Query Refinement for Patent Prior Art Search

Doctoral Dissertation submitted to the
Faculty of Informatics of the Università della Svizzera Italiana
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

presented by
Parvaz Mahdabi

under the supervision of
Prof. Fabio Crestani and Dr. Monica Landoni

July 2014

Dissertation Committee

Prof. Kai Hormann	Università della Svizzera Italiana, Switzerland
Prof. Evanthia Papadopoulou	Università della Svizzera Italiana, Switzerland
Prof. Gareth J. F. Jones	Dublin City University, Ireland
Prof. Andreas Rauber	Vienna University of Technology, Austria

Dissertation accepted on 30 July 2014

Research Advisor	Co-Advisor
Prof. Fabio Crestani	**Dr. Monica Landoni**

PhD Program Director

Prof. Stefan Wolf and Prof. Igor Pivkin

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

Parvaz Mahdabi
Lugano, 30 July 2014

To My Loved Ones

Somewhere, something incredible
is waiting to be known.

Carl Sagan

Abstract

A patent is a contract between the inventor and the state, granting a limited time period to the inventor to exploit his invention. In exchange, the inventor must put a detailed description of his invention in the public domain. Patents can encourage innovation and economic growth but at the time of economic crisis patents can hamper such growth. The long duration of the application process is a big obstacle that needs to be addressed to maximize the benefit of patents on innovation and economy. This time can be significantly improved by changing the way we search the patent and non-patent literature.

Despite the recent advancement of general information retrieval and the revolution of Web Search engines, there is still a huge gap between the emerging technologies from the research labs and adapted by major Internet search engines, and the systems which are in use by the patent search communities.

In this thesis we investigate the problem of patent prior art search in patent retrieval with the goal of finding documents which describe the idea of a query patent. A query patent is a full patent application composed of hundreds of terms which does not represent a single focused information need. Other relevance evidences (e.g. classification tags, and bibliographical data) provide additional details about the underlying information need of the query patent.

The first goal of this thesis is to estimate a uni-gram query model from the textual fields of a query patent. We then improve the initial query representation using noun phrases extracted from the query patent. We show that expansion in a query-dependent manner is useful. The second contribution of this thesis is to address the term mismatch problem from a query formulation point of view by integrating multiple relevance evidences associated with the query patent. To do this, we enhance the initial representation of the query with the term distribution of the community of inventors related to the topic of the query patent. We then build a lexicon using classification tags and show that query expansion using this lexicon and considering proximity information (between query and expansion terms) can improve the retrieval performance. We perform an empirical evaluation of our proposed models on two patent datasets. The exper-

imental results show that our proposed models can achieve significantly better results than the baseline and other enhanced models.

Acknowledgements

So many people have helped me come to this point in my education. My greatest debt is to my advisor, Fabio Crestani and my co-advisor, Monica Landoni. Thanks to Fabio's confidence in me I was able to explore research directions with freedom, while getting suggestion and feedback from him at critical times. I would like to thank Monica for the scientific support, for her incredible energy and for her friendship specially in difficult moments.

I am grateful to my doctoral thesis committee, Andreas Rauber, Gareth Jones, Kai Hormann, and Evanthia Papadopoulou for reading my thesis and providing me with valuable feedback. I am also thankful to the anonymous reviewers of my papers for providing me valuable feedback which helped me improve the quality of my work.

I also thank Linda Andersson, Florina Piroi, Mihai Lupu, Allan Hanbury, and John Tait for their support and interesting discussions on patent search, computational linguistics, and Information Retrieval. In particular I thank Linda Andersson for sharing with me her enthusiasm for patent search and for opening my eyes to the beauty of computational linguistics.

I would like to thank the members of the IR group, Mostafa Keikha, Shima Gerani, Ilya Markov, Giacomo Inches, Mark Carmen, and Morgan Harvey for sharing with we me their knowledge of the field of information retrieval and their passion for statistics and machine learning in one way or another. In particular, I would like to thank Mostafa Keikha and Shima Gerani, for giving me their insights, suggestions, feedback and for many interesting discussions about my work.

