Foley, C., & Smeaton, A. F. (2009). Synchronous collaborative information retrieval: Techniques and evaluation. In: *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, Toulouse, France. 42-53. doi:10.1007/978-3-642-00958-7\_7.
- Harman, D. (1992). Relevance feedback revisited. In: *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1-10.
- Hill, T., & Lewicki, P. (2007). *STATISTICS: Methods and applications*. Tulsa, OK.: StatSoft.
- Ingwersen, P., & Järvelin, K. (2005). *The turn: Integration of information seeking and retrieval in context (the information retrieval series)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.
- Jansen, B. J., Booth, D. L., & Spink, A. (2009). Patterns of query reformulation during web searching. *Journal of the American Society for Information Science and Technology*, 60(7), 1358-1371. doi:10.1002/asi.21071
- Jansen, B. J., & Spink, A. (2006). How are we searching the world wide web? A comparison of nine search engine transaction logs. *Information Processing & Management*, 42(1), 248-263.
- Jansen, B. J., Spink, A., & Saracevic, T. (2000). Real life, real users, and real needs: A study and analysis of user queries on the web. *Inf.Process.Manage.*, 36(2), 207-227. doi:10.1016/S0306-4573(99)00056-4
- Järvelin, K. (2009). Interactive relevance feedback with graded relevance and sentence extraction: Simulated user experiments. In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, Hong Kong, China. 2053-2056. doi:10.1145/1645953.1646299.
- Järvelin, K., & Kekäläinen, J. (2000). IR evaluation methods for retrieving highly relevant documents. In: *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Athens, Greece. 41-48. doi:10.1145/345508.345545.
- Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Trans.Inf.Syst.*, 20(4), 422-446. doi:10.1145/582415.582418
- Joachims, T. (1999). Making large scale SVM learning practical.
- Joachims, T., & Radlinski, F. (2007). Search engines that learn from implicit feedback. *Computer*, 40(8), 34-40. doi:10.1109/MC.2007.289
- Kamps, J., Geva, S., Peters, C., Sakai, T., Trotman, A., & Voorhees, E. (2009). Report on the SIGIR 2009 workshop on the future of IR evaluation. *SIGIR Forum*, 43(2), 13-23. doi:10.1145/1670564.1670567
- Kamvar, M., & Baluja, S. (2007). Deciphering trends in mobile search. *Computer*, 40(8), 58-62. doi:10.1109/MC.2007.270
- Kamvar, M., Kellar, M., Patel, R., & Xu, Y. (2009). Computers and iphones and mobile phones, oh my!: A logs-based comparison of search users on different devices. In: *Proceedings of the 18th International Conference on World Wide Web*, 801-810.
- Kanoulas, E., Carterette, B., Clough, P. D., & Sanderson, M. (2011). Evaluating multi-query sessions. *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1053-1062.
- Karat, C., Halverson, C., Horn, D., & Karat, J. (1999). Patterns of entry and correction in large vocabulary continuous speech recognition systems. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Pittsburgh, Pennsylvania, USA. 568-575. doi:10.1145/302979.303160.



- Kekäläinen, J., & Järvelin, K. (2002). Evaluating information retrieval systems under the challenges of interaction and multidimensional dynamic relevance. *In: Proceedings of the 4th CoLIS Conference*, 253-270.
- Kelly, D. (2005). Implicit feedback: Using behavior to infer relevance. *New directions in cognitive information retrieval* (pp. 169-186) Springer.
- Kelly, D. (2009). Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, 3(1—2), 1-224.
- Keskustalo, H. (2010). Towards simulating and evaluating user interaction in information retrieval using test collections. Ph.D. thesis. Tampere, Finland: University of Tampere
- Keskustalo, H., Järvelin, K., & Pirkola, A. (2006). The effects of relevance feedback quality and quantity in interactive relevance feedback: A simulation based on user modeling. *In: Proceedings of the 28th European Conference on Advances in Information Retrieval*, London, UK. 191-204. doi:10.1007/11735106\_18.
- Keskustalo, H., Järvelin, K., & Pirkola, A. (2008). Evaluating the effectiveness of relevance feedback based on a user simulation model: Effects of a user scenario on cumulated gain value. *Information Retrieval*, 11(3), 209-228. doi:10.1007/s10791-007-9043-7
- Keskustalo, H., Järvelin, K., Pirkola, A., Sharma, T., & Lykke, M. (2009). Test collection-based IR evaluation needs extension toward sessions --- A case of extremely short queries. *In: Proceedings of the 5th Asia Information Retrieval Symposium on Information Retrieval Technology*, Sapporo, Japan. 63-74. doi:10.1007/978-3-642-04769-5\_6.
- Lam-Adesina, A. M., & Jones, G. J. F. (2001). Applying summarization techniques for term selection in relevance feedback. *In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana, USA. 1-9. doi:10.1145/383952.383953.
- Lavrenko, V., & Croft, W. B. (2001). Relevance based language models. *In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana, USA. 120-127. doi:10.1145/383952.383972
- Li, H. (2011). Learning to rank for information retrieval and natural language processing. *Synthesis Lectures on Human Language Technologies*, 4(1), 1-113.
- Losee, R. M. (1998). *Text retrieval and filtering: Analytic models of performance*. Kluwer Academic Publishers.
- Maria, A. (1997). Introduction to modeling and simulation. *In: Proceedings of the 29th Conference on Winter Simulation*, Atlanta, Georgia, USA. 7-13. doi:10.1145/268437.268440
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press Cambridge.
- Marchionini, G. (1993). Information seeking in full-text end-user-oriented search systems: The roles of domain and search expertise. *Library and Information Science Research*, 15(1), 35-69.
- Marchionini, G. (1995). *Information seeking in electronic environments*. New York, NY, USA: Cambridge University Press.
- Oard, D. W., & Kim, J. (2001). Modeling information content using observable behavior.
- Qin, T., Liu, T., Xu, J., & Li, H. (2010). LETOR: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13(4), 346-374.
- Ricardo, B. Y. (1999). *Modern information retrieval*. Pearson Education.
- Ruthven, I. (2008). Interactive information retrieval. *Annual Review of Information Science and Technology*, 42(1), 43-91. doi:10.1002/aris.144.v42:1
- Ruthven, I., & Kelly, D. L. (2011). *Interactive information seeking, behaviour and retrieval*. Facet Publishing.
- Ruthven, I., & Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18(2), 95-145. doi:10.1017/S0269888903000638

- Ruthven, I., Lalmas, M., & Van Rijsbergen, K. (2003). Incorporating user search behavior into relevance feedback. *Journal of the American Society for Information Science and Technology*, 54(6), 529-549. doi:10.1002/asi.10240
- Sakai, T. (2006). Give me just one highly relevant document: P-measure. In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, Washington, USA. 695-696. doi:10.1145/1148170.1148322.
- Salton, G. (1970). Evaluation problems in interactive information retrieval. *Information Storage and Retrieval*, 6(1), 29-44.
- Saracevic, T. (1996). Relevance reconsidered. In: *Proceedings of the Second Conference on Conceptions of Library and Information Science (CoLIS 2)*, 201-218.
- Saracevic, T. (1997). The stratified model of information retrieval interaction: Extension and applications. In: *Proceedings of the American Society for Information Sciences*, 34 313-327.
- Saracevic, T. (2006). Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II. *Advances in Librarianship*, 30, 3-71.
- Saracevic, T. (2007). Relevance: A review of the literature and a framework for thinking on the notion in information science. Part III: Behavior and effects of relevance. *Journal of the American Society for Information Science and Technology*, 58(13), 2126-2144.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys*, 34(1), 1-47. doi:10.1145/505282.505283
- Siegel, S., & Castellan, N.J. (1988). Nonparametric statistics for the behavioral sciences. *McGraw-HiU Book Company, New York*.
- Smucker, M. D. (2009). Towards timed predictions of human performance for interactive information retrieval evaluation. In: *Third Workshop on Human-Computer Interaction and Information Retrieval (HCIR'09)*.
- Smucker, M. D., & Clarke, C. L. A. (2012a). Time-based calibration of effectiveness measures. In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Portland, Oregon, USA. 95-104. doi:10.1145/2348283.2348300
- Smucker, M. D., & Clarke, C. L. A. (2012b). Stochastic simulation of time-biased gain. In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, Maui, Hawaii, USA. 2040-2044. doi:10.1145/2396761.2398568.
- Smucker, M. D., & Clarke, C. L. A. (2012c). Modeling user variance in time-biased gain. In: *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*, Cambridge, California. 3 1-10. doi:10.1145/2391224.2391227.
- Sokolowski, J. A., & Banks, C. M. (Eds.). (2011). *Principles of modeling and simulation: a multidisciplinary approach*. John Wiley & Sons
- Sormunen, E. (2002). Liberal relevance criteria of TREC -: Counting on negligible documents? In: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland. 324-330. doi:10.1145/564376.564433.
- Strohman, T., Metzler, D., Turtle, H., & Croft, W. B. (2005). Indri: A language model-based search engine for complex queries. In: *Proceedings of the International Conference on Intelligent Analysis*, 2(6).
- Su, L. T. (1992). Evaluation measures for interactive information retrieval. *Information Processing & Management*, 28(4), 503-516. doi:10.1016/0306-4573(92)90007-M
- Tombros, A., & Sanderson, M. (1998). Advantages of query biased summaries in information retrieval. In: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia. 2-10. doi:10.1145/290941.290947.
- Toms, E. (2013). User-oriented information retrieval.7757, 76-85. doi:10.1007/978-3-642-36415-0\_4
- Turpin, A., Scholer, F., Jarvelin, K., Wu, M., & Culpepper, J. S. (2009). Including summaries in system evaluation. In: *Proceedings of the 32nd International ACM SIGIR*

- Conference on Research and Development in Information Retrieval*, Boston, MA, USA. 508-515. doi:10.1145/1571941.1572029.
- Vakkari, P. (2000). Cognition and changes of search terms and tactics during task performance: A longitudinal case study. *In: RIAO*, 894-907.
- Vakkari, P. (2003). Task-based information searching. *Annual Review of Information Science and Technology*, 37(1), 413-464. doi:10.1002/aris.1440370110
- Vakkari, P., & Hakala, N. (2000). Changes in relevance criteria and problem stages in task performance. *Journal of Documentation*, 56(5), 540-562.
- Vakkari, P., & Sormunen, E. (2004). The influence of relevance levels on the effectiveness of interactive information retrieval. *Journal of the American Society for Information Science and Technology*, 55(11), 963-969. doi:10.1002/asi.20046
- Voorhees, E. M., & Harman, D. K. (2000). The eighth text REtrieval conference (TREC-8), volume 8. national institute of standards and technology, NIST. *NIST Special Publication*, 500-246.
- White, R. W. (2011). Interactive techniques. In I. Ruthven, & D. Kelly (Eds.), *Interactive information seeking, behaviour and retrieval* (pp. 171) Facet Publishing.
- White, R. W., & Drucker, S. M. (2007). Investigating behavioral variability in web search. *In: Proceedings of the 16th International Conference on World Wide Web*, 21-30.
- White, R. W., Ruthven, I., & Jose, J. M. (2002). The use of implicit evidence for relevance feedback in web retrieval. *2291*, 93-109. doi:10.1007/3-540-45886-7\_7
- Yi, J., Maghoul, F., & Pedersen, J. (2008). Deciphering mobile search patterns: A study of yahoo! mobile search queries. *In: Proceedings of the 17th International Conference on World Wide Web*, Beijing, China. 257-266. doi:10.1145/1367497.1367533.
- Zhai, C., & Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. *In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 334-342.

# Appendix

## Term Space Reduction algorithms

In order to utilize various term space reduction methods (see for the formulas: Banerjee & Pedersen, 2003; Sebastiani, 2002; Siegel & Castellan, 1988, pp. 102-166 & 224-254), for each term, a 2 x 2 matrix is created, which incorporates the number of relevant and non-relevant documents in which a respective term occurs or does not occur (see Table A1). With the help of this matrix, a particular measure for the respective term can be calculated according to the formulas given below. Having a list of terms ordered by the magnitude of the calculated measure, the number of possible terms can be pruned to the desired size by neglecting the terms that are less significant.

Table A1. Term matrix for term space reduction

# of documents	Relevant Documents	Non-relevant Documents
Term existence	$n_{11}$	$n_{12}$
Term absence	$n_{21}$	$n_{22}$

In the table,  $n_{11}$  is the number of relevant documents in which the current term occurs;  $n_{12}$  is the number of non-relevant documents in which the current term occurs;  $n_{21}$  is the number of relevant documents in which the current term does not occur; and  $n_{22}$  is the number of non-relevant documents in which the current term does not occur.

**Mutual information gain** (MIG) measures the mutual dependence of two random variables. MIG is the expected value of pointwise mutual information. The

formula for MIG is given below. The expected value ( $E_{ij}$ ) for the pertinent cell position is the ratio of the product of marginal to the total number of frequencies, e.g., the number of documents:

$$E_{ij} = \frac{(\sum_{\text{column}} O) \cdot (\sum_{\text{row}} O)}{\sum O}, \text{ in first cell: } E_{11} = \frac{(n_{11} + n_{21}) \cdot (n_{11} + n_{12})}{(n_{11} + n_{12} + n_{21} + n_{22})}$$

Where  $O$  is the observed value for the respective cell.

$$\text{MIG} = \sum_i \sum_j n_{ij} \cdot \log\left(\frac{n_{ij}}{E_{ij}}\right)$$

**Pearson's chi-squared** test examines the variables according to chi-squared test. The test value will be calculated by summing the normalized squared deviations between observed and theoretically expected frequency.

$$X^2 = \sum_{i=1}^n \frac{(O_i - E_i)^2}{E_i}$$

Where  $X^2$  is the Pearson test statistics,  $O_i$  is observed frequency,  $E_i$  is expected frequency,  $n$  is the number of cells in the tables, i.e., 4.

**Odds ratio** also measures the association of two variables.

$$\text{Odds ratio} = \frac{(n_{11} \cdot n_{22})}{(n_{21} \cdot n_{12})}$$

**Kendall-Tau rank correlation coefficient**, as the name suggests, measures the rank correlation of two variables. It describes the similarity of orderings of variables, and can be calculated thus:

$$\tau = \frac{\# \text{ of concordant pairs} - \# \text{ of discordant pairs}}{0.5 * n \cdot (n - 1)}$$

Where  $n$  is the number of observations. Any pair of observations of two variables e.g.,  $(x_i, y_i)$ ,  $(x_j, y_j)$ , are concordant if both values of the pair are either greater ( $x_i > x_j$  and  $y_i > y_j$ ) or smaller ( $x_i < x_j$  and  $y_i < y_j$ ) than in the pair. Otherwise, the pairs are discordant unless the pair values are the same. The contingency table can be expressed as observation values of two variables in order to build concordant and discordant pairs. Then the Kendall-Tau rank correlation formula can be calculated.

**The Spearman rank correlation coefficient** is a non-parametric measure like the Kendall-Tau rank correlation, which examines the dependence of two variables.

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

Where  $x_i$  and  $y_i$  represent sample vectors, and  $\bar{x}$  and  $\bar{y}$  are sample means. Again, the contingency table can be reformulated as x and y vectors, e.g.,  $x = [n_{11}, n_{12}]$  and  $y = [n_{21}, n_{22}]$ , thereafter the Spearman rank correlation formula can be applied.

**Fisher's exact test** is an exact test to define the association of two variables in a contingency table; in the current thesis these variables are the existence and absence of terms. The probability of obtaining exactly these values as in contingency tables is given by hyper-geometric distribution. Fisher's exact test formula is given as:

$$p = \frac{\binom{n_{11} + n_{12}}{n_{11}} \cdot \binom{n_{21} + n_{22}}{n_{21}}}{\binom{n}{n_{11} + n_{21}}}$$

Where n is the sum of all cell values, and  $\binom{a}{b}$  represents the binomial coefficient, which is calculated as  $\frac{a!}{b!(a-b)!}$  and gives the number of 'b' element subsets from 'a' elements. As a result, the first binomial component of the numerator calculates the combinatorial number of term existence in relevant documents, and the second multiplier calculates the combinatorial number of term absence in non-relevant documents. Multiplication of both these numbers gives the number of all possible combinations, which is divided by the number of selecting all possible combinations of relevant documents in the document collection. Finally, the result of this division, which denotes the relative frequency of the occurrence of an experiments outcome, results in the probability of the term occurrence in relevant documents. Therefore, the probabilities of terms can be compared further, so as to order them accordingly.

# Journal of Information Science

<http://jis.sagepub.com/>

---

## **Effectiveness of search result classification based on relevance feedback**

Feza Baskaya, Heikki Keskustalo and Kalervo Järvelin

*Journal of Information Science* 2013 39: 764 originally published online 23 May 2013

DOI: 10.1177/0165551513488317

The online version of this article can be found at:

<http://jis.sagepub.com/content/39/6/764>

---

Published by:



<http://www.sagepublications.com>

On behalf of:



[Chartered Institute of Library and Information Professionals](#)

**Additional services and information for *Journal of Information Science* can be found at:**

**Email Alerts:** <http://jis.sagepub.com/cgi/alerts>

**Subscriptions:** <http://jis.sagepub.com/subscriptions>

**Reprints:** <http://www.sagepub.com/journalsReprints.nav>


**Permissions:** <http://www.sagepub.com/journalsPermissions.nav>

>> [Version of Record](#) - Nov 12, 2013

[OnlineFirst Version of Record](#) - May 23, 2013

[What is This?](#)

# Effectiveness of search result classification based on relevance feedback

Journal of Information Science  
39(6) 764–772  
© The Author(s) 2013  
Reprints and permissions:  
sagepub.co.uk/journalsPermissions.nav  
DOI: 10.1177/0165551513488317  
jis.sagepub.com  


**Feza Baskaya**

School of Information Sciences, University of Tampere, Finland

**Heikki Keskustalo**

School of Information Sciences, University of Tampere, Finland

**Kalervo Järvelin**

School of Information Sciences, University of Tampere, Finland

## Abstract

Relevance feedback (RF) has been studied under laboratory conditions using test collections and either test persons or simple simulation. These studies have given mixed results. Automatic (or pseudo) RF and intellectual RF, both leading to query reformulation, are the main approaches to explicit RF. In the present study we perform RF with the help of classification of search results. We conduct our experiments in a comprehensive collection, namely various TREC *ad-hoc* collections with 250 topics. We also studied various term space reduction techniques for the classification process. The research questions are: given RF on top results of pseudo RF (PRF) query results, is it possible to learn effective classifiers for the following results? What is the effectiveness of various classification methods? Our findings indicate that this approach of applying RF is significantly more effective than PRF with short (title) queries and long (title and description) queries.

## Keywords

Classification; IR; relevance feedback

## 1. Introduction

When interacting with information retrieval systems, the user's first query formulation usually acts as an entry to the search system and database, and is followed by browsing and query reformulations [1]. Because the selection of good search keys is difficult but crucial for good results, query modification is often necessary. Initial query results can be improved through the user's explicit reformulations, relevance feedback (RF) or pseudo RF where the first initial results are assumed relevant. In the latter two techniques, a new query is constructed on the basis of feedback. Query expansion (QE) typically is the technique for constructing the new query. Efthimiadis [2], Ruthven and Lalmas [3] and Ruthven et al. [4] provide useful reviews of the techniques. In the present paper we propose a novel approach to applying RF where the user's RF on the top of the pseudo RF search results, Top-10 in effect, is used to learn classifiers to classify the subsequent results, that is, Top-11–50.

We simulate a search scenario, where users point out relevant documents on the first result page (Top-10) and the retrieval system trains a classifier with relevant and non-relevant document clusters from this feedback and then classifies the rest of search result list, in effect Top-11–50. In this way we utilize both relevant and non-relevant feedback from the user. Onoda and colleagues [5] have experimented with Support Vector Machines (SVMs) in RF for document

---

## Corresponding author:

Feza Baskaya, School of Information Sciences, University of Tampere, Tampere FIN-33014, Finland.  
Email: Feza.Baskaya@uta.fi



retrieval and shown the potential of the approach. Recently, Chen and colleagues [6] proposed a text classification based method for RF. However, both studies applied a large number of RF iterations in order to show the effectiveness of the suggested approaches. In the present study, we assume just one round of RF and classification. We believe this is more realistic regarding user behaviour. We employ an extended version of text classification based RF. We run our experiments on a large collection (TREC 1-2-3-/7-8 test corpus with 250 topics).

In traditional RF, knowledgeable experienced searchers may benefit more of RF because they recognize relevant vocabulary and are better able to articulate their needs initially [7]. Users also seem more likely to identify highly relevant documents than marginal ones [8]. Baskaya and colleagues [9] showed that slightly incorrect recognition of relevant documents is not detrimental to RF effectiveness, in particular if the searcher identifies the best documents correctly. Earlier findings based on simulation [10, 11] suggest that RF is most effective when little feedback is given as early as possible followed by immediate reformulation, rather than extensively browsing the initial results. Therefore we limit the feedback to the Top-10 in our experiments; this corresponds to what users see at a glance in typical search environments.

However, there are two difficulties in providing feedback: searcher's capability and willingness [3]. Pseudo-relevance feedback (PRF) [3] avoids these challenges by assuming that the first documents of an initial search result are relevant. Long documents and non-relevant documents, however, introduce noise in the PRF process, thus causing query drift. To counteract this, one may use query-biased summaries [10, 12] for the identification of expansion keys. Yet another challenge to PRF is that real users tend to issue very short queries [13, 14] and employ shallow browsing and active query reformulation. As a consequence, the results of PRF tend to be of poor quality. In the present paper we examine the effectiveness of RF with result classification over the PRF results, as well as of PRF, both for short (title) and long (title and description) initial queries.

Our approach, learning classifiers to utilize RF for re-ranking results, differs from other learning to rank algorithms [15]. Hang [15] has given a concise description of learning to rank methods for information retrieval and natural language processing. Even though both approaches, the current study and learning to rank methods, employ machine learning techniques for re-ranking the documents, our approach utilizes the first page of the respective result sets as training data and consequent pages as test data. On the other hand, learning to rank methods try to find a model from some pre-labelled training data with the help of machine learning techniques and exert it on test data for ranking predictions. Xiubo Geng and colleagues [16] studied query-dependent ranking and applied the *K*-nearest neighbour (*KNN*) method for it. They, however, created a ranking model for a given query using the labelled neighbours of the query in the query feature space. In a broad sense, our approach can be seen as a variation of learning to rank for re-ranking retrieved documents. Our method learns a model for every query from RF supplied by a simulated searcher, and applies it on the following results in order to improve the ranking of the documents by discarding the non-relevant documents from the list.

We base our experiments on searcher simulation (like Baskaya et al. [9] and Keskustalo et al. [11]) rather than tests with real users. Simulation has several advantages, including cost-effectiveness and rapid testing without learning effects, as argued in the SIGIR SimInt 2010 Workshop [17]. In addition, the simulation approach does not require a user interface. The informative aspects and realism of searcher simulation can be enhanced by explicitly modelling those characteristics of searchers and RF that pertain to RF effectiveness.

Our evaluations are based on four standard information retrieval (IR) evaluation metrics ( $P@20$ ,  $P@30$ ,  $NDCG@20$  and  $NDCG@30$ ). The main role is given to  $P@20/NDCG@20$  and  $P@30/NDCG@30$  as clearly user-oriented measures – users frequently avoid browsing beyond a couple of results page, that is, 10 links/documents [13]. After giving RF and already browsing up to 10 documents, the  $P@20/NDCG@20$  can be seen as evaluation for quasi-first page and the  $P@30/NDCG@30$  for quasi-second page.

## 2. Study design

### 2.1. Research questions

Our main research question is: given RF on Top-10 results of pseudo RF query results, is it possible to learn effective classifiers for the following results, at ranks 11–50? More specifically:

**RQ1:** How effective is search result classification of result ranks 11–50 given RF on result ranks 1–10? How does this compare with PRF? How does this depend on initial query length ( $T$  = title, and  $T\&D$  = title and description queries)?

**RQ2:** How does classification effectiveness in RQ1 depend on term space reduction and classification methods?

**RQ3:** When should RF and classification be employed regarding the availability of relevant results in the initial Top-10?

## 2.2. Test collection, search engine and query construction

We used the TREC 1-2-3/7-8 *ad-hoc* test collection including 250 topics, topic numbers 51–200 and 351–450, with binary relevance assessments. The topics have, on average, 189 relevant documents in the recall base. The document database contains 741,865 documents indexed under the retrieval system Lemur Indri (<http://www.lemurproject.org/indri.php>). The index was constructed by stemming document words by Porter stemmer.

The research questions do not require any particular interactive method to be employed. We simulate interactive RF that takes place at document level: the simulated users point to relevant documents and the RF system then automatically trains the classifiers. The simulated user examines the entire Top-10 of the initial query result and marks each relevant document; the rest are assumed to be non-relevant. This decision is based on the relevance of each Top-10 result in the recall base of each topic.

The initial title queries are on average 3.1 words and title and description queries 17.8 words long. When constructing the queries, the topic words are stemmed. Queries are constructed as bag-of-words queries.

## 2.3. Classifiers and term space reduction

We studied several standard classification and clustering methods for the classification process [18, 9]:

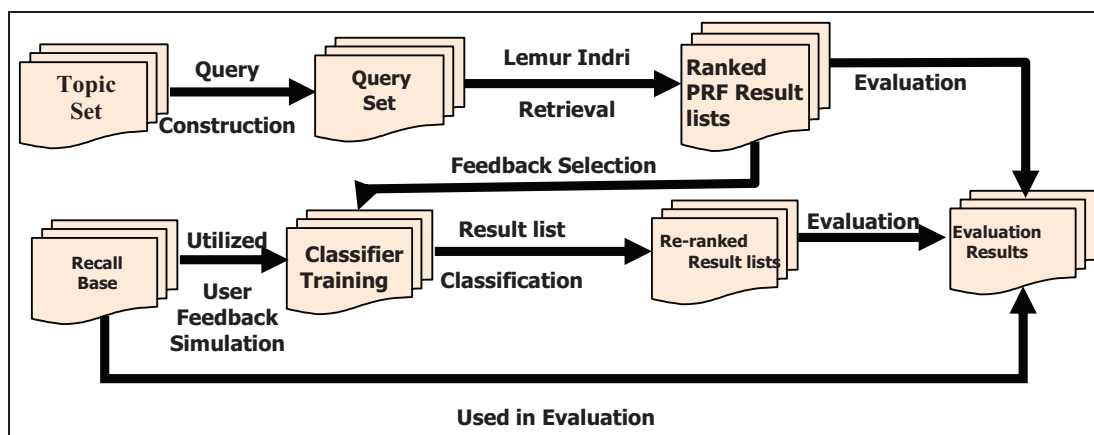
- KNN (*K*-nearest neighbours);
- KMeans;
- naive Bayes;
- SVM (Support Vector Machine).

These are suitable choices because they are widely used and well understood. Therefore one may assess whether the RF with a classification approach is at all useful. All the classification algorithms except SVM are implemented in Python programming language by the researchers. SVMlight [20] is utilized for SVM experiments.

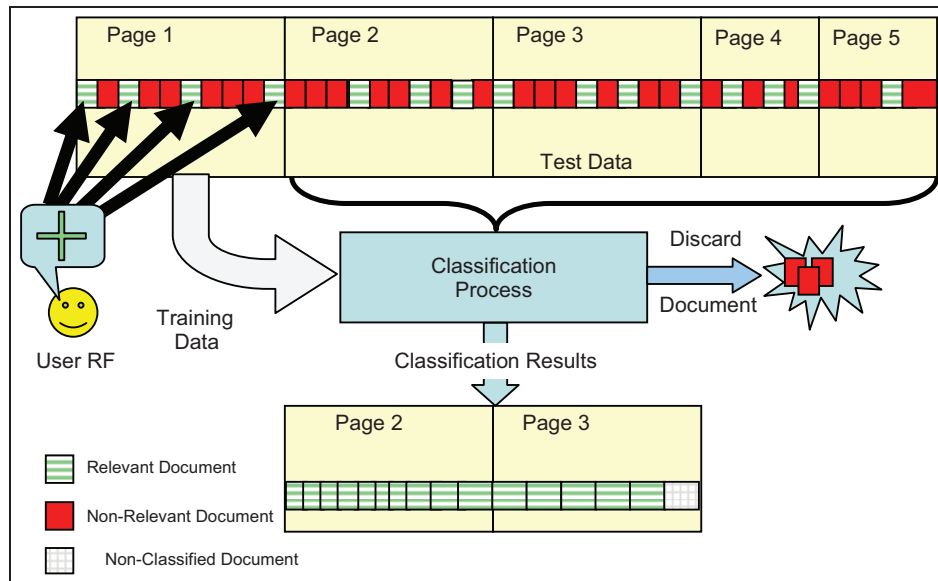
Often in text classification, term space reduction methods may be utilized to improve the efficiency of classification without a loss in effectiveness. We experimented with the following reduction methods: Fisher exact test, Pearson's chi-square test, Kendall–Tau rank correlation coefficient, Spearman rank correlation coefficient, information gain, and odds ratio. These are standard methods [21]. Having observed in initial tests that the other methods delivered comparable results, we focused on Kendall–Tau and information gain as the reduction methods in training the classifiers.

## 2.4. Experimental protocol

Figure 1 illustrates the overall experimental protocol. TREC topics are first turned to initial short and long queries (stemmed) and executed with Lemur Indri, followed by feedback document selection. This is based on the simulated searcher's feedback scenario (in the present experiments browsing first 10 documents and returning the relevant documents as positive RF). The feedback documents for each query are used to learn classifiers, and the rest of the result list



**Figure 1.** Classification-based RF retrieval process.



**Figure 2.** Classification of search results after RF by user.

for that query is classified. No new query is executed, and both the original ranked results as well as PRF results and re-ranked results classified by feedback go to evaluation and comparison.

The detailed process of training classifiers and classification of each document in a result set is depicted in Figure 2. Up to 40 documents from subsequent pages are classified as relevant or non-relevant. The non-relevant ones are discarded and the entire list is moved forwards. Evaluations are executed on the second and third pages. This process takes place for every query.

### 2.5. Evaluation and statistics

We use standard evaluation metrics available in the TREC-eval package and report evaluation results for  $P@20$  and  $P@30$  documents,  $NDCG@20$  and  $NDCG@30$  documents. These are motivated by real life findings – people often are precision-oriented and avoid excessive browsing – great results beyond the first couple of pages are of no importance.

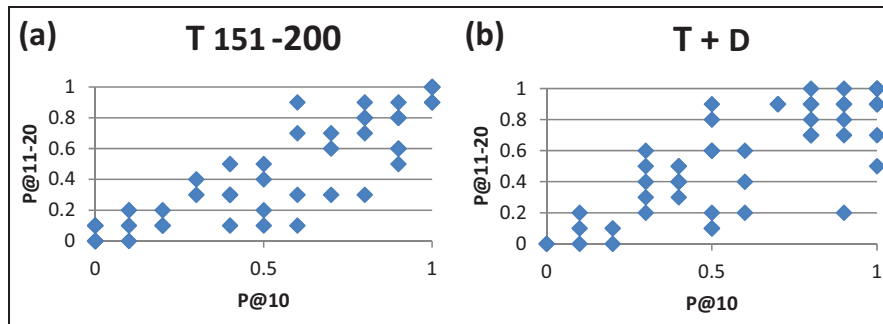
Statistical testing is based on Friedman's test comparing the RF with classification runs and PRF. PRF on the initial query result provides the stronger baseline, and therefore PRF is used as the baseline when (pairwise) statistical significance is evaluated. We ran several PRF experiment with 30–100 extension terms. We report results for PRF with two documents and 100 extension terms because they delivered better results.

## 3. Experimental results

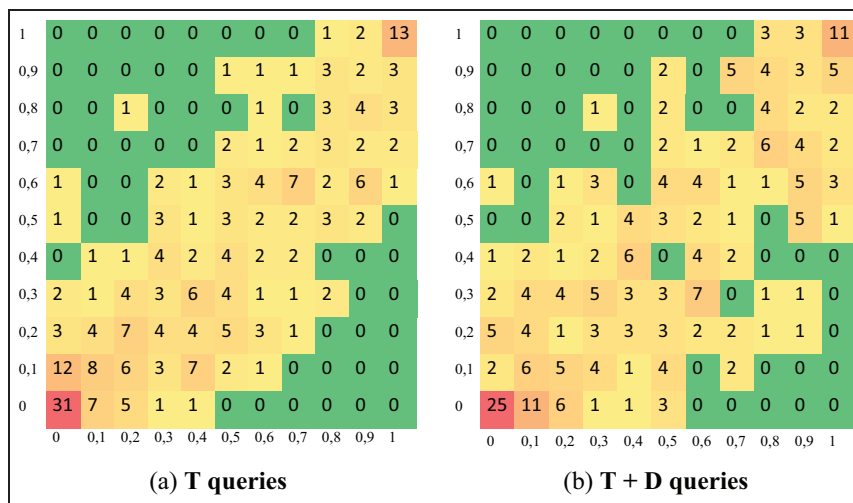
### 3.1. When to apply classification?

Before classifying the search results, we analysed the precision on the first and second page of the search results. This was done in order to learn how many relevant documents the Top-10 initial results provide and how their number is correlated to the number of relevant documents in the rank range 11–20. The former informs about the possibilities to learn classifiers and, from the user viewpoint, about the need to obtain more results. Very few relevant documents makes learning of classifiers challenging, whereas very many increases the probability that the user's need is satisfied in the Top-10 already. The latter informs about the density of relevant documents to be identified in classification when learning the classifiers from Top-10 RF is worthwhile. There need be both relevant and non-relevant documents in the ranks 11–20 for the classification to be worth the effort.

We compared  $P@10$  with  $P@11-20$ . In Figure 3, the horizontal axis represents the  $P@10$  values and the vertical axis  $P@11-20$ . The test collection is TREC, topics 151–200. Not surprisingly, the general trend is that, the more precise the first result page, the more precise the following page.



**Figure 3.** Scatterplots of correlation between P@10 vs P@11–20. (a) Title queries, only for TREC topics 151–200. (b) Title-and-Description queries, only for TREC topics 151–200.



**Figure 4.** Correlation between P@10 vs P@11–20. (a) Title queries, all 250 TREC topics,  $r = 0.845$ . (b) Title-and-Description queries, all 250 TREC topics,  $r = 0.782$ .

In Figure 4, the matrices have the same axes and the numbers in each cell represent the number of occurrences of the respective precision pair (P@10, P@11–20, values ranging from 0.0 to 1.0). Therefore the value ‘4’ in Figure 4(a) on line 0.2 and column 0.3 shows that there were four queries in the data set where the initial query P@10 was 0.3 and P@11–20 was 0.2. The data clearly concentrate along the diagonals. The correlation coefficients in the range  $0.78 < r < 0.85$  confirm this.

The findings in Figures 3 and 4 lead us to the conclusion that we exclude the queries with both the worst and the best precision in Top-10 from the classification effort. If a searcher finds no relevant documents on the first page, she will probably reformulate her query rather than examine the second page. In addition, learning a good classifier with no relevant documents would be difficult and the second result page probably would have only a few relevant documents to identify. On the other hand, if the searcher finds 10 relevant documents on the first page, it is highly probable that her information need is already satisfied. In addition, learning a good classifier would be difficult and the second page probably would have many relevant documents, making their classification-based identification futile.

The two arguments, on the probable searcher behaviour and on the learnability of classifiers, support the view that efforts in classifying the second result page should be focused on cases where the first page precision is 0.1–0.9. In our experiments, we do not apply the classification approach, and exclude the original result, when the initial Top-10 precision is 0.0 or 1.0. Note that such a decision can be done in real life as well by examining the searcher’s RF.

### 3.2. Training results

The search space for the best classifiers is large because we examine three basic approaches (KMeans, KNN and naive Bayes); all can be used with the full or reduced feature set, there are several feature set reduction methods (Fisher exact

**Table 1.** Training phase results for reduction method.

	Topic set	Classification method	Feature selection method	Number of features
T queries	51–100	KMeans	Kendall–Tau	440
	101–150	KMeans	Kendall–Tau	430
	151–200	KMeans	Information Gain	300
	351–400	KMeans	Information Gain	250
	401–450	KMeans	Information Gain	600
T + D queries	51–100	KMeans	Kendall–Tau	240
	101–150	KMeans	Kendall–Tau	250
	151–200	KNN	Kendall–Tau	530
	351–400	KMeans	Kendall–Tau	400
	401–450	KMeans	Kendall–Tau	710

test, information gain, Kendall–Tau correlation, Pearson’s chi-square test, Spearman correlation coefficient and odds ratio), and various levels of feature set reduction can be applied. We employed extensive manual hill-climbing to explore the search space and to identify the best classification methods for reduction and associated feature selection methods and feature set sizes. The results are given in Table 1.

Table 1 gives for T, T + D queries and the indicated TREC topic sets the best performing classification method, the best performing feature selection method for the classifier, and the best number of features identified in training experiments. From Table 1 one may conclude that, overall, KMeans with information gain or Kendall–Tau feature reduction down to about 300–500 features is a reasonable choice for reduction method. KMeans with Information gain reduction with 300 features is applied for T queries. For the T + D queries, KMeans with Kendall–Tau reduction with 500 features is utilized. These selections of the number of features can be interpreted as arbitrary, but the main point is to reduce huge space of the features to some manageable and convenient size. This in turn improves the efficiency of the employed method. Moreover, selection of the number of features can be seen as an indication of the robustness of the reduction method.

In addition to the training for reduction method, we executed experiments with KMeans, KNN, naive Bayes and SVM methods without reduction. We report the results in Table 2 and Table 3 for three of them. Naive Bayes delivered inferior results in comparison to the others; therefore it was excluded from further experiments.

Regarding the KMeans clustering method parameters, we utilized two centroids, a maximum of 30 iterations and a convergence threshold 0.001. For KNN classification method  $K$  was set to one. A multinomial version of naive Bayes with Laplace smoothing was implemented for the naive Bayes classification method. Further, Euclidean distance was used as a distance metric between documents, and all documents were normalized before further processing by the respective algorithms.

During SVM training phase we could not achieve better results than what the other methods delivered. Having observed the classification results, the poor quality of the SVM could be attributed to data imbalance. The first page of the IR experiment results usually has a varying number of relevant and non-relevant documents. This could not be alleviated with the cost factor parameter in spite of many experiments conducted. The problem could be circumvented by balancing the training document numbers. We included in the training set only the minimum number of relevant and non-relevant documents for each set. That is, for example, if only two relevant and eight non-relevant documents were available in the first result page, the SVM training set was established by two relevant and two first non-relevant documents.

### 3.3. Test results

The test results are reported in Table 2 (for T queries) and Table 3 (for T + D queries). In Table 2, the first block (T\_51-100) reports results for the TREC topic set 51–100. The rows within this block report results for the four metrics employed. The columns are the initial query, the PRF baseline (with top two documents and 100 extension keys) and the classification-based results. PRF was applied as provided by Lemur Indri. The columns KNN, KMeans and SVM give results for the three classification methods without feature reduction. The column REDUC indicates the results for the selected classification method with a feature reduction indicated in the table caption – in this case KMeans/Information Gain with 300 features. The effectiveness values for each metric in the block T\_51-100 are the average effectiveness values obtained for the topic set T\_51\_100. The other blocks have analogous content; just the test sets vary. The final block

**Table 2.** Comparative experiment results (%) for Title queries.

Improvements over PRF change (%)											
Collection	Metric	BASE	PRF	KNN	KMeans	REDUC	SVM	KNN	KMeans	REDUC	SVM
T 51–100	NDCG@20	42.28	49.32	49.79	50.56	<b>50.75</b>	50.71	0.96	2.51	2.91	2.82
	NDCG@30	40.94	47.02	48.10	48.94 <sup>+</sup>	48.60	<b>49.54</b>	2.29	4.06	3.35	5.36
	P@20	41.21	49.66	50.34	51.38	51.55	<b>51.72</b>	1.39	3.47	3.82	4.17
	P@30	39.65	46.21	47.70	48.74 <sup>+</sup>	48.16	<b>49.65</b>	3.23	5.47	4.23	7.46
T 101–150	NDCG@20	43.96	44.90	47.89 <sup>+</sup>	<b>48.86*</b>	48.10 <sup>+</sup>	48.75*	6.68	8.82	7.13	8.59
	NDCG@30	41.90	44.60	46.28	46.93 <sup>+</sup>	46.31	<b>47.00<sup>+</sup></b>	3.76	5.23	3.83	5.38
	P@20	42.36	43.06	47.22 <sup>+</sup>	<b>48.61<sup>+</sup></b>	47.50 <sup>+</sup>	48.47 <sup>+</sup>	9.68	12.90	10.32	12.58
	P@30	39.91	43.24	45.09	45.93	45.09	<b>46.02<sup>+</sup></b>	4.28	6.21	4.28	6.42
T 151–200	NDCG@20	48.63	51.17	53.54*	52.86 <sup>+</sup>	<b>53.80*</b>	53.11*	4.62	3.30	5.14	3.79
	NDCG@30	46.30	49.08	50.50	50.42	50.76	<b>50.87</b>	2.90	2.74	3.44	3.67
	P@20	47.65	50.15	53.38*	52.50 <sup>+</sup>	<b>53.68*</b>	52.79 <sup>+</sup>	6.45	4.69	7.04	5.28
	P@30	44.12	46.86	48.53	48.63	48.82	<b>49.02</b>	3.56	3.77	4.18	4.60
T 351–400	NDCG@20	38.95	40.21	43.08*	43.09*	<b>43.85*</b>	42.70 <sup>+</sup>	7.14	7.16	9.06	6.21
	NDCG@30	36.15	37.43	39.21	39.64 <sup>+</sup>	<b>40.40*</b>	38.69	4.74	5.89	7.93	3.36
	P@20	35.27	36.62	40.41*	40.41*	<b>41.35*</b>	39.73 <sup>+</sup>	10.33	10.33	12.92	8.49
	P@30	30.54	31.80	33.78	34.32	<b>35.23<sup>+</sup></b>	32.97	6.23	7.93	10.77	3.68
T 401–450	NDCG@20	42.69	43.40	<b>46.78*</b>	44.46	44.94 <sup>+</sup>	45.54*	7.78	2.43	3.53	4.93
	NDCG@30	40.51	41.82	<b>44.69*</b>	42.56	42.63 <sup>+</sup>	43.74*	6.86	1.78	1.93	4.61
	P@20	37.50	38.10	<b>42.74*</b>	39.88	40.48	40.95 <sup>+</sup>	12.19	4.69	6.25	7.50
	P@30	31.99	33.25	<b>36.67*</b>	34.76	34.68	35.48 <sup>+</sup>	10.26	4.53	4.29	6.68
Average	NDCG@20	43.30	45.80	48.22*	47.96*	<b>48.29*</b>	48.17*	5.28	4.73	5.43	5.17
	NDCG@30	41.16	43.99	45.75*	45.70*	45.74*	<b>45.97*</b>	4.01	3.88	3.98	4.50
	P@20	40.80	43.51	46.82*	46.56*	<b>46.91*</b>	46.73*	7.59	6.99	7.80	7.40
	P@30	37.24	40.27	42.35*	42.47*	42.40*	<b>42.63*</b>	5.17	5.47	5.27	5.85

\*Statistically significant difference (Friedman,  $p < 0.01$ ) from the PRF results; <sup>+</sup> statistically significant difference (Friedman,  $p < 0.05$ ) from the PRF results (REDUC, reduction method, KMeans, information gain, number of features, 300).

in Table 2 shows the total average of the all test sets for each metric. The maximum value for every row is highlighted in bold type. The asterisks mark statistically significant differences of classification results compared with the PRF results. Almost all experiment results are statistically significant in comparison to the initial query results, but these are not marked separately. We employed Friedman tests with  $p < 0.01$  indicating high statistical significance. In addition, Friedman tests were also conducted with  $p < 0.05$ . In this case most of the results are statistically significantly different from PRF; these are marked separately on the tables with a plus sign.

Table 2 suggests that, in the case of short T queries, one of the classification approaches provides the best average performance. Most often the top approach is the reduction-based approach or SVM without feature set reduction. All classification-based methods constitute a statistically significant difference at the level of  $p < 0.01$ .

Table 3 has the same structure; the only difference is that queries here are longer (T + D). The results indicate that the maximum values for average appear in both the KNN and SVM columns. Even though all these methods have a statistically significant difference with regard to PRF baseline queries, they do not show any significant difference to each other.

#### 4. Discussion and conclusion

We have proposed an alternative approach to implement RF. Instead of query reformulation based on query expansion provided by RF documents, one learns classifiers from the PRF top results after simulated user RF. These classifiers are then applied to identify relevant documents among the subsequent documents in the result list. We addressed three research questions in the present paper (Section 2.1).

*RQ1:* Tables 2 and 3 indicate the effectiveness of the proposed classification approach. In case of the short T queries (Table 2), classification improves retrieval effectiveness over the initial query results by almost 11% at NDCG@20 and NDCG@30. For P@20 and P@30 the corresponding readings are > 14% and almost 14%, respectively. Over the PRF baseline, the improvements are smaller: > 5% at NDCG@20 and > 4% at NDCG@30. For P@20 and P@30 the



**Table 3.** Comparative experiment results (%) for T + D queries.

Improvements over PRF change (%)											
Collection	Metric	BASE	PRF	KNN	KMeans	REDUC	SVM	KNN	KMeans	REDUC	SVM
T + D 51–100	NDCG@20	46.86	53.01	55.76 <sup>+</sup>	<b>56.99*</b>	56.65*	55.99 <sup>+</sup>	5.19	7.51	6.87	5.62
	NDCG@30	44.83	51.95	54.82	<b>55.70*</b>	54.83*	53.66	5.53	7.23	5.54	3.30
	P@20	42.90	51.29	55.16 <sup>+</sup>	<b>56.94*</b>	56.45*	55.48 <sup>+</sup>	7.55	11.01	10.06	8.18
	P@30	41.29	50.32	53.98 <sup>+</sup>	<b>55.06*</b>	53.87 <sup>+</sup>	52.26	7.27	9.40	7.05	3.85
T + D 101–150	NDCG@20	44.65	46.58	49.00	48.88	48.50	<b>49.40<sup>+</sup></b>	5.18	4.92	4.12	6.03
	NDCG@30	42.05	45.13	46.86	46.88	46.57	<b>48.25*</b>	3.84	3.89	3.19	6.93
	P@20	42.35	45.15	48.68 <sup>+</sup>	48.53	47.94	<b>49.12*</b>	7.82	7.49	6.19	8.79
	P@30	39.31	43.53	45.69	45.78	45.29	<b>47.55*</b>	4.95	5.18	4.05	9.23
T + D 151–200	NDCG@20	49.52	52.92	<b>55.73*</b>	55.33*	54.73 <sup>+</sup>	54.87*	5.32	4.55	3.43	3.69
	NDCG@30	47.57	50.48	52.60*	52.10*	52.50 <sup>+</sup>	<b>52.93*</b>	4.21	3.21	4.00	4.86
	P@20	47.38	52.75	<b>56.63*</b>	56.00*	55.13 <sup>+</sup>	55.38*	7.35	6.16	4.50	4.98
	P@30	44.92	48.83	51.42 <sup>+</sup>	50.67 <sup>+</sup>	51.33 <sup>+</sup>	<b>52.00*</b>	5.29	3.75	5.12	6.48
T + D 351–400	NDCG@20	40.71	42.73	<b>46.24*</b>	45.45*	45.95*	45.40*	8.22	6.36	7.54	6.24
	NDCG@30	38.85	41.00	<b>43.47*</b>	42.58*	42.58*	42.45*	6.04	3.85	3.86	3.54
	P@20	36.43	38.45	<b>43.33*</b>	42.14*	42.86*	42.14*	12.69	9.60	11.46	9.60
	P@30	33.10	35.48	<b>38.25*</b>	37.14	37.06 <sup>+</sup>	36.91	7.83	4.70	4.48	4.03
T + D 401–450	NDCG@20	43.05	44.56	47.99*	46.45 <sup>+</sup>	47.64*	<b>48.00*</b>	7.69	4.25	6.91	7.71
	NDCG@30	42.12	44.04	45.60	44.58	46.36 <sup>+</sup>	<b>46.48<sup>+</sup></b>	3.56	1.24	5.29	5.56
	P@20	37.67	39.65	<b>44.30*</b>	42.33	43.84*	44.07*	11.73	6.74	10.56	11.14
	P@30	33.18	35.50	37.29	36.28	<b>38.29</b>	<b>38.29</b>	5.02	2.18	7.86	7.86
Average	NDCG@20	44.96	47.96	<b>50.94*</b>	50.62*	50.70*	50.73*	6.22	5.54	5.70	5.77
	NDCG@30	43.09	46.52	48.67*	48.37*	48.57*	<b>48.76*</b>	4.63	3.98	4.41	4.81
	P@20	41.35	45.46	<b>49.62*</b>	49.19*	49.24*	49.24*	9.15	8.20	8.32	8.31
	P@30	38.36	42.73	45.32*	44.99*	45.17*	<b>45.40*</b>	6.06	5.27	5.71	6.24

\*Statistically significant difference (Friedman,  $p < 0.01$ ) from the PRF results; <sup>+</sup> statistically significant difference (Friedman,  $p < 0.05$ ) from the PRF results (REDUC, KMeans; reduction method, Kendall–Tau; number of features, 500).

corresponding readings are  $> 7\%$  and  $> 5\%$ , respectively. This suggests that short initial query results can be improved to a useful degree by the proposed classification approach.

In case of the long T + D queries (Table 3), classification improves retrieval effectiveness over the initial query results by  $> 12\%$  at NDCG@20 and NDCG@30. For P@20 and P@30 the corresponding readings are around 19% and  $> 17\%$ , respectively. Over the PRF baseline, the improvements are smaller: almost 6% at NDCG@20 and  $> 4\%$  at NDCG@30. For P@20 and P@30 the corresponding readings are  $> 8$  and  $> 5\%$ , respectively. Even though long initial queries provide so much evidence to a modern search engine, classification methods can still improve the results by learning through top document RF.

In all, both the short and the long queries can be improved by the classification approach. Furthermore all classification methods provide statistically significantly better results over PRF and initial queries.

*RQ2:* Tables 2 and 3 indicate that feature set reduction is not more effective than using the full feature set in T + D queries but provides a marginal boost in the shorter T queries. The best classification methods for short queries are the reduction method and SVM with a full feature set most of the time, while the differences from the other classification methods are minor and statistically not significant. For long queries the best classification methods are KNN and again SVM with full feature set, but even this one provides only a minor advantage over other classification methods.

*RQ3:* the probable searcher behaviour, the learnability of classifiers and the high correlation of P@10 with P@11–20/30 supported the view that efforts in classifying the second/third result page should be focused on cases where the first page precision is between 0.1 and 0.9. In the case of T-queries, there were 178 (Figure 4) topics for which the initial query result fell in the P@10 range of 0.1–0.9. Therefore this 71% of the topics was responsible for the observed overall improvement. In the case of T + D queries, there were 190 topics for which the initial query result fell in the P@10 range of 0.1–0.9. So in this case about 76% of the topics were responsible for the observed overall effects.

Our findings are based on searcher simulation. Simulation entails using a symbolic model of a real-world system in order to study real-world problems. The model is a simplified representation of real world. The relevant features of the real world should be represented while other aspects may be abstracted out. We modelled searcher interaction features during RF and assumed feedback on the Top-10 of the PRF search results. Browsing only the Top-10 is quite realistic

while the assumption on RF for all documents in Top-10 may be little optimistic compared with observations on searcher behaviour in the IR literature. On the other hand, if the searcher broke off variably after identifying three relevant documents (on average before scanning the entire Top-10), the classification results might be better than what we report above. Other simulations with RF [22] have indicated this at least for the traditional query reformulation based RF.

Our experimental evaluation was based on user-oriented metrics,  $P@20/P@30$  and  $NDCG@20/NDCG@30$ . Compared with explicit query reformulation, RF and scanning one or two pages of classification-based results may be an option for the user, if RF is made convenient and classification is fast. Therefore the metrics  $@20$  and  $@30$  are relevant. In the future we aim to developing simulation of user interaction in IR towards more fine-grained models of user interaction. Namely we apply the ideas of user fallibility [9] in RF with a classification approach. We also plan to apply classification process and compare this approach with RF with various query expansion methods.

## Funding

This research was funded by Academy of Finland grant number 133021.

## References

- [1] Marchionini G, Dwiggins S, Katz A and Lin X. Information seeking in full-text end-user-oriented search systems: The roles of domain and search expertise. *Library and Information Science Research* 1993; 15(1): 35–70.
- [2] Efthimiadis EN. Query expansion. *Annual Review of Information Science and Technology* 1996; 31: 121–187.
- [3] Ruthven I and Lalmas M. A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review* 2003; 18(2): 95–145.
- [4] Ruthven I, Lalmas M and van Rijsbergen K. Incorporating user search behaviour into relevance feedback. *Journal of the American Society for Information Science and Technology* 2003; 54(6): 529–549.
- [5] Onoda T, Murata H and Yamada S. Relevance feedback document retrieval using Support Vector Machines. In *AM–2003 post-proceedings*. Springer Lecture Note 3430. Berlin: Springer, 2005, pp. 59–73.
- [6] Chen Z and Lu Y. *Using text classification method in relevance feedback. Intelligent information and database systems*. Berlin: Springer, 2010.
- [7] Sihvonen A and Vakkari P. Subject knowledge improves interactive query expansion assisted by a thesaurus. *Journal of Documentation* 2004; 60(6): 673–690.
- [8] Vakkari P and Sormunen E. The influence of relevance levels on the effectiveness of interactive information retrieval. *Journal for American Society for Information Science and Technology* 2004; 55(11): 963–969.
- [9] Baskaya F, Keskustalo H and Järvelin K. Simulating simple and fallible relevance feedback. *European Conference on Information Retrieval ECIR*, 2011, pp. 593–604.
- [10] Järvelin K. Interactive relevance feedback with graded relevance and sentence extraction: Simulated user experiments. In Cheoung D et al. (eds) *Proceedings of the 18th ACM conference on information and knowledge management (ACM CIKM'09)*, 2009, pp. 2053–2056.
- [11] Keskustalo H, Järvelin K and Pirkola A. Evaluating the effectiveness of relevance feedback based on a user simulation model: Effects of a user scenario on cumulated gain value. *Information Retrieval* 2008; 11(5): 209–228.
- [12] Lam-Adesina AM and Jones GJF. Applying summarization techniques for term selection in relevance feedback. In *Proceedings of the 24th Annual ACM conference on research and development in information retrieval*, 2001, pp. 1–9.
- [13] Jansen MJB, Spink A and Saracevic T. Real life, real users, and real needs: A study and analysis of user queries on the Web. *Information Processing and Management* 2000; 36(2): 207–227.
- [14] Stenmark D. Identifying clusters of user behaviour in intranet search engine log files. *Journal of the American Society for Information Science and Technology* 2008; 59(14): 2232–2243.
- [15] Hang L. *Learning to rank for information retrieval and natural language processing*. Morgan & Claypool, 2011.
- [16] Geng X, Liu TY, Qin T, Amold A, Li H and Shum HY. Query dependent ranking using K-nearest neighbor. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval (SIGIR '08)*, 2008, pp. 115–122; doi: 10.1145/1390334.1390356; <http://doi.acm.org/10.1145/1390334.1390356>
- [17] Azzopardi L, Järvelin K., Kamps J and Smucker MD. Simulated evaluation of interactive information retrieval. *SIGIR workshop proposal*, 2010, p. 3.
- [18] Manning C, Raghavan P and Schütze H. *Introduction to information retrieval*. Cambridge: Cambridge University Press, 2008.
- [19] Sebastiani F. *Machine learning in automated text categorization*. Pisa: Consiglio Nazionale delle Ricerche, 2002.
- [20] Joachims T, Making large-scale SVM learning practical. In Schölkopf B, Burges C and Smola A (eds) *Advances in kernel methods - Support Vector Learning*. Cambridge, MA: MIT Press, 1999.
- [21] Sotiris BK. Supervised machine learning: A review of classification techniques. *Informatica (Slovenia)* 2007; 31(3): 249–268.
- [22] Keskustalo H, Järvelin K and Pirkola A. The effects of relevance feedback quality and quantity in interactive relevance feedback: A simulation based on user modelling. In Lalmas M et al. (eds) *28th European conference on information retrieval ECIR '06*, 2006, Vol. 3936, pp. 191–204.



# Simulating Simple and Fallible Relevance Feedback

Feza Baskaya, Heikki Keskustalo, and Kalervo Järvelin

Department of Information Studies and Interactive Media,  
FIN-33014 University of Tampere, Finland

{Feza.Baskaya,Heikki.Keskustalo,Kalervo.Jarvelin}@uta.fi

**Abstract.** Much of the research in relevance feedback (RF) has been performed under laboratory conditions using test collections and either test persons or simple simulation. These studies have given mixed results. The design of the present study is unique. First, the initial queries are realistically short queries generated by real end-users. Second, we perform a user simulation with several RF scenarios. Third, we simulate human fallibility in providing RF, i.e., incorrectness in feedback. Fourth, we employ graded relevance assessments in the evaluation of the retrieval results. The research question is: how does RF affect IR performance when initial queries are short and feedback is fallible? Our findings indicate that very fallible feedback is no different from pseudo-relevance feedback (PRF) and not effective on short initial queries. However, RF with empirically observed fallibility is as effective as correct RF and able to improve the performance of short initial queries.

**Keywords:** Relevance feedback, fallibility, simulation.

## 1 Introduction

Query modification (QM) means query reformulation by changing its search keys (or modifying their weights) in order to make it better match relevant documents. Query formulation, reformulation, and expansion have been studied extensively because the selection of good search keys is difficult but crucial for good results. Real searchers' first query formulation often acts as an entry to the search system and is followed by browsing and query reformulations [9]. Relevance feedback (RF) based on initial query results and query expansion (QE) have been the main approaches to QM. Efthimiadis [2], Ruthven and Lalmas [11], Ruthven, Lalmas and van Rijsbergen [12] provide useful reviews of the techniques.

In the present paper we focus on interactive RF. In this method, users either point out relevant documents and the retrieval system infers the expansion keys for the feedback query, or the retrieval system presents a list of candidate expansion keys for the user to choose from. Knowledgeable experienced searchers may benefit more of RF because they recognize relevant vocabulary and are better able to articulate their needs initially [13]. Users also seem more likely to identify highly relevant documents than marginal ones [18].

There are two difficulties in providing feedback: searcher's capability and willingness [11]. Pseudo-relevance feedback (PRF) [11] avoids these challenges by assuming

that the first documents of an initial search result are relevant. Long documents and non-relevant documents however introduce noise in the PRF process thus causing query drift. To counteract this, one may use query-biased summaries [8], [16] for the identification of expansion keys. Lam-Adesina & Jones [8] and Järvelin [5] have shown that query-biased summaries positively affect PRF effectiveness. Yet another challenge to PRF is that real users tend to issue very short queries [4] and employ shallow browsing. As a consequence, the initial query results tend to be of poor quality and sparse regarding relevant documents, thus making PRF ineffective regarding the computational effort. Query-biased summaries may nevertheless counteract the latter to some degree [8].

Järvelin [5] argued that while RF is more effective than PRF, the performance difference does not justify the necessary searcher's effort. His results were however based on long queries (Title+Description). In the present paper we examine the effectiveness of RF and PRF under short initial queries. This is motivated by observed searcher behavior [4]. This leaves a chance for RF score higher than PRF since the initial performance may not be good enough for PRF to be effective.

However, searcher's capability to identify relevant documents may be limited. Humans are fallible. Turpin and colleagues [17] showed that snippets (i.e. query-biased summaries) are important in IR interaction and bad snippets may lead to incorrect relevance decisions. Vakkari and Sormunen [18] showed that humans may well err on marginal and non-relevant documents while are likely to identify the highly relevant ones correctly. Foley and Smeaton [3] examine collaborative IR where the collaborators may err. These findings suggest that the effect of correctness of RF should be examined. Since searcher performance may vary greatly across situations, we investigate in the present paper a range of fallibility scenarios.

Some earlier studies [3] and [5] suggest that RF is most effective when little feedback is given as early as possible – that is, the searcher should identify one or two first relevant documents in the initial result and stop browsing there. One should not be picky regarding the quality of the feedback documents, i.e. marginal ones would do. Therefore in the present study, our main RF scenario is based on shallow browsing (max top-10) and identifying the first two relevant documents of whatever relevance degree (perhaps erroneously) as feedback.

We base our experiments on searcher simulation (like [3] and [7]) rather than tests with real users. Simulation has several advantages, including cost-effectiveness and rapid testing without learning effects as argued in the SIGIR SimInt 2010 Workshop [1]. Besides, the simulation approach does not require a user interface. The informativeness and realism of searcher simulation can be enhanced by explicitly modeling, in the present case, those aspects of searchers and RF that pertain to RF effectiveness. In the present paper, two issues are significant: (a) realistic short queries, and (b) realistic fallibility of searchers' relevance judgments. While we perform our study in a test collection, we employed test persons to generate short queries (length 1 – 3 words). These are more realistic and controllable than, e.g. the title elements of TREC topics. To study the effects of fallibility, we employ several fallibility scenarios ranging from random judgments to perfect judgments with one scenario based on the empirical findings by Vakkari & Sormunen [18]. We implement them as probability distributions over possible degrees of relevance. In this way, we may employ both analytical variety and empirical grounding in our simulations.

Our evaluations are based on three metrics (MAP, P@10 and P@20) and three levels of relevance. Regarding the metrics, the main role is given to P@10 and P@20 as clearly user-oriented measures – users frequently avoid browsing beyond the first results page, i.e. 10 links/documents [4]. After giving RF and already browsing up to 10 documents, the P@20 can be seen as evaluation for quasi first page. For comparison, MAP is reported as well. The three levels of evaluation are liberal (i.e. even marginal documents are taken as relevant), fair (medium and highly relevant documents are relevant), and, strict (only highly relevant documents matter). This is justified because the user may not benefit from many marginal documents at all, and because there are systematic performance differences across the evaluation levels.

We utilize the TREC 7-8 corpus with 41 topics for which graded relevance assessments are available [14]. The search engine is Lemur. The fallibility simulations are based on the relevance degrees of documents given in the recall base of the test collection (the qrels files) and probability distributions across the possible (partially erroneous) simulated user judgments. A random number generator is used to drive the judgments. All experiments are run 50 times with random decisions and the reported results are averages over the 50 runs. We will use PRF results as baselines to our simulated RF experiments.

## 2 Study Design

### 2.1 Research Questions

Our overall research question is: how does RF affect IR performance when short initial queries are employed and fallible feedback is provided? More specifically:

- RQ 1: How effective are PRF and RF when employed on the results of short initial queries and shallow browsing?
- RQ 2: Does RF effectiveness seriously deteriorate when RF is of progressively lower quality?
- RQ 3: How does RF effectiveness in RQ2 depend on evaluation by liberal, fair vs. strict relevance criteria?

### 2.2 The Test Collection, Search Engine, and Query Expansion Method

We used the reassessed TREC 7-8 test collection including 41 topics [14]. The document database contains 528155 documents indexed under the retrieval system *Lemur Indri*. The index was constructed by stemming document words. The relevance assessments were done on a four-point scale: (0) irrelevant, (1) marginally relevant, (2) fairly relevant, and (3) highly relevant document. In the recall base there are on average 29 marginally relevant, 20 fairly relevant and 10 highly relevant documents for each topic. For three topics there were no highly relevant documents. This recall base with its intrinsic human judgment errors is taken as a gold standard for further fallibility study and evaluation.

The research questions do not require any particular interactive query expansion method to be employed. We simulate interactive RF that takes place at document level: the simulated users point to relevant documents and the RF system then automatically extracts the expansion keys. We follow Tombros and Sanderson [16], Lam-Adesina & Jones [8] and Järvelin [5] who have shown that query-biased summaries positively affect RF effectiveness. Given a query and an indicated relevant document, our QE method ranks the document sentences by their query similarity, then extracts the top- $n$  ( $n=5$ ) sentences, and then collects the non-query words from these sentences, scores them by their ( $tf \cdot idf$  based) discrimination power, and chooses the top- $k$  ( $k=30$ ) most significant words as expansion keys to be appended to the RF query. When multiple documents are indicated for feedback, top- $n$  sentences are collected from each and then pooled before sentence scoring and key extraction. The parameter values for  $n$  and  $k$  were found reasonable in prior studies [5]. When scoring sentences, if a non-stop query word did not match any sentence word, an  $n$ -gram type of approximate string matching with a threshold was attempted [10].

Initial short queries, 1-3 words in length, were constructed based on real searchers' suggestions (see below) but the query keys were stemmed. Multi-word queries were constructed as bag-of-word queries. Feedback queries were constructed by appending the feedback keys to the initial query as a second bag-of-words.

### 2.3 User Modeling for RF Simulation

The design of RF simulation requires several decisions to be made: (1) user's willingness to browse the initial result, (2) user's willingness to provide RF, (3) the level of relevance of the RF documents, and (4) user's fallibility in making relevance judgments. The first three decisions are suggested in Keskustalo and colleagues [7] as a user model. Their general recommendation was also that RF is most effective when the browsing depth is shallow (we use 10 documents here), when only little RF is given as early as possible (we provide the first two relevant document as RF, and then stop to browse), and that even marginal documents as RF as early as possible are better than highly relevant documents given late (we provide the first two relevant document as RF whatever their degree of relevance). Järvelin [5] confirmed these findings. In these simulation studies, the recall base of the test collection was used as the source of relevance judgments for RF. This means that the initial query result was scanned and each document ID on the ranked list was checked against the recall base of the topic in question.

The fourth decision, on human fallibility, is a novelty in RF simulation. This is motivated by Turpin and colleagues [17] and Vakkari and Sormunen [18], who point out errors in human relevance judgments. In the present study, the recall base is still a source in relevance judgment, but not taken as a fact as such. We simulate users that with some probability make correct judgments, and with some other probabilities err more or less. We have thus a probability distribution around the correct judgment. For example, such a distribution could state for a document of relevance degree, say 'fair', that there is a 10% probability for the user to assess the document as non-relevant, 20% probability as marginal, 50% as fair (correct), and 20% as highly relevant. Table 1 summarizes the fallibility scenarios employed in the present study.

**Table 1.** Fallibility probability distributions

Fallibility Scenario	Human Judgment Probabilities				
	n	m	f	h	
<b>1.00</b>	n	<b>1.0</b>	0.0	0.0	0.0
	m	0.0	<b>1.0</b>	0.0	0.0
	f	0.0	0.0	<b>1.0</b>	0.0
	h	0.0	0.0	0.0	<b>1.0</b>
<b>0.75</b>	n	<b>0.75</b>	0.125	0.075	0.05
	m	0.10	<b>0.75</b>	0.10	0.05
	f	0.05	0.10	<b>0.75</b>	0.10
	h	0.05	0.075	0.125	<b>0.75</b>
<b>0.50</b>	n	<b>0.50</b>	0.25	0.15	0.10
	m	0.20	<b>0.50</b>	0.20	0.10
	f	0.10	0.20	<b>0.50</b>	0.20
	h	0.10	0.15	0.25	<b>0.50</b>
<b>0.25</b>	n	<b>0.25</b>	0.25	0.25	0.25
	m	0.25	<b>0.25</b>	0.25	0.25
	f	0.25	0.25	<b>0.25</b>	0.25
	h	0.25	0.25	0.25	<b>0.25</b>
<b>0.50-0.80</b>	n	<b>0.5</b>	0.4	0.1	0.0
	m	0.4	<b>0.5</b>	0.1	0.0
	f	0.0	0.1	<b>0.8</b>	0.1
	h	0.0	0.0	0.2	<b>0.8</b>

fairly consistent to fully random. The final set, labeled as 0.50-0.80, is based on Vakari and Sormunen's [18] empirical findings. They reported that searchers are able to recognize highly relevant documents quite consistently but tend to err on marginal and non-relevant ones. Also Sormunen [14] found the judges inconsistent: most inconsistency occurred between neighboring relevance classes. Therefore the scenarios in Table 1 allocate intuitively more of the probability mass to neighboring classes than to more distant ones.

In our simulations, we use a random number generator together with the judgment scenarios to drive simulated relevance judgments. Because RF effectiveness is bound to be sensitive to random judgments, we run each RF experiment 50 times over and report the average effectiveness.

## 2.4 Short Initial Queries

Test collections such as the TREC collections provide their test topics structured as titles (T), descriptions (D), and narratives (N). In our TREC7-8 test collection, the titles of the 41 topics vary in length from 1 to 3 words, with 2.4 words average. The descriptions have an average length of 14.5 words. Real-life searchers often prefer very short queries [4] [15]. Jansen and colleagues [4] analyzed transaction logs containing thousands of queries posed by Internet search service users. They discovered that one in three queries had only *one* keyword. The average query length was 2.21 keys. Less than 4 % of the queries in Jansen's study had more than 6 keywords. The

In Table 1, the row sets represent fallibility scenarios. The first set, labeled 1.00, represents the gold standard for RF, always correct judgments of the feedback documents. The rows within 1.00 represent ground truth relevance of non-relevant (n), marginal (m), fair (f), and highly relevant (h) documents. The human judgment probabilities in columns represent the simulated human judgments. In the gold standard all judgments are correct, indicated by probability 1.0 in the diagonal.

The next three sets are labeled as 0.75, 0.50, and 0.25, indicating progressively more random judgments among the retrieved ranked documents, from

average number of keywords per query was even less, 1.45, in Stenmark's study [15], focusing on intranet users. Therefore it makes sense to test the effectiveness of initial queries of length of 1 to 3 words in RF scenarios. A further point is that test collection topic titles are carefully crafted to summarize each topic whereas end users are rather characterized by trial-and-error carelessness. Therefore we wanted to have end-user created short queries for our experiments.

The 41 topics were analyzed intellectually by test persons to form query candidate sets. A group of seven undergraduate information science students performed the analysis. Regarding each topic a printed topic description and a task questionnaire were presented for the test persons. Each of the 41 topics was analyzed by a student. The subjects were asked to directly select and think up good search keys from topical descriptions and to create various query candidates.

First a two-page protocol explaining the task was presented by one of the researchers. Information in the description and narrative fields of the test collection topics was presented to the users. Descriptions regarding non-relevance of documents were omitted to make the task more manageable within the time limitation of 5 minutes per topic. The test persons were asked to mark up all potential search words directly from the topic description and to express the topic freely by their own words. Third, they were asked to form various query candidates (using freely any kinds of words) as unstructured word lists: (i) the query they would use first ("1<sup>st</sup> query"); (ii) the one they would try next, assuming that the first attempt would not have given a satisfactory result ("2<sup>nd</sup> query"). Finally, the test persons were asked to form query versions of various lengths: (iii) one word (1w), (iv) two words (2w), and (v) three or more words (3w+). The very last task was to estimate how appropriate each query candidate was using a four-point scale. During the analysis the test persons did not interact with a real IR system.

In the present experiment, we used the short queries, ranging from 1 to 3 words, from this data set as the initial queries. The results of these were subject to RF under various feedback and fallibility scenarios.

## 2.5 Experimental Protocol

Figure 1 illustrates the overall experimental protocol. TREC topics are first turned to initial short queries (stemmed) of given length and executed with Lemur, followed by feedback document selection. This is based on the simulated searcher's feedback scenario (in the present experiments browsing up to 10 documents and returning the first two documents fallibly judged relevant as RF). The random judgments were repeated 50 times. In each case, the feedback documents for each query are split into sentences, and the sentences are scored on the basis of the query word scores. Word to word matches are facilitated by stemming and, in the case of Out-of-Vocabulary words (OOVs), by n-gram string matching. The sentences are ranked and the  $k$  best ones are extracted for each document. After processing the feedback documents, the  $m$  ( $m=5$ ) overall best sentences are identified for expansion key extraction. For each query's set of feedback sentences, their non-query, non-stop words are ranked by their scores and the 30 overall best keys are identified as expansion keys for the query and

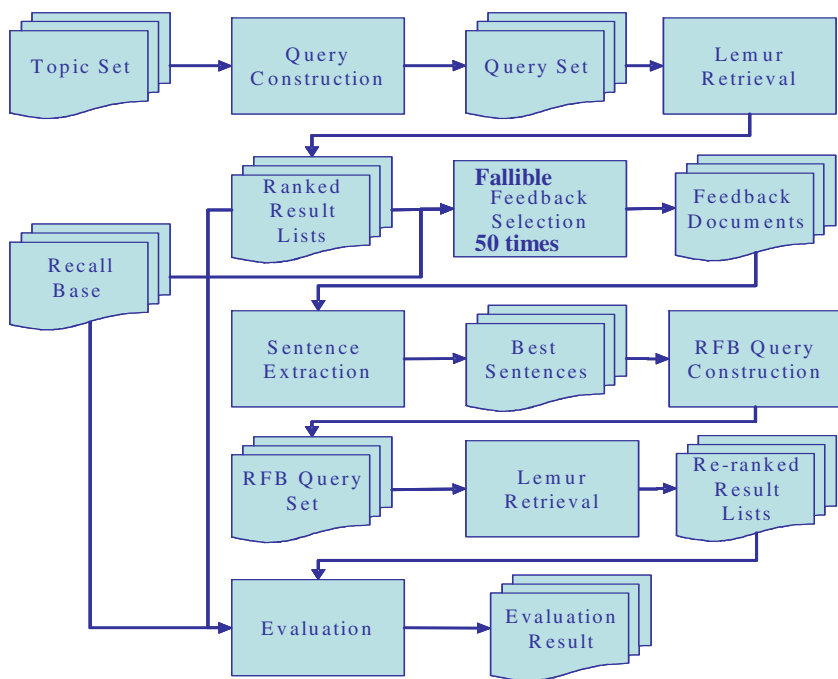


Fig. 1. Query-biased summarization process

added to the initial query. The new query is executed and both the original and feedback query results go to evaluation.

## 2.6 Evaluation and Statistics

In evaluation we employ full freezing (e.g. [7]) of the all documents 'seen', this is, (1) freezing all initially scanned (say,  $f$ ) documents for RF, relevant or not, at their ranks, (2) removing all initially seen documents from the RF query result, and (3) then filling the positions from  $f+1$  with the feedback query results. We use standard evaluation metrics available in the TREC-eval package and report evaluation results for  $P@10/20$  documents, and mean average precision MAP. The former are motivated by real life findings – people most often are precision-oriented and avoid excessive browsing – great results beyond the first pages don't matter. We employ liberal RF but three final evaluation levels, where liberal accepts all at least marginal documents as relevant, fair accepts all at least fairly relevant as relevant, and strict only highly relevant as relevant. Statistical testing is based on Friedman's test between RF runs and the baseline. PRF on the initial query result provides the stronger baseline, and therefore PRF is used as the baseline when statistical significance is evaluated. We ran several PRF experiment with 1, 2, 5 and 10 PRF documents. We report results for 2 PRF documents because using more did not consistently improve effectiveness.

### 3 Findings

#### 3.1 Initial and PRF (Baseline) Queries

Table 2 reports the initial query performances for user-defined one, two and three-word queries, as well as PRF queries at the three evaluation scenarios (liberal, fair and strict). The best query performance values are indicated by dark gray background. We see, among others, that the initial one-word queries are 3.4 (at fair evaluation) to 4.2 (at liberal) % units (MAP) weaker than 2-word queries except at strict level. Initial query MAP values for 3-word queries are 1.9 (at fair) to 4.0 (at liberal) % units better than one-word initial query values, and 1.3 (at fair) to 2.3 (at liberal) % units better than two-word query results. At strict evaluation results are slightly worse than one-word query results. On the other hand, P@10 initial values for two-word queries improve continuously the initial one-word query results from 9.3 % units (at liberal) to 1.1 % units (at strict). Compared to one-word queries, P@10 initial values for three-word queries improve also the initial results from 10.8 % units (at liberal) to 1.8 % units (at strict). P@20 initial query values for two-word queries improve continuously the initial query results from 7.8 % units (at liberal) to 1.4 % units (at strict). P@20 initial values for three-word queries improve also the initial results for one-word queries.

The PRF for one-word queries improves both MAP and P@10 only around 1 % and 0.5 % units respectively at liberal evaluation. At strict evaluation it decreases the MAP reading 1.7 %. The greatest PRF improvement in P@10 for one-word queries is 0.5 % units (at liberal). We can confirm earlier findings that tighter evaluation weakens PRF effectiveness [6]. The greatest PRF improvements in MAP for two-word queries are from 1.8 % units (at liberal) to 0.5 % units (at fair). The greatest PRF improvements in P@10 for two-word queries are 2.4 % units (at strict) to 1.0 % units (at fair) and in P@20 for two-word queries are 2.2 % units (at liberal) to 0.2 % units (at fair). When initial query length grows, the initial query effectiveness grows greatly, e.g. with liberal evaluation, P@10 grows by 10.7 % units and P@20 grows by 8.3% units. Likewise, the PRF to initial query effectiveness for P@10 improves by 3.9 % – 2.6% units depending on query length and evaluation stringency. Further, the shorter the initial queries are, the less PRF contributes. Thus PRF seems not capable of improving poor initial results. These findings hold for all evaluation metrics.

The findings above are deliberately for short initial queries reflecting real life searcher behavior. PRF on top of the RF query results (with no fallibility) did not yield any improvement.

#### 3.2 Expanded Runs and Fallibility in the Process

Table 2 also reports RF query effectiveness for all metrics (MAP, P@10 and P@20) under several user fallibility and evaluation scenarios. Refer to Table 1 for the explanation of the fallibility scenarios. Friedman’s test indicates overall significant statistical differences in each block of experiments defined by initial query length, metric and evaluation scenario ( $p < 0.05$ ). This allows examining the pair wise significant differences among the results in each block. Table 2 indicates (by ‘\*’) those pair wise differences between the PRF as baseline and fallible RF that are significant at the risk



**Table 2.** Simulated RF effectiveness for short queries

Queries		Liberal			Fair			Strict		
Fallibility		MAP	P@10	P@20	MAP	P@10	P@20	MAP	P@10	P@20
1-Word	Initial	0.143	0.246	0.209	0.164	0.210	0.171	0.190	0.111	0.080
	PRF	0.151	0.251	0.212	0.164	0.210	0.171	0.173	0.111	0.079
	1.00	0.161	0.261	0.243*	0.172*	0.215	0.192*	0.195*	0.108	0.090
	0.75	0.159	0.258	0.235	0.170	0.213	0.186	0.194	0.107	0.087
	0.50	0.158	0.257	0.232	0.169	0.213	0.182	0.193	0.108	0.085
	0.25	0.154	0.253	0.223	0.166	0.210	0.175	0.191	0.107	0.081
	0.5-0.8	0.161	0.261	0.242*	0.172*	0.215	0.191*	0.195*	0.108	0.089
2-Word	Initial	0.185	0.339	0.287	0.198	0.278	0.224	0.178	0.121	0.095
	PRF	0.203	0.356	0.309	0.203	0.288	0.227	0.192	0.145	0.097
	1.00	0.215	0.376	0.334*	0.218	0.302	0.243	0.197*	0.145	0.109
	0.75	0.213	0.376	0.330	0.216	0.305	0.241	0.195	0.145	0.108
	0.50	0.210	0.373	0.324	0.213	0.302	0.236	0.192	0.143	0.106
	0.25	0.206	0.367	0.315	0.209	0.298	0.231	0.189	0.141	0.102
	0.5-0.8	0.215*	0.378	0.336*	0.218*	0.306	0.244	0.196*	0.145	0.110*
3-Word	Initial	0.183	0.354	0.292	0.182	0.266	0.209	0.187	0.129	0.095
	PRF	0.209	0.393	0.326	0.199	0.305	0.235	0.195	0.155	0.107
	1.00	0.219	0.400	0.339	0.204	0.295	0.237	0.205	0.153	0.108
	0.75	0.217	0.394	0.339	0.203	0.291	0.237	0.203	0.151	0.109
	0.50	0.215	0.389	0.338	0.200	0.287	0.237	0.201	0.149	0.107
	0.25	0.208*	0.380	0.328	0.194*	0.281*	0.230	0.196	0.145	0.103
	0.5-0.8	0.220	0.398	0.340	0.205	0.294	0.238	0.205	0.151	0.109

**Legend:** \* indicates statistically significant difference to PRF baseline, Friedman’s test,  $p < 0.05$ .

level  $p < 0.05$ . In Table 2, background shading indicates the best performance in each column – lighter shading the strongest initial query and darker shading the strongest (P)RF query. PRF is also highlighted with a gray background.

Correct RF nearly always yields better effectiveness than PRF, but the difference is not always statistically significant. In MAP the difference is 0.6 to 2.2 % units, in P@10, -1.0 to 2.0 % units, and in P@20, 0.1 to 3.1 % units depending on initial query length and evaluation scenario. In MAP, there is a tendency for the difference to grow by tighter evaluation. In P@10 and P@20, the difference of correct feedback to PRF diminishes by tightening the evaluation. While both PRF and correct RF generally benefit from growing query length, PRF seems to benefit more.

The distribution of the fallibility results for MAP, P@10 and P@20 follows the judgment capability of the user. As the probability of incorrect judgments increases, the results are decreasing. A clear trend between 100 % correct RF and random RF (fallibility 0.25) is that the latter delivers worse results. Random RF rarely yields results significantly different from PRF, which was expected. While both generally

yield some improvement over the initial query baseline, the difference is not significant and tends to shrink by tighter evaluation criteria, being sometimes negative by strict criteria. Further, better relevance judgment capability clearly improves the results. In case of fallibility 0.75 the results are slightly better than with fallibility 0.5. The empirically grounded fallibility in RF is never significantly different in effectiveness from correct RF. The difference is  $\pm 0.4$  % units. This means that RF with empirically observed fallibility is as good as correct RF.

In summary, when initial queries are realistically short, the initial query results are relatively weak. This renders blind techniques, PRF and random RF ineffective. There is room for effective human interaction even when the initial queries are short. Despite their fallibility, humans can identify the relevant bits in poor results reliably enough for the benefit of their searching. However, RF requires human effort while PRF is automatic. The practical effectiveness difference is not material.

## 4 Discussion and Conclusion

Simulation entails using a symbolic model of a real-world system in order to study the real-world problems. The model is a simplified representation of the real world. The relevant features of the real world should be represented while other aspects may be abstracted out. This motivates our present study in which we model user interaction features during RF and vary them systematically. The validity of our simulation model is justified by observations in IR literature regarding query lengths, RF behavior and relevance judgment fallibility.

We started our simulation experiment by discussing relevant features of the real world searching. In the most general level one can observe that interaction is vital in real life IR. Secondly, individual users vary greatly. However, typical real life user interaction can be characterized as being simple and error-prone, more specifically: (1) searchers prefer using short (or even very short) queries; (2) searchers prefer shallow browsing (e.g., at most the top-10 documents observed, not top-1000); (3) searchers may be reluctant to give RF, (4) even if they are eager to give RF, they may make errors.

In the present paper we performed a simulation based on modeling real life features listed above, in other words, (1) very short initial queries are used (one, two, and three-word queries); (2) shallow browsing is assumed (at most top-20 documents per query); (3) PRF is also modeled, because it avoids requiring direct RF from the user; (4) fallibility is modeled based on several scenarios assuming that the simulated user makes errors during the selection of feedback documents. These scenarios range from assuming perfect user judgments (no errors) to random judgments (lots of errors). Importantly, we also construct a scenario based on empirical findings on the level of fallibility when the user attempts to recognize relevant documents belonging to various relevance levels [18]. In all, five different fallibility scenarios were studied. All experiments were run 50 times with random decisions and the reported results were averaged over the 50 runs.

Evaluation of the experiments was based on user-oriented measures, P@10 / P@20, and the traditional system-oriented measure, MAP. We used three distinct relevance levels because in real life different kinds of users exist. Some users prefer

finding mixed-level documents, while others want to focus on the best (highly-relevant) documents. We used full freezing during evaluation because it closely imitates the point of view of a real user who has wasted effort in inspecting any number of documents, regardless of their relevance level.

Regarding the first research question, our results suggest that using query-biased summaries is a promising method to approach both PRF and direct user-RF when initial very short queries are assumed. For the second research question we observed that although increasing fallibility decreases the performance compared to perfect RF, it is slightly better than the best performing PRF. Surprisingly, RF with a realistic level of fallibility yields results that are close to perfect RF. Third, when realistic fallibility is assumed and a user-oriented evaluation measure ( $P@10/P@20$ ) is used, at the liberal relevance level RF systematically improves the performance of all short-query types (one word, two word, and three word queries). However, when strict evaluation is demanded, RF does not improve the performance of all short queries against PRF (Table 2). This suggests that the results of very short initial queries do not provide often enough sufficiently good RF documents even for human eyes. This may in part explain the low pick-up rate of RF in real life. Searchers rather issue a new query.

In the future we aim at developing simulation of user interaction in IR toward more fine-grained models of user interaction.

## Acknowledgement

This research was funded by Academy of Finland grant number 133021.

## References

1. Azzopardi, L., Järvelin, K., Kamps, J., Smucker, M.: Report on the SIGIR 2010 Workshop on the Simulation of Interaction. *SIGIR Forum* 44(2), 35–47 (2010)
2. Efthimiadis, E.N.: Query expansion. In: Williams, M.E. (ed.) *Annual Review of Information Science and Technology ARIST*, vol. 31, pp. 121–187. Information Today, Inc., Medford (1996)
3. Foley, C., Smeaton, A.F.: Synchronous Collaborative Information Retrieval: Techniques and Evaluation. In: Boughanem, M., Berrut, C., Mothe, J., Soule-Dupuy, C. (eds.) *ECIR 2009*. LNCS, vol. 5478, pp. 42–53. Springer, Heidelberg (2009)
4. Jansen, M.B.J., Spink, A., Saracevic, T.: Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *Information Processing & Management* 36(2), 207–227 (2000)
5. Järvelin, K.: Interactive Relevance Feedback with Graded Relevance and Sentence Extraction: Simulated User Experiments. In: Cheung, D., et al. (eds.) *Proceedings of the 18th ACM Conference on Information and Knowledge Management (ACM CIKM 2009)*, Hong Kong, November 2–6, pp. 2053–2056 (2009)
6. Keskustalo, H., Järvelin, K., Pirkola, A.: The Effects of Relevance Feedback Quality and Quantity in Interactive Relevance Feedback: A Simulation Based on User Modeling. In: Lalmas, M., MacFarlane, A., Rüger, S.M., Tombros, A., Tsirikla, T., Yavilinsky, A. (eds.) *ECIR 2006*. LNCS, vol. 3936, pp. 191–204. Springer, Heidelberg (2006)

7. Keskustalo, H., Järvelin, K., Pirkola, A.: Evaluating the Effectiveness of Relevance Feedback Based on a User Simulation Model: Effects of a User Scenario on Cumulated Gain Value. *Information Retrieval* 11(5), 209–228 (2008)
8. Lam-Adesina, A.M., Jones, G.J.F.: Applying Summarization Techniques for Term Selection in Relevance Feedback. In: *Proc. of the 24th Annual ACM Conference on Research and Development in Information Retrieval*, pp. 1–9. ACM Press, New York (2001)
9. Marchionini, G., Dwiggins, S., Katz, A., Lin, X.: Information seeking in full-text end-user-oriented search systems: The roles of domain and search expertise. *Library and Information Science Research* 15(1), 35–70 (1993)
10. Pirkola, A., Keskustalo, H., Leppänen, E., Känsälä, A.-P., Järvelin, K.: Targeted S-Gram Matching: A Novel N-Gram Matching Technique for Cross- and Monolingual Word Form Variants. *Information Research* 7(2) (2002), <http://InformationR.net/ir/7-2/paper126.html>
11. Ruthven, I., Lalmas, M.: A survey on the use of relevance feedback for information access systems. *Knowledge Engineering Review* 18(2), 95–145 (2003)
12. Ruthven, I., Lalmas, M., van Rijsbergen, K.: Incorporating user search behaviour into relevance feedback. *Journal of the American Society for Information Science and Technology* 54(6), 529–549 (2003)
13. Sihvonen, A., Vakkari, P.: Subject knowledge improves interactive query expansion assisted by a thesaurus. *J. Doc.* 60(6), 673–690 (2004)
14. Sormunen, E.: Liberal Relevance Criteria of TREC - Counting on Negligible Documents? In: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 320–330. ACM Press, New York (2002)
15. Stenmark, D.: Identifying Clusters of User Behavior in Intranet Search Engine Log Files. *Journal of the American Society for Information Science and Technology* 59(14), 2232–2243 (2008)
16. Tombros, A., Sanderson, M.: Advantages of query biased summaries in information retrieval. In: *Proc. of the 21st Annual ACM Conference on Research and Development in Information Retrieval*, pp. 2–10. ACM Press, New York (1998)
17. Turpin, A., et al.: Including Summaries in System Evaluation. In: *Proc. of the 32nd Annual ACM Conference on Research and Development in Information Retrieval*, pp. 508–515. ACM Press, New York (2009)
18. Vakkari, P., Sormunen, E.: The influence of relevance levels on the effectiveness of interactive IR. *J. Am. Soc. Inf. Sci. Tech.* 55(11), 963–969 (2004)

# Time Drives Interaction: Simulating Sessions in Diverse Searching Environments

Feza Baskaya, Heikki Keskustalo, Kalervo Järvelin  
School of Information Sciences  
FI-33014  
University of Tampere, Finland  
{ Feza.Baskaya, Heikki.Keskustalo, Kalervo.Jarvelin }@uta.fi

## ABSTRACT

Real life information retrieval takes place in sessions, where users search by iterating between various cognitive, perceptual and motor subtasks through an interactive interface. The sessions may follow diverse strategies, which, together with the interface characteristics, affect user effort (cost), experience and session effectiveness. In this paper we propose a pragmatic evaluation approach based on scenarios with explicit subtask costs. We study the limits of effectiveness of diverse interactive searching strategies in two searching environments (the scenarios) under overall cost constraints. This is based on a comprehensive simulation of 20 million sessions in each scenario. We analyze the effectiveness of the session strategies over time, and the properties of the most and the least effective sessions in each case. Furthermore, we will also contrast the proposed evaluation approach with the traditional one, rank based evaluation, and show how the latter may hide essential factors that affect users' performance and satisfaction - and gives even counter-intuitive results.

## Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Search process

## Keywords

Session-based evaluation, simulation, time-based evaluation

## 1. INTRODUCTION

Interaction through search interface and environment greatly affects the user behavior, user experience, and user performance.

Many earlier studies have extended the traditional Cranfield view of IR and discussed various aspects of interactive searching (see, e.g., [4], [5], [6], [13], [21]), user interaction, and query modification (see, e.g., [3], [10], [14], [28]).

During interaction the user selects between *subtasks*, e.g., whether to scan the result or launch a new query instead, and how to construct the query. Such selections obviously affect session gains. However, different subtasks also have costs, e.g., they take time. This is important because real life IR often takes place under (time)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGIR '12, August 12-16, 2012, Portland, Oregon, USA.

Copyright 2012 ACM 978-1-4503-1472-5/12/08...\$10.00.

constraints. In particular, keeping the overall session cost reasonable may be essential for end users.

The costs of subtasks may vary for many reasons between searching environments. For example, regarding the query side, small devices and touch screens are inconvenient for typing [11]. Recently, novel kinds of searching devices, including personal phone-based mobile devices, have become increasingly popular.

In order to minimize the overall session costs, a mobile phone user might e.g., avoid typing and prefer result scanning. Low input costs might change the situation from the user's point of view, leading to longer queries. Therefore, if we assume two users having identical needs and identical cost constraints regarding the overall session time, it is possible that different devices render different subtask combinations optimal in searching.

Traditional IR evaluation focuses on the quality of the ranked output. In this view, the costs of posing queries are non-problematic, even uninteresting. In this paper we will utilize simple scenarios to bring time factors into the research setting. Scenarios formalize and quantify the gains and costs of interactive sessions. We construct two cases – a personal desktop computer (PC) and a smart phone (SP) case, with subtask costs derived from the literature. We will simulate session interaction involving multiple queries based on prototypical but empirically grounded query modification strategies using a test collection. We then explore the effectiveness of searching via the exhaustive set of querying-scanning combinations possible, and evaluate the effectiveness of both scenarios in terms of Cumulated Gain (CG) [16] under time constraint (overall session time). We use non-normalized metrics, because normalized metrics may yield misleading results, especially if time is taken into account.

Early papers on IR evaluation had a comprehensive approach to interactive IR evaluation. Cleverdon et al. [8] pointed out, among others, physical and intellectual user effort as an important factor in IR evaluation. Salton [24] identified user effort measures in the context of IR evaluation. More recently Su [30] gave a comparison of 20 different evaluation measures for interactive IR, including actual cost of search, several utility measures, and worth of search results vs. time expended. The interactive aspect of IR requires attention because previous studies have repeatedly shown that discrepancy exists between interactive and non-interactive evaluation results. Hersh et al. [12] showed that a weighting scheme giving maximum improvement over the baseline in non-interactive batch evaluation failed to surpass others when real users performed a simulated task. Turpin and Hersh [31] observed that a system superior over the baseline in batch evaluation, measured by mean

average precision, was not superior in an interactive situation. Turpin and Scholer [32] found no significant relationship between the search engine effectiveness measured by mean average precision and real user success in a precision-oriented task. Smith and Kantor [25] observed that users of degraded systems were as successful as those using non-degraded systems. They suggested that users achieved this by altering their behavior.

Dunlop [9] proposed “time-to-view” graphs, which incorporate user interface and system as well as the time component into the same framework for evaluation of system effectiveness. However he did not analyze time constraints, query modification strategies and different devices.

Smucker [26] brought time factors into the traditional Cranfield setting by augmenting it with the use of the GOMS [7] model (acronym for Goals, Operators, Methods, and Selections). He suggests a user model for IR where the search process is seen as a sequence of actions (e.g., typing; clicking; evaluating a summary; waiting for the results to load) with associated times and probabilities (e.g., whether the simulated user will click on a relevant summary). He used the model in a simulated study to demonstrate the impact of changes in the IR system interface (e.g., when the speed and accuracy of the summary evaluation is varied) on user performance (the number of relevant documents read within a given time frame). While his experiment was limited to single query situations, the approach can be extended to multiple query scenarios, e.g., for computing the costs of specific query reformulations.

Azzopardi [2] addressed the cost aspect by treating interactive IR as an *economical* problem and studied the trade-off between querying and browsing while maintaining a given level of normalized CG (NCG) [15] in sessions. His analysis focused on querying – scanning depth combinations for various formal retrieval methods that deliver a given level of NCG.

Our approach in the present paper differs from earlier studies. Our study is based on the simulation of multiple-query sessions generated with various query modification and scanning strategies in different searching environments.

In the next section we start by discussing session generation with costs, and present the research questions. In Section 3 we describe the research setting. In Section 4 we will run an experiment in a test collection based on scenarios and discuss the results. We close the paper by discussing the significance of our approach in the last section.

## 2. CONSTRUCTION OF SESSIONS

A *use case* is “a relatively informal description of system’s behavior and usage, intended to capture the functional requirements of the system by describing the interaction between the outside actors and the system, to reach the goal of the primary actor” [19]. We utilize simplified use cases, which we call scenarios, to present an alternative way to look at the effectiveness of IR approaches based on the user viewpoint. The next subsections will first explain the session generation formally, and then explain the specific query modification (QM) and scanning strategies utilized in the scenarios.

### 2.1 Session generation

For session simulation, we first formally generate all possible sessions under constraints. We will represent sessions as sequences of actions with costs. For example the tuple  $\langle (a_1, c_1), (a_2, c_2), \dots, (a_n, c_n) \rangle$  is a session of  $n$  actions and each pair  $(a_i, c_i)$  in the session

representation represents an action  $a_i$  and its cost  $c_i$ . The elementary action types are:

- Initial query, represented as  $(‘iq’, ic)$ , where ‘iq’ is the action label and  $ic (\in \mathbb{R})$  the cost in seconds.
- Query reformulation  $(‘q’, qc)$ , where ‘q’ is the action label and  $qc (\in \mathbb{R})$  the cost in seconds.
- Document snippet scan  $(‘s’, sc)$ , where ‘s’ is the action label and  $sc (\in \mathbb{R})$  the cost in seconds.
- Next page request  $(‘n’, nc)$ , where ‘n’ is the action label and  $nc (\in \mathbb{R})$  the cost in seconds.

The constraints are:

- MaxSLen, maximum session length in terms of elementary actions, here 50 actions.
- MaxSCost, maximum session cost (seconds), here 60, 90 or 120 seconds.
- A session always begins with an initial query.
- All queries (initial and reformulation) are followed by at least one snippet scan.
- The longest scan sequence we consider is a scan of 10 snippets (i.e. one typical result page).

In effect, the shortest possible session therefore is initial action  $IA = \langle (iq, ic), (s, sc) \rangle$ , consisting of an initial query followed by the scan of one snippet (with costs). To generate longer sessions, we define the set  $NA$  for the possible subsequent actions:

$$NA = \{ \langle (q, qc), (s, sc) \rangle, \langle (s, sc) \rangle, \langle (n, nc), (s, sc) \rangle \}$$

Note here that the next actions are tuples of one or two elementary actions; a scan may appear individually, while a reformulation / next page requires a scan to follow. Sessions are generated by concatenating next actions to the initial action. Concatenation of two tuples  $S_1 = \langle e_1, e_2, \dots, e_n \rangle$  and  $S_2 = \langle f_1, f_2, \dots, f_m \rangle$  is denoted by  $\langle S_1, S_2 \rangle = \langle e_1, e_2, \dots, e_n, f_1, f_2, \dots, f_m \rangle$ . This operation generalizes over a set of session tuples  $S_i$ , denoted as:

$$\times_{i=1 \dots n} S_i = \langle \dots \langle \langle S_1, S_2 \rangle, S_3 \rangle, \dots \rangle, S_n \rangle.$$

The cost of a session  $S$  is, informally, the sum of its action costs. More formally, we derive this cost by the function  $s-cost$  as follows:

$$s-cost(S) = \sum_{(a,c) \in S} c$$

[N.B. we extend the definition of the set membership operator from sets to tuple components in an obvious way.] For example, the cost of the session  $S1 = \langle (‘iq’, ic), (‘s’, sc), (‘q’, qc), (‘s’, sc) \rangle$  is  $s-cost(S1) = ic+sc+qc+sc$ .

The condition of maximum scan length of  $n$  in a session  $S$  is enforced by the Boolean predicate  $max-scan(S, n)$ . It yields ‘true’ for a given session  $S$  if  $S$  does not contain a subsequence of scan actions  $\langle (‘s’, sc)_1, (‘s’, sc)_2, \dots, (‘s’, sc)_n \rangle$ , otherwise ‘false’ (formal definition here omitted for brevity).

To generate sessions, we first generate all sessions up to the maximum number of actions MaxSLen. This session set is MLS:

$$MLS = \bigcup_{i=1 \dots MaxSLen} \{ \langle IA, \times_{j=1 \dots i} ac_j \rangle \mid ac_j \in NA \}$$

We then select the subset of sessions fulfilling the time constraint MaxSCost and the scan length constraint as follows. All sessions in MLS with maximal cost MaxSCost (or less) form the set MCS:

$$MCS = \{ S \in MLS \mid s-cost(S) \leq MaxSCost \wedge max-scan(S, 11) \}$$

Note that this approach does not define the query contents or modifications in sessions. However, it keeps them within constraints and guarantees that the last action is a document snippet scan. In our experiments, we excluded the next page action from NA due to the max scan length constraint of 10. The next two sub-sections explain and justify the query modification and scanning strategies used in the experiment.

## 2.2 Query Modification Strategies

We will simulate interactive search sessions as querying-scanning iterations having a goal, a procedure to reach the goal, and constraints regarding the procedure. We define the goal in terms of maximizing CG during the session under the constraint on the overall session time available. The procedure is defined in terms of QM and scanning strategies.

The previous section did not define any particular QM strategies. We assume that a set of individual words  $\{w_1, w_2, w_3, w_4, w_5\}$  is available for each particular topic, and QM strategies determine how elements from this set are combined to form queries (either the initial query, or one of the subsequent queries). In other words, given a set of individual search words for the topic, the QM strategy defines how to form a sequence of queries.

Five QM strategies (S1 – S5) were used in the experiment. These prototypical strategies are based on term level changes which have grounding in the observed real life behavior and are justified by literature (see [1], [20], [33]):

- **S1:** an initial one-word query ( $w_1$ ) is followed by repeatedly varying the search word :  
 $Q_1: w_1 \rightarrow Q_2: w_2 \rightarrow Q_3: w_3 \rightarrow Q_4: w_4 \rightarrow Q_5: w_5$
- **S2:** an initial two-word query ( $w_1 w_2$ ) is followed by queries formed by repeatedly varying the second word :  
 $Q_1: w_1 w_2 \rightarrow Q_2: w_1 w_3 \rightarrow Q_3: w_1 w_4 \rightarrow Q_4: w_1 w_5$
- **S3:** an initial three-word query ( $w_1 w_2 w_3$ ) is followed by queries formed by repeatedly varying the third word :  
 $Q_1: w_1 w_2 w_3 \rightarrow Q_2: w_1 w_2 w_4 \rightarrow Q_3: w_1 w_2 w_5$
- **S4:** an initial one-word query ( $w_1$ ) is followed by adding one word to each subsequent query :  
 $Q_1: w_1 \rightarrow Q_2: w_1 w_2 \rightarrow Q_3: w_1 w_2 w_3 \rightarrow Q_4: w_1 w_2 w_3 w_4 \rightarrow \dots$
- **S5:** an initial two-word query ( $w_1 w_2$ ) is followed by adding one word to each subsequent query :  
 $Q_1: w_1 w_2 \rightarrow Q_2: w_1 w_2 w_3 \rightarrow Q_3: w_1 w_2 w_3 w_4 \rightarrow \dots$

This means that the sessions consist of at most 3 to 5 queries; this reflects real life behavior [22]. Generally speaking, constructing a query entails a cost due to the cognitive user load plus the edit costs. We will return to the cost factors in Section 2.4.

## 2.3 Scanning Strategies

The user may simply scan one or more documents after each query before formulating the next query candidate or ending the session. After a *single* query  $Q_i$  a sequence of one or more document snippets may be scanned:

$$Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow \dots$$

The cost of this session manifests as:

$$qc_1 + sc_{11} + sc_{12} + sc_{13} + \dots$$

When a *set* of queries is available for one topic, the user can scan varying numbers of document snippets after any particular query, leading to a vast number of possible querying-scanning *sessions*, e.g.,

$$Q_1 \rightarrow s_{11} \rightarrow Q_2 \rightarrow s_{21} \rightarrow Q_3 \rightarrow s_{31} \rightarrow \dots \text{ or}$$

$$Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow Q_2 \rightarrow s_{21} \rightarrow \dots \text{ or}$$

$$Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow Q_2 \rightarrow s_{21} \rightarrow s_{22} \rightarrow Q_3 \rightarrow s_{31} \rightarrow \dots \text{ etc.}$$

In real life a session typically continues until the user has found what he was looking for, at least partially, and/or when he runs out of time or queries. The scanning lengths may fluctuate for many reasons. In this paper we study the properties of optimal and less optimal interactive behaviors in sessions below the given overall time constraint. Therefore we produced *all* possible sessions as follows. For all five QM strategies we formed all possible combinations of scanning lengths exhaustively (from 1 to 10 documents) using a sequence of all possible queries available per topic (cf. equation MCS in Section 2.1). We focus on the top documents because only few top documents may be inspected by the user in real life [14], [23], and only these may matter for the user [1]. As we had 5 words for each topic, sessions had at most 5 queries, controlled by the QM strategy and time constraint. As the query words were ordered by quality (see 3.1), the query words were used in that particular order, not permuted.

## 2.4 Cost Factors

There is a cost involved with the subtasks of formulating the query and scanning. We assume that the *absolute cost* is partially determined by the scenario. Empirical studies show that it takes significantly more time to enter queries by using a small smart phone keypad than by using an ordinary keyboard [17]. To study the significance of subtask costs under overall session cost constraint we define two scenarios, i.e., a Desktop PC scenario (PC) and a Smart phone scenario (SP). These scenarios have different subtask costs. This is justified because the properties of the devices partially determine the subtask costs [17].

Obviously, also forming queries under different QM strategies S1 – S5 have very different *relative costs*. All queries in strategies S1, S2 and S3 have a fixed query length in sessions (one, two or three words, correspondingly) while in strategies S4 and S5 the queries grow longer. In real life the typing speed is affected by, e.g., the experience and knowledge of the person, the size of the keyboard, the layout of the keyboard (e.g., nine-key multi-tap vs. qwerty keyboard) [17], [18], and whether predictive text feed is available and used. We used literature to derive the cost values in scenarios PC and SP regarding the initial query cost and the subsequent query cost (Table 1). The query costs in S1 – S5 in the Desktop PC case are based on the typing costs of 3.0 seconds per word. The corresponding Smart Phone costs are based on [17]. The authors performed a large-scale log analysis of cell phone usage and observed that an average smart phone query length was 2.56 words and the average query-entry time was 39.8 seconds (average typing cost of 15.5 seconds per word). We assume in our simulations that the cost of *adding* one word to a query (that is, S4 and S5) or *replacing* one word at the end of the previous query (that is, S1, S2, S3) is a constant, i.e., either 3.0 or 15.5 seconds depending on the scenario.

**Table 1. Average subtask costs (in seconds) of five QM strategies (S1-S5) for two scenarios: (i) initial query cost, (ii) subsequent query cost, and (iii) the cost of scanning one document snippet**

Scenario 1: Desktop PC					
QM strategy	S1	S2	S3	S4	S5
Initial query	3.0	6.0	9.0	3.0	6.0
Subsequent query	3.0	3.0	3.0	3.0	3.0
Snip. scanning cost	3.0	3.0	3.0	3.0	3.0
Scenario 2: Smart Phone					
QM Strategy	S1	S2	S3	S4	S5
Initial query	15.5	31.0	46.5	15.5	31.0
Subsequent query	15.5	15.5	15.5	15.5	15.5
Snip. scanning cost	3.0	3.0	3.0	3.0	3.0

To check whether these costs are reasonable we also performed a small-scale experiment where four test persons typed the initial and subsequent queries according to strategies S1-S5 using two types of interfaces (Desktop PC and Smart Phone) for three test topics. The experiment corroborated that the query time estimates were reasonable.

The document snippet scanning costs in real life are affected by the motor and perceptual costs plus the cognitive load related to the task. In this study we assume that the document snippet scanning cost is constant in both scenarios and across the searching strategies S1 – S5 (see Table 1). In the SP case we defined a scanning cost of three seconds per snippet. We justify this by an observation by Kamvar and Baluja [17] that the average cell phone user used 30 seconds to scan the search results before selecting one, after receiving 10 search results. For the snippet scanning costs in the Desktop PC case we decided to use the same value. Obviously, our methodology is well-suited to experiment with different costs. The overall cost constraint of a session was defined as 60, 90, or 120 seconds. In the simulations all subtasks (querying and scanning) had to be performed within this time constraint. We excluded the eventual thinking time in producing query words.

## 2.5 Research questions

We set forth the following research questions:

1. How effective are the five QM strategies (S1 to S5) in terms of CG when we compare the Desktop PC and the Smart Phone scenarios under overall time constraint?
2. What are the characteristics of the best and the worst sessions achieved in terms of average scan length, and average number of queries?
3. How stable are the observed trends when the overall time constraint changes? Can we recommend QM strategies based on the scenario - what to do, and what not to do, assuming a specific time constraint?
4. What is proper evaluation methodology when time is part of the evaluation setting?

## 3. RESEARCH SETTING

### 3.1 Test Collection and Search Engine

We used a subset of the TREC 7-8 document collection with 41 topics for the experiment. The documents have graded relevance assessments on a four-point scale with respect to the topics. [27]

The present authors obtained query words for session generation for the test topics from [20] where the authors used real test persons to suggest keywords of various lengths for queries on the 41 topics. The test persons were asked to directly propose good search words from topic descriptions (descriptions and narratives) in a structured way. Among others, they produced query versions of various lengths: (i) one word, (ii) two words, and (iii) three or more words. These were collected per topic as ordered word lists of 5 words for each topic. During the query formulation experiment the test persons did not interact with a real retrieval system. While this may have affected negatively the quality of queries, Keskustalo and colleagues [20] suggest that the test persons were able to construct the query words in a descending order of effectiveness.

Retrieval system *LeMUR* with language modeling and two-stage smoothing options was used in the experiment.

### 3.2 Session Data

For each topic we utilized a minimum of 1 query and a maximum of 5 queries in each session. A minimum of 1 document snippet and a maximum of 10 document snippets were scanned per query.

In Table 2, the number of possible scanning paths is given for consecutive queries. If the session comprises at most 2 queries, first there are 10 possible paths after the first query, and for every path there are 10 possible paths after the second query. So the combinations of these at most two queries sum up to  $10+10*10=110$  possible paths. In our experiment design, users can pose up to 5 queries depending on session strategy; this presents altogether 111,110 possible paths, which are taken into consideration.

**Table 2. Number of possible sessions per number of queries, when at most 10 documents can be scanned after each query**

Queries	1	2	3	4	5	$\Sigma$
Possible sessions	10	100	1000	10,000	100,000	111,110

We ran all 41 topics \* 5 QM strategies \*  $Q$  queries,  $Q \in \{3, 4, 5\}$  depending on the strategy, and collected their results. Then we generated all 111K possible sessions from the query results, pruned the ones exceeding the time constraint in each scenario, and by using the recall base (qrels), evaluated the CG of the scanned snippets for each session. For example, for the session  $Q_1 \rightarrow s_{11} \rightarrow s_{12} \rightarrow s_{13} \rightarrow Q_2 \rightarrow s_{21} \rightarrow s_{22} \rightarrow Q_3 \rightarrow s_{31}$ , the CG is calculated on the basis of the snippet sequence  $s_{11}, s_{12}, s_{13}, s_{21}, s_{22}, s_{31}$ . Altogether about 45 million sessions (41 topics \* 5 QM strategies \* 111,110 possible scanning sessions \* 2 scenarios) were evaluated. As the collection has graded relevance assessments, CG was incremented by 3 points for the highly relevant documents, 2 points for the fairly relevant documents and 1 point for the marginal ones. Whenever a duplicate was retrieved by a subsequent query in a session, its gain was nullified. Finally, we ranked all sessions within a topic and a strategy by their CG scores. In this data set per topic, strategy and time constraint, each session is represented by its tuple of actions (see 2.1) and its gain.



### 3.3 Data Analysis

The action tuples allow the analysis of the number of queries and the length of each scan in a session. The ranked order of sessions allows identification of the best and the worst session across topics, strategy, scenario, and time constraint. We analyze the sets of 10 best sessions, and 10 worst sessions per topic as averages instead the single best or worst session. This approach smoothes minor random variations in human behavior and thus the set of top (bottom) 10 sessions provide more reliable measurements compared to the single best/worst session when we explore their properties under varying conditions. Since the present study does not aim to prove one retrieval method better than another, we report the findings without tests on significance of statistical differences.

## 4. EXPERIMENTS

### 4.1 Results for the 60 Seconds Time Frame

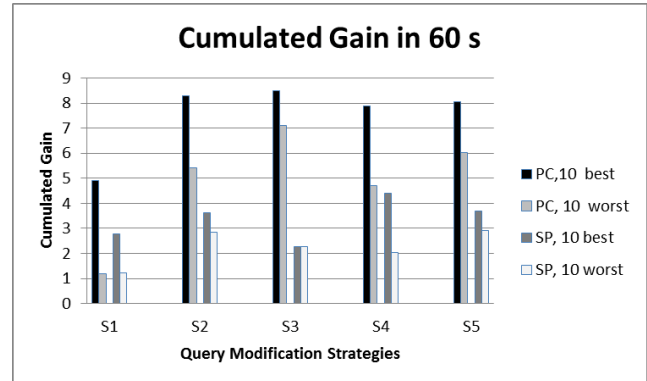
First we discuss the CG results under the two scenarios, PC and SP. We present the best case and worst case results regarding all querying-scanning sessions based on the five QM strategies: S1 (*sequence of individual words*); S2 (*two-words; last word varied*); S3 (*three-words; last word varied*); S4 (*incremental extension starting from one word*); and S5 (*incremental extension starting from two words*). Table 3 gives the averaged CG values, the number of queries and scans per query for 10 best and 10 worst cases for every QM strategy for the 60 second time constraint, which are utilized in the following figures in this section.

**Table 3. Averaged CG, number of Queries (#q) and Scans per Query (s/q) for scenarios PC and SP, for 5 strategies for the 10 best (b) and 10 worst (w) sessions, time constraint 60 seconds**

Time (60 s)	Environment	best/worst	Query Modification Strategies				
			S1	S2	S3	S4	S5
avg. CG	PC	b	4.9	8.3	8.5	7.9	8.1
		w	1.2	5.4	7.1	4.7	6.0
	SP	b	2.8	3.6	2.3	4.4	3.7
		w	1.2	2.8	2.3	2.0	2.9
avg. #q	PC	b	2.7	2.6	2.5	4.2	3.0
		w	5.0	4.0	3.0	5.0	4.0
	SP	b	1.9	1.5	1.0	2.0	1.5
		w	2.7	1.7	1.0	2.6	1.7
avg. s/q	PC	b	6.4	6.3	6.2	3.8	5.3
		w	3.0	3.8	5.0	3.0	3.8
	SP	b	4.7	3.6	2.5	4.0	3.6
		w	1.6	2.5	2.5	1.7	2.5

Table 4 and Table 5 are equivalent to Table 3 but for the time constraints 90 and 120 seconds, respectively. Figure 1 shows the CG of the best (worst) sessions for each strategy in both scenarios under the overall cost constraint of 60 seconds. Note that all sessions require 60 seconds or less if no further action fits in (the absolutely worst imaginable session without any time requirement, in terms of the CG, would naturally consist of the initial action (IA) only). In other words, regarding the worst results, we report CG for the worst possible 60 second performance.

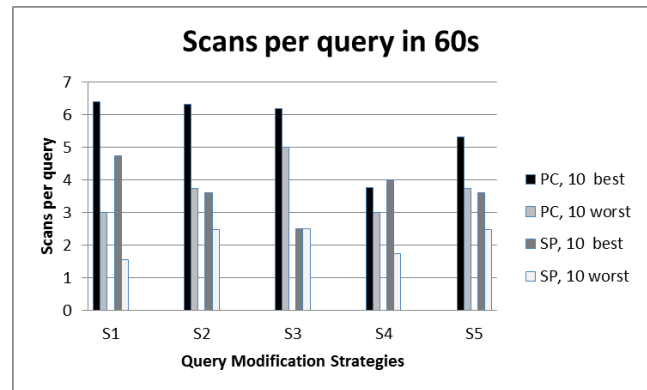
When the best sessions of the PC and SP cases are compared in Figure 1, the PC case performs at a considerably higher level (average CG is above 8 in three strategies) than the SP case (average CG is below 5 in all strategies).



**Fig 1. Cumulated Gain under cost constraint of 60 seconds.**

Second, when the best and the worst cases are compared within the scenarios, not surprisingly, the best case results are typically clearly better than the worst case results except in SP case for S3. In the latter case both the best and the worst session may not contain more than one query because of high query entry cost.

Third, among the best cases for PC the strategies S2 and S3 are almost equally good. For the SP case, the strategy S2 (varying the second word), S4 (extending from one word), and S5 (extending from two words) lead to much higher gain than S1 and S3. An interesting trade-off in the SP scenario can be observed when the scanning length is considered. In the best case the gain reached increases from S1 to S2. However, the average scanning length decreases (Fig. 2). In other words, a better result is achieved using the longer queries although a smaller number of documents are scanned on the average; the ranking is simply better.



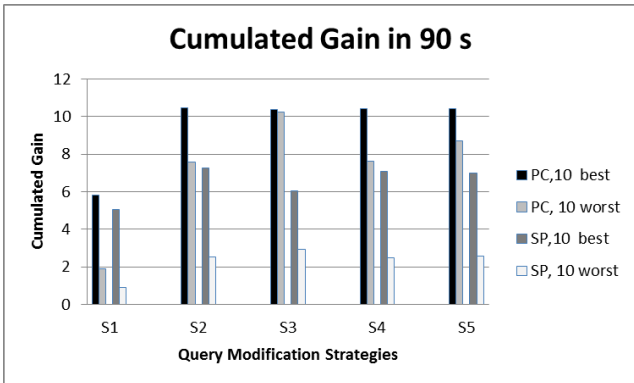
**Fig 2. Average number of scanned document snippets per query under cost constraint of 60 seconds.**

**Table 4. Averaged CG, number of Queries (#q) and Scans per Query (s/q) for scenarios PC and SP, for 5 strategies for the 10 best (b) and 10 worst (w) sessions, time constraint 90 seconds**

Time (90 s)	Environment		Query Modification Strategies				
	best/worst		S1	S2	S3	S4	S5
avg. CG	PC	b	5.8	10.5	10.4	10.4	10.5
		w	1.9	7.6	10.3	7.6	8.7
	SP	b	5.0	7.3	6.0	7.1	7.0
		w	0.9	2.5	3.0	2.5	2.6
avg. #q	PC	b	3.8	3.5	3.0	5.0	4.0
		w	5.0	4.0	3.0	5.0	4.0
	SP	b	2.0	2.0	1.9	2.5	2.0
		w	4.1	3.4	2.6	4.1	3.4
avg. s/q	PC	b	6.9	7.3	8.3	5.0	6.3
		w	5.0	6.3	8.3	5.0	6.3
	SP	b	8.8	6.8	4.7	6.3	6.8
		w	1.6	1.4	1.7	1.4	1.4

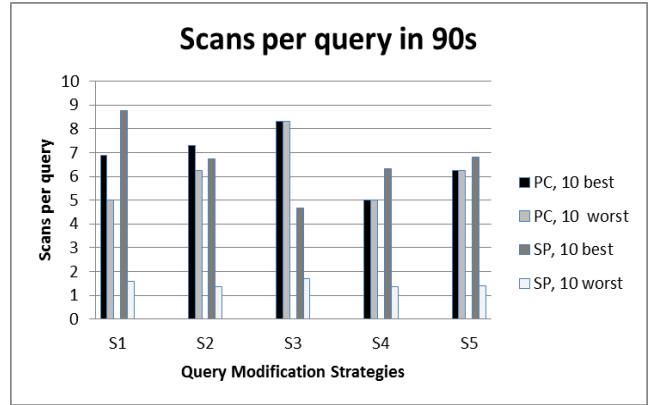
#### 4.2 Results for the 90 Seconds Time Frame

Figure 3 shows the CG results when the sessions take 90 seconds. In this case, the observations comply with the 60 second case. Difference between S3's best and worst CG values is closing in the PC scenario; this is because of lacking further scanning options, there is now enough time to scan almost all 10 documents for each query. S3 strategy has a maximum of 3 queries to execute before the 5 keywords run out. This in turn confines the possible scanning space. It is also conspicuous that the difference between best and worst CG values in SP case is much larger than in PC case.



**Fig 3. Cumulated Gain under cost constraint of 90 seconds.**

When scanning in the best sessions of the PC and SP cases is compared (Fig. 4), we notice that even though the scans per query values for SP case are higher than or similar to the PC case, the CG values are always poorer (Fig. 3). This is due to the smaller number of posed queries in SP case than in PC case. This follows from the trade-off between query vs. scan costs.



**Fig 4. Average number of scanned document snippets per query under cost constraint of 90 seconds.**

Interestingly, the difference between the best and the worst sessions both in terms of gain and average scan length remains great in SP case, but fades away in PC case. In the latter, 90 seconds allows the searcher to launch almost all queries and scan the best results in all cases. When the results are compared between different strategies, the strategy S4 with on average 5 scans in PC case and approximately 6 scans in SP case (Fig. 4) produce similar CG values as the other QM strategies (Fig. 3). Again, larger queries yield better rankings. On the other hand, S3 in SP case has less than 5 scans per query, and still achieves slightly better CG results than S1 strategy.

**Table 5. Averaged CG, number of Queries (#q) and Scans per Query (s/q) for scenarios PC and SP, for 5 strategies for the 10 best (b) and 10 worst (w) sessions, time constraint 120 seconds**

Time (120 s)	Environment		Query Modification Strategies				
	best/worst		S1	S2	S3	S4	S5
avg. CG	PC	b	6.4	11.1	11.4	11.7	11.5
		w	3.4	10.5	11.4	10.0	10.9
	SP	b	5.6	9.1	9.2	9.1	8.9
		w	1.1	5.1	6.7	4.5	5.7
avg. #q	PC	b	4.8	4.0	3.0	5.0	4.0
		w	5.0	4.0	3.0	5.0	4.0
	SP	b	3.0	2.9	2.0	3.0	2.9
		w	5.0	4.0	3.0	5.0	4.0
avg. s/q	PC	b	7.3	8.8	10.0	7.0	8.8
		w	7.0	8.8	10.0	7.0	8.8
	SP	b	7.6	6.6	8.8	7.6	6.5
		w	2.8	3.5	4.7	2.8	3.5

#### 4.3 Results for the 120 Seconds Time Frame

Figure 5 shows the CG values under the cost constraint of 120 seconds. In the PC case, the gaps between the best and worst CG values are diminishing. This can be explained so that every strategy except S1 and S4 has enough time to pose all the queries and employ much scanning. According to the experiment design, worst cases must also use up the allocated time, and this results in that there is enough time to launch all queries and scan the results. When the best sessions of the PC and SP cases are compared, we notice that there are no large differences. Again, in Figure 6 we can see as many scans per query (S/Q) for S1 and S4 in the SP case as in the

PC case for best sessions. Besides all the strategies for PC case show the same S/Q for 10 best and 10 worst sessions. Although in SP case S/Q values diverge from each other, Figure 6 exhibits similar patterns as Figure 4. From Figure 5 one can conclude that, if there is enough time for searching, one should use at least two word queries for good results. If the queries are of lower quality like S1, then scanning matters. In short, the more you scan, the more you get.

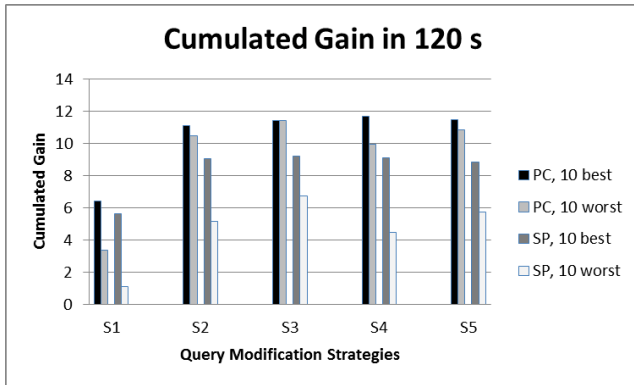


Fig 5. Cumulated Gain under cost constraint of 120 seconds.

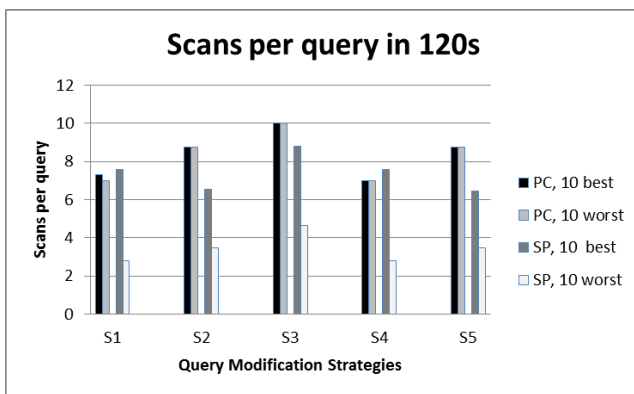


Fig 6. Average number of scanned document snippets per query under cost constraint of 120 seconds.

## 5. DISCUSSION

We had three empirical and one methodological research question. The three empirical ones were about effectiveness of different QM strategies under time constraints, characteristics of the best and the worst QM sessions, and the stability of the observed trends. The methodological one was about proper evaluation of sessions under time constraints. We will consider each of the questions below.

**Strategy effectiveness.** Given a stringent time frame in the PC scenario, the user cannot use the entire vocabulary (all queries) and perform exhaustive scanning for all queries. Short queries (strategy S1) are clearly inferior regarding session effectiveness. It seems reasonable to invest on two to three word queries (S2, S3) because the evidence thereby added for ranking significantly improves the quality of the results. This can also be seen in strategies S4 and S5, when they have enough time to advance beyond the first query. When more time is allocated to searching, the weaker strategies catch up because there is more time for scanning the results and the weaker ranking effectiveness is not that critical.

In the SP scenario the rules of the game change a bit. In a stringent time frame there is no time for tedious query input, and one must compromise toward short scanning of weaker quality rankings. The more effective strategies cannot be applied at all due to high query input cost. Again, when more time is allocated, weaker strategies catch up. In the longest sessions of S2-S5, the gap between the best vs. worst sessions begins to close.

**Session characteristics.** In the PC scenario, under stringent time constraints, the best sessions involved less queries and longer scans than the worst sessions (Table 3). However, as the time allocation grows, the differences disappear. Between the best strategies in the PC case, both the number of queries and the average scan lengths increase as time allocation grows (Tables 3-5). Correspondingly, in the worst sessions, the number of queries does not change as time grows, but the scan lengths grow. This is because the worst sessions consume all possible queries even under the shortest time frame. Similarity with best sessions grows.

In the SP scenario, under stringent time constraints, the best sessions also involved less queries and longer scans than the worst sessions (Table 3). As the time allocation grows, the differences remain, probably due to shortage of time even in the longer sessions. Between the best strategies in the SP case, both the number of queries and the average scan lengths increase as time allocation grows, the latter dramatically between 60 and 90 seconds (Tables 3-4). Correspondingly, in the worst sessions, the number of queries grows along time, but the scan lengths remain low. The worst behavior here means investing the effort in query input. Also here there were interesting differences in scan lengths between queries in sessions.

All in all, if time allows, two to three first query words that one identifies, followed by a longer scan, seem to provide reasonable performance, no matter what the strategy among S2-S5 is.

**Effect of time.** With limited time allowance, it seems important to make a good compromise between providing evidence for ranking (longer queries) and scanning the search results. The compromise depends on the overall cost levels related to the stringency of the time frame and on the relations between cost types. This depends on the searching device. Expensive input favors scanning at length, cheap input favors better queries. The more time is available the less it matters how one searches – there will be time to identify the relevant documents.

**Evaluation methodology.** Typical IR evaluation metrics are based on the quality of ranking alone. In session-based evaluation they must be applied with great care because they may be insufficient or even misleading. They may be partially insensitive to the user's experience and observed costs and benefits. This is particularly critical, when user's costs (time expenditure) are taken into account and the metric employs normalization, i.e. scaling the measurements to a predefined range such as [0, 1]. For example, the popular NDCG metric [15] and its non-discounted counterpart NCG should be avoided in any comparisons between searching environments, and between strategies within a given searching environment *when* input costs are taken into account. This is because the ideal gain vector used for normalization is read to vastly different lengths between strategies or environments. For example, consider Figure 7, which plots NCG over time for strategy S2 in the two scenarios.

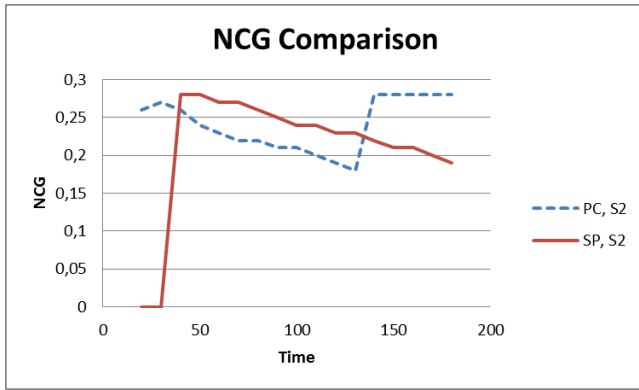


Fig 7. NCG vs. time comparison of PC and SP for S2 (41 Topics).

Due to normalization (division by the ideal cumulated gain vector) the SP scenario seems to have better performance in the time frame from 40 to 135 seconds. This is due to (a) ranking being somewhat effective, and (b) the number of documents seen in each session: in the PC case the user sees 15 to 35 documents, but in the SP case only 5 to 20 documents in the indicated time frame. Figure 8 plots CG with the corresponding data and makes the difference clear.

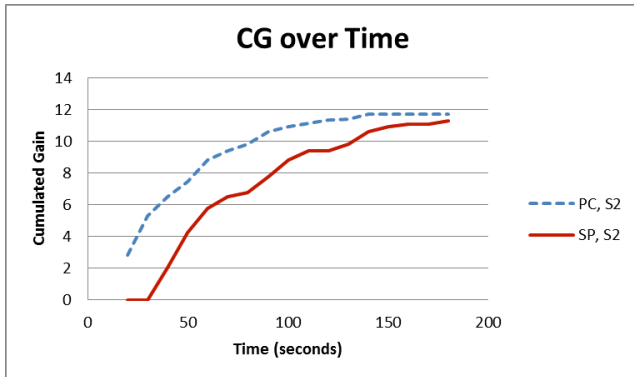


Fig 8. CG over time for S2 in scenarios PC and SP (41 Topics).

Similar pitfalls also plague the most classic metric, MAP. Consider the following two rankings observed for a given topic in two scenarios and/or strategies under the same time constraint (say, one minute; queries omitted and binary relevance for simplicity):

r1: 0 0 0 0 0 0 0 1 1

r2: 1 0 0 0

Further, assume that there are three relevant documents for the topic. The MAP for the ranking r1 is  $(1/9 + 2/10 + 0)/3 = 0.103$  and for r2  $(1 + 0 + 0)/3 = 0.333$ . Arguably, r2 is the better ranking, but if both require one minute, what is the user's opinion? The first session collected twice as many relevant documents.

Even within the un-normalized metric, such as CG, incorporating time in session-based evaluation has profound effects. Consider Figures 9 and 10. The former gives traditional cumulated gain over ranks for strategies S1 and S3 for the 41 topics. The latter gives CG over time in the two scenarios.

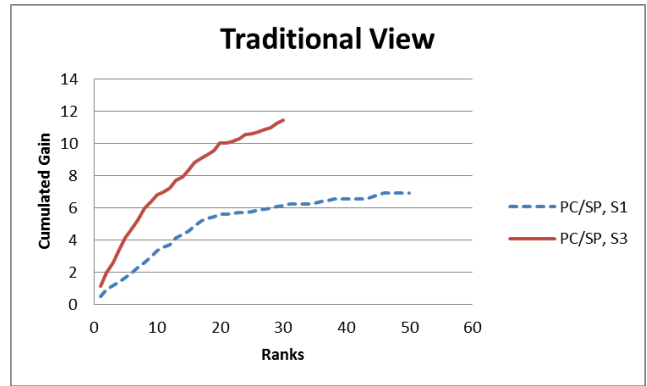


Fig 9. Traditional View, CGs over ranks for 41 topics, scenarios PC and SP for strategies S1 (allowing 5 queries) and S3 (allowing only 3 queries).

In Figure 9, both scenarios PC and SP have the same observed effectiveness, because the evaluation focuses on the gain (CG) over the result ranks, no matter how long it takes to retrieve the documents. The two strategies S1 and S3 differ in effectiveness, S3 providing far better effectiveness than S1. However, when time is taken into account (Fig. 10), the scenarios and strategies differ greatly from each other. Up to 60 seconds, S3 in the SP case is the worst strategy and this is entirely due to the high input cost of the long query. With enough time (180 sec.), S3 in SP catches up S3 in PC case. Also, PC and SP do not much differ for S1 due to the relatively low input cost and weak result quality. Comparing Figures 9 and 10, it is easy to see that time drives interaction and profoundly affects both user experience and effectiveness in sessions in different scenarios.

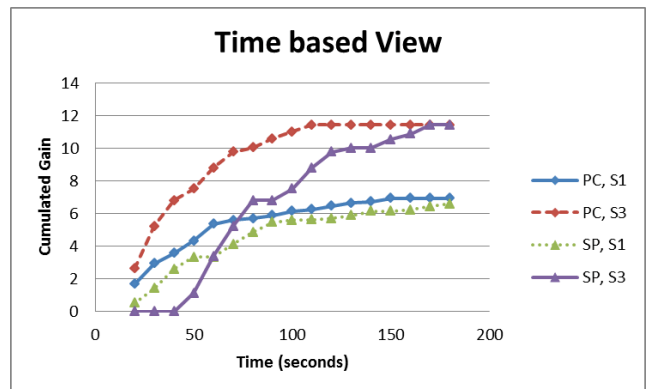


Fig 10. Time based View, CGs over time for 41 topics, scenarios PC and SP for strategies S1 and S3.

**Limitations.** In our study we did not take into account the time, which users spend for pondering about possible query words. One might argue that the more words one needs to identify, the harder (and slower) per word it comes. However, the thinking time is the same between sessions using the same number of words. In addition, this could be taken into account by revising subsequent query costs (Table 1). We have chosen to short-cut here in order to avoid too much complexity at this stage. Furthermore, we do not consider the time users spend in examining documents. This may depend on the device used. This can be seen as an artificial limitation. Tackling it would, however, complicate analysis, and this is therefore left for later study. We did not simulate user's learning during a session. Admittedly, learning from snippets and seen documents take place.

This is not impossible to simulate but some challenges remain to be solved.

We employed in the evaluation relatively limited query vocabularies, simple bag-of-word queries, and relatively short time frames. The query vocabularies and structure are justified by query length statistics in many search environments [14], [29], and the time frames by our simulation capabilities. However, the time frames are for *effective search time in sessions*, excluding thinking and document examination time. While the query vocabularies are short, they are human-generated for this collection, and therefore more realistic than words mined, e.g., from known relevant documents (in qrels).

We did not cover all the imaginable complex sessions. However we employed idealized and literature-based sessions, which shed the light on the peculiar evaluation problems beyond the traditional rank-based evaluation. This is a step forward while we are not suggesting that anyone follows a single strategy consistently in real life.

Our initial results are promising. First, the scenario, and to a large extent the device itself, dictate what kind of interactive behavior can be successful. Because real users do have limited resources and they use various devices having different properties, our methodology has unquestionable user relevance and potential pragmatic value for the industry. Measuring the effectiveness of systems from the pragmatic point of view may increase the validity of the results achieved. This may lead to greater user satisfaction. Secondly, our experimental results suggest that strict time constraints determine some session strategies as the best strategies as they maximize CG. The strengths of our approach are:

- The QM strategies S1-S5 have an empirical real life grounding
- The query vocabularies were generated by real test persons, and only thereafter used in automatic simulation
- We were able to evaluate over 20M sessions in each scenario; this is clearly intractable both physically, intellectually and economically with human test persons.

We have only taken the first steps. In future, we will study the dimensions of variation related to users, systems, information sources and sessions to construct more fine-grained scenarios explicating hypotheses about user goals, learning, and behaviors to validate evaluation measures used. [19]

## 6. CONCLUSIONS

In this study, we have shown the necessity of a pragmatic evaluation approach based on scenarios with explicit subtask costs under an overall time constraint. Effectiveness of various query modification and scanning strategies for two scenarios, namely, PC and SP is analyzed. Furthermore, the characteristics of the best and the worst interactive search sessions are examined. Expensive input favors scanning at length, cheap input favors better queries. The more time is available the less it matters how one searches – there will be time to identify the relevant documents. We have shown that the effort required by searching devices and the overall search time allocation drive interaction and profoundly affect both user experience and effectiveness in sessions in different scenarios. Moreover, we have also pointed out the inapt use of all normalized rank-based measures. Thus, we hope we could instigate new evaluation metrics for time-based comparisons.

## 7. ACKNOWLEDGMENT

This research was funded by Academy of Finland grant number 133021.

## 8. REFERENCES

- [1] Azzopardi, L. 2007. Position Paper: Towards Evaluating the User Experience of Interactive Information Access Systems. In *SIGIR'07 Web Information-Seeking and Interaction Workshop*, 5 p.
- [2] Azzopardi, L. 2011. The economics of interactive information retrieval. In *Proceedings of the 34<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 15-24.
- [3] Bates, M. J. 1979. Information search tactics. *Journal of the American Society for Information Science*, 30(4), 205-214.
- [4] Bates, M. J. 1989. The Design of Browsing and Berrypicking Techniques for the Online Search Interface. *Online Review*, 13(5), 407-424.
- [5] Beaulieu, M. 2000. Interaction in Information Searching and Retrieval. *Journal of Documentation*, 56(4), 431-439.
- [6] Belkin, N. L. 1980. Anomalous States of Knowledge as a Basis for Information Retrieval. *Canadian Journal of Information and Library Science*, 5, 133-143.
- [7] Card, S. K., Moran, T. P., and Newell, A. 1983. *The Psychology of Human-Computer Interaction*. L. Erlbaum Assoc. Inc., Hillsdale, NJ, USA.
- [8] Cleverdon, C.W., Mills, L., and Keen, M. 1966. Factors determining the performance of indexing systems, vol. 1-design. In *Aslib Cranfield Research Project*, Cranfield.
- [9] Dunlop, M. D. 1997. "Time Relevance and Interaction Modeling for Information Retrieval". In *Proceedings of the 20<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, 206-213.
- [10] Fidel, R. 1985. Moves in online searching. *Online Review*, 9 (1), 62-74.
- [11] Hearst, M. A. 2011. "Natural" Search User Interfaces. *Communications of the ACM*, vol. 54, 60-67.
- [12] Hersh, W., Turpin, A., Price, S., Chan, B., Kraemer, D., Sacherek, L., and Olson, D. 2000. Do Batch and user Evaluations Give the Same Results? In *Proceedings of the 23<sup>rd</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 17-24.
- [13] Ingwersen, P. and Järvelin, K. 2005. *The Turn: Integration of Information Seeking and Retrieval in Context*. Heidelberg, Springer.
- [14] Jansen, M. B. M., Spink, A., and Saracevic, T. 2000. Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *Information Processing & Management*, 36(2), 207-227.
- [15] Järvelin, K. and Kekäläinen, J. 2002. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, 20(4), 422-446.
- [16] Järvelin, K. and Kekäläinen, J. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23<sup>rd</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Athens, Greece, 41-48.
- [17] Kamvar, M. and Baluja, S. 2007. Deciphering Trends in Mobile Search. *Computer*, 40(8), 58-62.
- [18] Karat, C-M., Halverson, C., Horn, D., and Karat, J. 1999. Patterns of entry and correction in large vocabulary continuous

- speech recognition systems. In *ACM Conference on Human Factors in Computing Systems*, 568-575.
- [19] Karlgren, J., Järvelin, A., Eriksson, G., and Hansen, P. 2011. Use cases as a component of information access evaluation. In *DESIRE'11 workshop*, October 28, 2011, Glasgow, Scotland, UK.
- [20] Keskustalo, H., Järvelin, K., Pirkola, A., Sharma, T. and Lykke, M. 2009. Test Collection-Based IR Evaluation Needs Extension Toward Sessions – A Case of Extremely Short Queries. In *Proceedings of the 5<sup>th</sup> Asia Information Retrieval Symposium (AIRS'09)*, 63-74.
- [21] Kuhlthau, C. C. 1991. Inside the Search Process. *Journal of the American Society for Information Science*, 42(5), 361-371.
- [22] Price, S.L., Nielsen, M.L., Delcambre, L.M.L., and Vedsted, P. 2007. Semantic Components Enhance Retrieval of Domain-specific Documents. In *Proceedings of the 16<sup>th</sup> ACM CIKM*, 429-438.
- [23] Ruthven, I. 2008. Interactive Information Retrieval. In *Annual Review of Information Science and Technology*, vol. 42, 2008. 43-91.
- [24] Salton, G. 1970. Evaluation Problems in Interactive Information Retrieval. *Information Storage and Retrieval*, 6, 29-44.
- [25] Smith, C. L. and Kantor, P. B. 2008. User Adaptation: Good Results from Poor Systems. In *Proceedings of the 31<sup>st</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 147-154.
- [26] Smucker, M. D. 2009. Towards Timed Predictions of Human Performance for Interactive Information Retrieval Evaluation. In *Third Workshop on Human-Computer Interaction and Information Retrieval (HCIR'09)*, October 23, 2009, Washington DC, USA.
- [27] Sormunen, E. 2002. Liberal Relevance Criteria of TREC – Counting on Negligible Documents? In *Proceedings of the 25<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, 324-330.
- [28] Spink, A. 1997. Study of Interactive Feedback during Mediated Information Retrieval. *Journal of the American Society for Information Science*, 48(5), 382-394.
- [29] Stenmark, D. 2008. Identifying Clusters of User Behavior in Intranet Search Engine Log Files. *Journal of the American Society for Information Science*, 59(14), 2232-2243.
- [30] Su, L.T. 1992. Evaluations Measures for Interactive Information Retrieval. *Information Processing & Management* 28(4), 503-516.
- [31] Turpin, A. and Hersh, W. 2001. Why Batch and User Evaluations Do Not Give the Same Results. In *Proceedings of the 24<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 225-231.
- [32] Turpin, A. and Scholer, F. 2006. User Performance versus Precision Measures for Simple Search Tasks. In *Proceedings of the 29<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 11-18.
- [33] Vakkari, P. 2000. Cognition and changes of search terms and tactics during task performance. In *Proceedings of RIAO 2000 Conference*, Paris: C.I.D., 894-907.

# Modeling Behavioral Factors in Interactive Information Retrieval

Feza Baskaya, Heikki Keskustalo, Kalervo Järvelin

School of Information Sciences

University of Tampere

Finland

{Feza.Baskaya, Heikki.Keskustalo, Kalervo.Jarvelin}@uta.fi

## ABSTRACT

In real-life, information retrieval consists of sessions of one or more query iterations. Each iteration has several subtasks like query formulation, result scanning, document link clicking, document reading and judgment, and stopping. Each of the subtasks has behavioral factors associated with them. These factors include search goals and cost constraints, query formulation strategies, scanning and stopping strategies, and relevance assessment behavior. Traditional IR evaluation focuses on retrieval and result presentation methods, and interaction within a single-query session. In the present study we aim at assessing the effects of the behavioral factors on retrieval effectiveness. Our research questions include how effective is human behavior employing search strategies compared to various baselines under various search goals and time constraints. We examine both ideal as well as fallible human behavior and wish to identify robust behaviors, if any. Methodologically, we use extensive simulation of human behavior in a test collection. Our findings include that (a) human behavior using multi-query sessions may exceed in effectiveness comparable single-query sessions, (b) the same empirically observed behavioral patterns are reasonably effective under various search goals and constraints, but (c) remain on average clearly below the best possible ones. Moreover, there is no behavioral pattern for sessions that would be even close to winning in most cases; the information need (or topic) in relation to the test collection is a determining factor.

## Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Search process

## Keywords

Session-based evaluation, IR interaction, behavioral factors, frustration, simulation, multi-query scanning models

## 1. INTRODUCTION

In real life, information retrieval (IR) takes place in sessions. When users interact with a search system, they formulate queries iteratively. User interaction in search sessions can be divided into subtasks like query formulation, result scanning, document link clicking,

document reading and judgment, and stopping. Moreover, each subtask is affected by associated behavioral factors. Such behavioral factors include search goals and cost constraints, query formulation strategies, scanning and stopping strategies, and relevance assessment behavior, which are the focus of the present paper. They are discussed in the literature of user-oriented IR (e.g. [10] [11]). User-oriented experiments indicate that such factors affect IR interaction, but their effects and interactions are both challenging and expensive to study (e.g., [14]). Still, many other factors, which are not the focus of the present study, affect real life IR: varying situation and task perception, searcher's knowledge on work and search tasks, and searcher's search vocabularies.

One approach to study user interaction is based on simulation. Session-based simulation, which extends single query simulations, is not a new approach in IR. The ACM SIGIR 2010 hosted a workshop on the simulation of interaction in IR [2]. Harman [9] simulated the effectiveness of relevance feedback in a test collection already in 1992 using a wide range of parameters including the method of term selection; the number of expansion terms; and the effectiveness of multiple iterations of relevance feedback. Others have more recently compared the effectiveness of short sessions with single long queries [12], and analyzed the trade-offs between querying and scanning in sessions [1], the effects of human fallibility in relevance feedback [3], and simulated the variance in user behavior related to scanning profiles [6]. In general, the strengths of modeling behavioral factors in IR interaction include: control over experimental parameters, unlimited supply of "test subjects" with no fatigue, low cost, no (non-programmed) learning effects, and repeatability of experiments. The limitations include the lack of full-fledged human subjects, which may lead to unrealistic and biased designs and findings. [2]

A specific aspect that has received attention in recent interactive IR (IIR) studies has been searcher's effort (or cost / time). Searcher's effort affects retrieval effectiveness and satisfaction. Azzopardi [1] addressed the cost aspect by treating interactive IR as an *economic* problem and studied the trade-off between querying and browsing. Smucker and Clarke [16] focused on single query sessions but studied searcher's effort in examining and assessing documents of various lengths. Baskaya and colleagues [4] focused on the effects of searching interfaces on searcher's effort and optimal behavior. Their study was limited to ideal behavior and confined to querying and scanning actions. In the present paper we also examine fallible behavior, which is modeled as a stochastic process, and more fine-grained interactions between the searcher and an IR system.

Section 2 describes our research design. In Section 3 we describe our approach to modeling behavioral factors. Section 4 presents the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

CIKM'13, October 27 - November 01 2013, San Francisco, CA, USA

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-2263-8/13/10...\$15.00.

<http://dx.doi.org/10.1145/2505515.2505660>







constraint was reached or alternatives in query modification ran out.

There is a cost involved with the subtasks of formulating the query, scanning, reading snippets and (full) documents, and judging their relevance. Table 1 gives the subtask costs derived from earlier studies [1], [15] and employed in the present study. Even though reading and evaluating a document depend on the document length, we assumed the given average cost of 30 seconds, for reading and evaluating a document for its relevance.

**Table 1. Average subtask costs (in seconds)**

Session subtask	Cost
Entering a query word	3.0
Scanning one document snippet	4.5
Reading to assess one document	30.0
Entering the relevance judgment	1.0

We operationalized time constraints by setting the maximum session duration to 3 and 6 minutes. The simulated sessions were not allowed to continue by new subtasks (e.g., typing a query or browsing snippets) once the time limit was reached. We also experimented with the open case (no time limit).

### 3.2 Query Formulation Strategies

We explored the limits of session-based searching by experimenting with a limited set of five search words for each particular topic. By using five words it is possible to construct 31 different word combinations (i.e., unstructured queries) for a topic. We had test persons to generate in a systematic way the five words for 41 topics of the TREC 7-8 test collection.

We limited the length of the sessions to at most three queries. The query formulation strategies entail all one to three query permutations of the 31 possible queries, producing nearly 28K distinct strategies.

We paid attention in particular to the following four prototypical query formulation (QF for short) strategies (S1-S4) selected from among the almost 28K QF strategies (see [1] [4] [12]):

S1: One-word variations:  $w_1 \rightarrow w_2 \rightarrow w_3$

S2: Second word variations:  $w_1 w_2 \rightarrow w_1 w_3 \rightarrow w_1 w_4$

S3: Third word variations:  $w_1 w_2 w_3 \rightarrow w_1 w_2 w_4 \rightarrow w_1 w_2 w_5$

S4: Two words extended:  $w_1 w_2 \rightarrow w_1 w_2 w_3 \rightarrow w_1 w_2 w_3 w_4$

We studied exhaustively the effectiveness of all possible, nearly 28K, QF strategies and the two other baselines (single long query and random sessions) as explained in Section 2.2.

### 3.3 Snippet Scanning Strategies and Stopping

A searcher may in principle scan one or more documents after each query before formulating the next query candidate or ending the session. In more detail, consider the handling of a *single* query  $Q_i$  result up to 10 document snippets:

$$Q_i \rightarrow s_{i1} \rightarrow c_{i1} \rightarrow r_{i1} \rightarrow j_{i1} \rightarrow s_{i2} \rightarrow s_{i3} \rightarrow c_{i3} \rightarrow r_{i3} \rightarrow j_{i3} \rightarrow \dots$$

Here  $s_{ij}$  stands for scanning a snippet,  $c_{ij}$  clicking on the snippet,  $r_{ij}$  reading the linked document, and  $j_{ij}$  judging its relevance. In the deterministic case, the simulated searcher clicks on every snippet representing a relevant document, and reads and judges every relevant document. The cost of this session is composed of the costs of its component subtasks.

In the literature (e.g. [7]), one can find several models for scanning behavior, such as the *Cascade model* or *Expected Search Length*

and measures based on these models such as *Expected Reciprocal Rank* and *Expected Browsing Utility* to describe browsing behavior. However, these models do not cover varying session gain goals nor gains cumulated through earlier query results of the current session. Yet these factors affect the searcher's decision whether to continue scanning or to stop the scan in favor of query formulation or to end the session. We propose such a formula  $P(\text{skip})$  below. It gives the probability for the searcher to skip scanning of the current query result at the current rank. To our knowledge, there is no empirical model for multi-query browsing patterns under various search goals. Therefore the parameters of the formula need to be estimated in future studies.

$P(\text{skip}) =$

$$\max\left(\min\left(1, \frac{\sum_{j=1}^{Cur.Doc} \beta^j}{Cur.Doc} - \alpha \frac{\sum_{j=1}^{Cur.Doc} g(j)}{Cur.Doc} + (1 - \alpha) \frac{Gain_{prev.query}}{Total Goal}\right), 0\right)$$

In the  $P(\text{skip})$  formula, *Cur.Doc* is the current rank, " $\beta$ " is a parameter for current query effort factor, " $\alpha$ " is a weighting factor between the current query gain and the gain of previous queries in the session ( $\frac{Gain_{prev.query}}{Total Goal}$ ), and  $g(j)$  is the gain of the  $j$ th document in the ranking. *Total Goal* is the gain at which user information need is satisfied. The first major component in the formula accumulates searcher effort by each snippet/document scanned for the current query, thereby increasing skipping probability. The second component decreases the skipping probability for each relevant document found, representing growing searcher interest. The third component takes the effect of previous queries into account; the more the previous queries accumulate gain, the more the skipping probability increases. Skipping probability increases especially when current query does not retrieve relevant documents. The parameter " $\beta$ " ( $\beta \geq 1.0$ ) is set to 1.1 in the present experiments and the parameter " $\alpha$ " ( $0 \leq \alpha \leq 1$ ) to 0.5 in order to give equal importance between both types of gains.

### 3.4 Relevance Related Behavior

Snippets are not always informative and/or the searcher does not understand (or overlooks) their relevance [18]. Moreover, the searcher does not always understand (or notice) the relevance of the documents (s)he has read. Therefore their relevance judgments may be incorrect. According to [8] [19], this depends on document relevance level. These can be modeled as probabilities. Table 2 shows correct clicking and assessment probabilities by the relevance degree of the underlying document. For example, the simulated searcher will click the snippet of a non-relevant document (of relevance degree 0) with the probability of 27%. The probabilities increase toward highly relevant documents (cf. [19]) which are judged as relevant with the probability of 97%.

**Table 2. Action probabilities and relevance scores by document relevance degree**

Feature of Behavior	Relevance degree			
	0	1	2	3
Clicking Probability	0.27	0.27	0.34	0.61
Judgment-as-Relevant Prob.	0.20	0.88	0.95	0.97
Flat Gain Scores	0	1	1	1
Skewed Gain Scores	0	1	5	10

We employ two relevance scoring schemes (Table 2): a binary one (named *flat*), and a non-binary one, giving more weight to more relevant documents (named *skewed*). Relevance scoring is difficult to relate to searcher preferences which may vary between searchers

and their situations. As a variable, relevance scoring allows experimentation with possible effects.

## 4. EXPERIMENTAL SETTING

### 4.1 Session Generation

Typically, users continue search sessions until either their information need is at least partly satisfied or they run out of time or ideas for a new query. Thereby they can formulate a varying number of queries in different ways. The scanning length of the search results may fluctuate for many reasons as well. We examined all possible sessions under the constraints explained above. We formed all possible 3 query permutations as sessions using a sequence of all possible queries available per topic, and simulated scanning the results under fallible behavior through one thousand trials. Keskustalo and colleagues [12] indicated that the available query words were likely created in descending order of effectiveness. Therefore, when generating sessions for predefined strategies, the query words were used in that particular order, and not permuted.

There are two types of session generation: deterministic and stochastic. Altogether, we ran 41 topics \* 27931 permutations = 1,145,171 sessions for each experiment in the deterministic case.

Stochastic session generation can be described best with a state automaton, which is depicted in Fig. 1. Searcher's actions depend on probabilities (see Table 2 and Section 3.3).

Because the execution of experiments entails probabilistic decisions, the outcome of every experiment varies accordingly. As we sought statistical stability in our findings, we applied the Monte Carlo method where the experiment is repeated several times. In order to find the optimal number of repetitions (or cycles), we conducted several experiments with varying numbers of cycles. The average of maximum CG values quickly reach an asymptote. For robust results on interactive behaviors we utilized 1000 cycles for each session. Therefore we ran  $27931 * 41 * 1000 = 1,145,171,000$  sessions for each experiment in the stochastic case.

### 4.2 Test Collection and Search Engine

We used a subset of the TREC 7-8 document collection with 41 topics for the experiment. The documents have graded relevance assessments on a four-point scale with respect to the topics. [17]

The IR system *Indri* (<http://www.lemurproject.org/indri/>) with language modeling and two-stage smoothing was used.

### 4.3 Data Analysis

With *by-topic trained best strategy* we mean the strategy that is distinctly optimized for each topic and then CG values are averaged across topics. One should notice that these maximally performing strategies are not the same across the topics. With *across-topics trained best strategy* we mean the strategy that is on average the most robust one across topics but not always the best one for an individual topic.

We conduct robustness analysis of the QF strategies by collecting the top performing strategies which achieve 90% of the maximum gain in each experiment and by comparing them across topics. The ten percent slack enables us to compare the predefined strategies with not only the outliers but the best performers in this range. Therefore practical conclusions can be inferred.

## 5. EXPERIMENTS

### 5.1 Baseline Sessions

We shall first look at how effective our baseline sessions are: single long query sessions, best possible three-query sessions, and what is the expected effectiveness of fully random strategy selection. Table

3 reports for three session types the CG and cost under two scoring schemes and two behaviors: deterministic and stochastic.

**Table 3. The effectiveness and cost (s) of baseline sessions**

Session Type	Deterministic		Stochastic	
	CG	Cost	CG	Cost
<b>Flat Scoring 0-1-1-1</b>				
<b>One Long Query</b>	8.6	406.7	2.4	272.0
<b>Best Session**</b>	9.0	404.9	3.0	292.9
<b>Random Session*</b>	4.7	300.5	1.4	267.1
<b>Skewed Scoring 0-1-5-10</b>				
<b>One Long Query</b>	47.3	406.7	16.0	258.4
<b>Best Session**</b>	48.7	404.9	21.3	292.9
<b>Random Session*</b>	26.8	300.5	10.2	265.4

\* One hundred cycles

\*\* Best by CG (1K cycles)

We can see in Table 3 that deterministic scanning (without frustration) yields roughly three times more gain than the stochastic one but also costs 0.5 to 2.5 minutes more in time. Random sessions yield roughly 55% to 60% of the gain of "one long query" sessions but also require less time, from 75% to 100% of the time of the "one long query" sessions.

Further, Table 3 shows that skewed weighting with fallible scanning yields about 35% of the gain of deterministic sessions whereas flat weighting only yields about 30%.

The best sessions are comparable in gain and cost to the "one long query" session except in the case of stochastic sessions with skewed scoring, where the best sessions gain about 30% more than the long query sessions with a penalty of 30 seconds.

### 5.2 Ideal Sessions

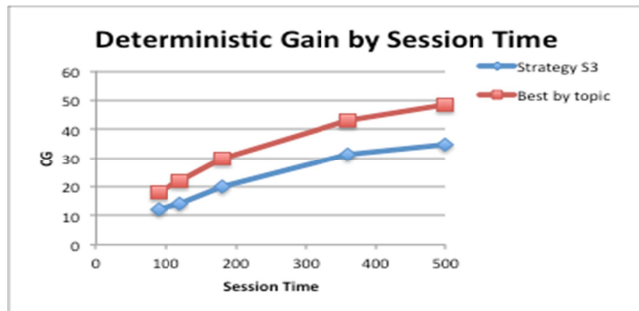
Table 4 presents the results on ideal sessions. We find how effective various QF strategies under time constraints and two weighting schemes are. The column "Actual Time" indicates the actual time spent for the goal. It is often a bit over the constraint, because any action that was initiated before meeting the constraint was carried to its end (e.g., relevance judgments were not interrupted). However, with higher constraints, it is often less than the constraint indicating that the three queries have been used and all results scanned.

**Table 4. The average effectiveness of ideal sessions under cost constraints (s). N= 41 topics, (Selected results)**

Time Constraint	Strategy	Actual Time	CG	
			Flat	Skewed
<b>180</b>	<b>S1</b>	174.1	2.2	14.0
	<b>S2</b>	188.8	3.5	19.9
	<b>S3</b>	187.8	3.4	20.1
	<b>S4</b>	187.3	3.4	19.7
	<b>Best</b>	175.6	4.3	29.6
<b>360</b>	<b>S1</b>	230.6	3.0	19.0
	<b>S2</b>	301.5	5.3	29.4
	<b>S3</b>	312.7	5.4	31.3
	<b>S4</b>	304.6	5.0	28.5
	<b>Best</b>	312.1	7.2	43.2
<b>Open time</b>	<b>S1</b>	238.5	3.1	19.1
	<b>S2</b>	336.0	5.9	31.8
	<b>S3</b>	351.0	6.1	34.7
	<b>S4</b>	326.1	5.3	29.8
	<b>Best</b>	404.9	9.0	48.7

We can see in Table 4 that the predefined strategies S2 and S3 with ideal behavior are the most effective (shaded cells in Table 4). They are clearly more effective than the expected effectiveness of random

sessions (deterministic) at comparable time under both scoring schemes, but also inferior to “one long query” sessions (see Table 3). Under open time and both scoring schemes S3 yields about 73% of the long query performance. Interestingly, the by-topic optimized best sessions exceed the one long query sessions in effectiveness.



**Figure 2.** Gain by deterministic session time; open time set at 500 sec., skewed scoring

The ideal S2 and S3 yield over 80% of the best by-topic trained performance (flat weighting) except when time is unlimited (yielding 67%). With skewed weighting the percentages are about 70. Fig. 2 shows for the best strategy and S3 the cumulated gain by session time. The returns on effort diminish moderately.

### 5.3 Fallible Sessions

Table 5 presents the results on fallible sessions. We find how effective various QF strategies under time constraints are.

We can confirm from results (Table 5), again, the predefined strategy S3 with fallible behavior is the most effective under both scoring schemes. They are clearly more effective than the expected effectiveness of stochastic random sessions (of Table 3) at comparable time under both weighting schemes, but also inferior to “one long query” sessions (in Table 3). The longest sessions are also shorter than the corresponding ideal ones despite of time constraints: due to frustration and skipping, the searcher often runs out of query words before the time constraint hits and ends the session.

**Table 5.** The average effectiveness of fallible sessions under cost constraints (average over 1K cycles). N= 41 topics

Time Constraint	Strategy	Actual Time	CG	
			Flat	Skewed
180	S1	173.6	0.7	5.5
	S2	177.7	1.3	9.1
	S3	179.6	1.4	10.0
	S4	177.4	1.4	9.6
	Best	173.9	2.1	16.0
360	S1	250.5	0.8	6.4
	S2	261.9	1.7	11.8
	S3	262.9	1.9	13.5
	S4	249.3	1.7	12.0
	Best	274.4	2.9	20.3
Open time	S1	266.3	0.9	6.6
	S2	272.0	1.8	12.0
	S3	273.5	2.0	14.1
	S4	254.2	1.8	12.1
	Best	292.9	3.0	21.3

The fallible S3 yields almost 70% of the best by-topic trained performance (flat weighting). With skewed scoring the percentage is 60 to 70.

The predefined strategies and the best across-topic strategy are compared to the best strategy for each topic (Table 6). The table indicates the share of topics for each strategy that achieve at least 90% of the top topic specific performance.

As we can see even the best across-topic trained strategy seldom achieves performance level, which is at least 90% of the performance of by-topic trained optimal strategies. This happens in deterministic case only in 29% to 42% of topics and in stochastic case only in 22% to 24% of topics (see last line of Table 6). In each case, the best predefined strategies are less robust by 5% to 7% units.

**Table 6.** The 90% robustness scores of ideal and fallible sessions under open gain goal and no cost constraints, Share(%) of 41 topics (average over 1K cycles)

Strategy	Deterministic		Stochastic	
	Flat	Skewed	Flat	Skewed
S1	4.9	7.3	4.9	4.9
S2	24.4	36.6	9.8	12.2
S3	24.4	31.7	17.1	17.1
S4	17.1	22.0	9.8	14.6
Best across-topic	29.3	41.5	22.0	24.4

## 6. DISCUSSION AND CONCLUSIONS

We have simulated both ideal human search behavior and the more realistic fallible human search behavior in a test collection. We employed a comprehensive session model allowing multiple queries and several interactive subtasks as depicted in Fig. 1. We held the interface properties, the test collection, and the search engine constant in the simulations, and used fixed probability distributions for snippet and document relevance assessment and snippet scanning behaviors. We then varied systematically the following behavioral factors: (1) the use of QF strategies and (2) cost constraints (time). Empirically grounded QF strategies were compared to three baselines: one long query, the best possible three query session, and random QF with three queries.

The first RQ was about the effectiveness of ideal human behavior employing predefined QF strategies in comparison to deterministic baselines under various CG goal and time constraints. We found that the predefined QF strategies S2 and S3 with ideal behavior are the most effective under time constraints (shaded cells in Table 4). They are clearly more effective than random query sessions with ideal behavior at open time constraints under both scoring schemes, but also inferior to “one long query” sessions (Table 3). Under open time and both scoring schemes S3 yields about 73% of the long query performance and 71% of the by-topic optimized best session performance (skewed scoring). All predefined QF strategies S2 to S4 are close to each other in effectiveness with ideal behavior and are clearly better than the random query baseline (by up to 30%). Effective sessions seem to consist of queries of at least 2 words.

The second RQ was about the effectiveness of fallible human behavior employing predefined QF strategies in comparison to stochastic baselines. The predefined QF strategy S3 with fallible behavior is the most effective under time constraints and gain goals. S3 is clearly more effective than the expected effectiveness of random query sessions with fallible behavior at open time constraint under both scoring schemes (by 40%), but also inferior to “one long query” sessions (by 12-15 %, Table 3). When one does not have the query words initially, trial and error with Bates’ vary tactic [5] seems effective. All predefined QF strategies S2 to S4 are close to

each other in effectiveness with fallible behavior. The strategy S1 is again ineffective as in deterministic case.

The third RQ was about the difference in effectiveness between ideal and fallible human behavior. In the baselines, the effectiveness of the fallible behavior is 28% to 44% of the ideal behavior. Comparing results, fallible behavior reaches 27% to 58% of ideal behavior effectiveness, with averages of 39% (flat scoring) and 46% (skewed scoring). Probabilistic scanning and fallible relevance assessment clearly decrease effectiveness but less regarding highly relevant documents – reflected in the higher figure for skewed scoring – due to fewer errors in their assessment (cf. Table 2). This is an encouraging finding about human effectiveness.

The fourth RQ was about identifying a winning QF strategy across topics. Table 3 shows the effectiveness of the by-topic optimized best sessions. For these figures, different topics may have had different formulation strategies. Indeed, there was no QF strategy among the almost 28K examined ones that was optimal for more than one topic. Table 6 shows that the best across-topic QF strategies achieve good performance in 29% to 42% of the topics in the ideal case, and in 22% to 24% of the topics in the fallible case. For the best predefined QF strategies the percentages are 5% to 7% units smaller. The findings suggest that topic-focused interaction is necessary for good session effectiveness. Alternatively, if one has the words available, a long query combined with persistent scanning is effective.

The fifth RQ was about the simulation methodology. We employed a comprehensive multiple query session model including several subtasks, and several behavioral factors, goals and constraints. This combination is unique. Prior simulations of interactive IR have focused on single query sessions (e.g., [15]), have only had one goal of maximizing gain (e.g., [3]), have a snippet scanning model not taking the sessions goal (or frustration) explicitly into account (e.g., [7]), have not employed time constraints (e.g., [3][12]), have not considered QF strategies (e.g., [15]), or have not considered fallibility in snippet or document relevance assessments (e.g. [4]).

Despite of the great number of sessions simulated, much more could have been done. The possibilities and the limitations at the present study include: the parameters of the experiments, the search engine(s), the search interface(s), and the test collection.

## 7. CONCLUSIONS

We have proposed a novel approach to study the effects of searcher-related behavioral factors in interactive IR on retrieval effectiveness. This approach allows extending the use of traditional test collections to incorporate behavioral factors in a controlled experimental design.

We found among others that (a) there is no single best strategy for all topics but the strategy must be adapted to the topic, (b) the best predefined strategies are top-scoring in no more than one in six topics; however some strategies people use in real life are clearly inferior even compared to randomly structured queries; the best strategies utilized two or three words per query, (c) fallible behavior is clearly inferior to the ideal one while the latter is not realistic, and (d) the models for scanning behavior proposed earlier in the literature for individual queries should be extended for multi-query sessions and varying search goals.

## 8. ACKNOWLEDGMENT

This research was funded by Academy of Finland grant #133021.

## 9. REFERENCES

- [1] Azzopardi, L. 2011. The economics of interactive information retrieval. In: *Proc. of the 34<sup>th</sup> SIGIR Conf.*, pp. 15-24.
- [2] Azzopardi, L., Järvelin, K., Kamps, J. and Smucker, M. 2010. Report on the SIGIR 2010 Workshop on the Simulation of Interaction. *SIGIR Forum*, 44(2): pp. 35-47.
- [3] Baskaya, F., Keskustalo, H. and Järvelin, K. 2011. Simulating Simple and Fallible Relevance Feedback. In: *Proc. of the 33<sup>th</sup> ECIR*, pp. 593-604.
- [4] Baskaya, F., Keskustalo, H. and Järvelin, K. 2012. Time Drives Interaction: Simulating Sessions in Diverse Searching Environments. In: *Proc. of the 35<sup>th</sup> SIGIR Conf.*, pp. 97-106.
- [5] Bates, M. J. 1979. Information search tactics. *JASIST*, 30(4):205-214.
- [6] Carterette, B., Kanoulas, E. and Yilmaz, E. 2011. Simulating Simple User Behavior for System Effectiveness Evaluation. In: *Proc. of the 20<sup>th</sup> CIKM Conf.*, pp. 611-620.
- [7] Carterette, B., Kanoulas, E. and Yilmaz, E. 2012. Incorporating variability in user behavior into systems based evaluation. In: *Proc. of the 21<sup>th</sup> CIKM Conf.*, pp. 135-144.
- [8] Dupret, G. and Piwowarski, B. 2013. Model Based Comparison of Discounted Cumulative Gain and Average Precision. *Journal of Discrete Algorithms*, 18:49-62.
- [9] Harman, D. 1992. Relevance feedback revisited. In: *Proc. of the 15<sup>th</sup> SIGIR Conf.*, pp. 1-10.
- [10] Ingwersen, P. and Järvelin, K. 2005. *The Turn: Integration of Information Seeking and Retrieval in Context*. Springer, 448 p.
- [11] Kelly, D. 2009. Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, 3(1-2):1-224.
- [12] Keskustalo, H., Järvelin, K., Pirkola, A., Sharma, T. and Lykke, M. 2009. Test Collection-Based IR Evaluation Needs Extension Toward Sessions. In: *AIRS Conf.*, pp. 63-74.
- [13] Sakai, T. 2006. Give Me Just One Highly Relevant Document: P-Measure. In: *Proc. of the 29<sup>th</sup> SIGIR Conf.*, pp. 695-696.
- [14] Smith, C. L. and Kantor, P. B. 2008. User Adaptation: Good Results from Poor Systems. In: *Proc. of the 31<sup>st</sup> SIGIR Conf.*, pp. 147-154.
- [15] Smucker, M. D. 2009. Towards Timed Predictions of Human Performance for Interactive Information Retrieval Evaluation. In: *Third Workshop on HCIR*.
- [16] Smucker, M.D. and Clarke, C. 2012. Time-Based Calibration of Effectiveness Measures. In: *Proc. of the 35<sup>th</sup> SIGIR Conf.*, pp. 95-104.
- [17] Sormunen, E. 2002. Liberal Relevance Criteria of TREC – Counting on Negligible Documents? In: *Proc. of the 25<sup>th</sup> SIGIR Conf.*, pp. 324-330.
- [18] Turpin, A., Scholer, F., Järvelin, K., Wu, M.F. and Culpepper, S. 2009. Including Summaries in System Evaluations. In: *Proc. of the 32<sup>nd</sup> SIGIR Conf.*, pp. 508-515.
- [19] Vakkari, P. and Sormunen, E. 2004. The influence of relevance levels on the effectiveness of interactive information retrieval. *JASIST*, 55(11):963-969.