I also thank Jimmy Huang for his supervision during my internship in York University in Toronto. I thank Narges Nattaghi, Vivian Hu, Zahra Amin Nayeri and Hoda forghani for their friendship during this period.

I would like to thank Claire Cardie for her support during few months that I spent in her Natural Language Processing lab at Cornell. It was very exciting to learn about computational linguistics and machine learning through the passionate discussions in the weekly NLP-seminars. Thanks to Lu Wang and Bishan

Yang for the interesting discussions and their collaboration.

My deepest thanks to all the faculty, staff, and students at University of Lugano who helped me in ways large and small. My special thanks goes to the staff members of the Faculty of Informatics, Nina Caggiano, Elisa Larghi, and Danijela Milicevic for providing help and warm support during my PhD time.

My special thanks to my office mates (cubicle mates), Randolf Schärfig, Tim Winkler, Francesco Alberti, Saeed Aghaee, Mattia Vivanti, Amanj Sherwany and Nosheen Zaza for interesting discussions and tolerating me in difficult moments.

I am particularly thankful for the extraordinary friendship of Parisa Jalili Marandi, for the many good times we shared and for the endless encouragement and empathy she had for me. In particular her amazing zest for research has always stayed with me and inspired me. When things got tough, Parisa often reminded me not to give up and to look at research problems like a detective trying to solve a challenging case. I also thank Maïa Khassina, Rebeka Johnson, and Sara Vannini for all the good times we shared and for making my PhD life multi-dimensional. I want to thank the following students at the university of Lugano, for all the fun we shared and for their friendship, which make part of my life today: Lile Hattori, Amir Malekpour, Sandeep Kumar Dey, Mehdi Mirzaaghaei, Elena Khramtcova, Marianna Saba, Navid Ahmadi, Hamid Ghods Elahi, Olga Kaiser, Somaye Danafar, Sebastian Daum, Marco Pasch, Katharina Hohmann, Ricardo Padilha, and Kourosh Khazai.

I am grateful to my husband and my best friend, Nicolas Schiper, for his generous support, who encouraged me and stood by me through thick and thin. I am thankful to my parents in law for their sincere support and continuous encouragement while I was writing my dissertation. I am thankful to them for the numerous interesting discussions we had on research, creativity, innovation, and inspiration.

I sincerely thank my brother, Mahdad, for many years of moral support and placing his trust in me. I thank my family and close friends, in particular my grandmother and my aunts for their endless encouragements, when things got tough.

To finish, I want to dedicate this work to the memory of my parents, Simin and Bahram, who always encouraged me throughout difficult challenges in my life. Thanks for installing in me a love for learning and appreciating beauty. Thanks for transmitting your sensitivity, faith, and strength to me. Although you are not physically present, your unconditional love will always be with me, in every single day of my life. Thank you.

Contents

Figures

Tables

1

Introduction

"One doesn't discover new lands without consenting to lose sight, for a very long time, of the shore."
– Andre Gide

1.1 Introduction

The role of intellectual property in the form of patents is indispensable in the growth of today's knowledge-based economy where production and services are based on knowledge-intensive activities. This makes the patent information retrieval an economically important search activity.

A patent is a legal document granted by a state or by a regional office acting for several states, which gives a set of rights of exclusivity and protection to the owner of an invention for a limited period, generally 20 years. The patent allows the inventor to exclude anyone else from making, using, selling, offering for sale, or importing the patented invention [79]. In exchange for this right of exclusivity, the owner of the patent is obliged to disclose the details of the invention to the public as well as the related technical and scientific background of the invention.

Patents are granted by patenting authorities or central offices that are usually part of the national governments in many countries around the world. The process by which patenting authorities and inventors negotiate toward the terms of a patent is called "patent examination" and is also referred to as "patent prosecution". Patent examiners, who are employed by a national or regional patenting authority, conduct patent examination.

A particular invention needs to meet the following conditions to be granted a patent: novelty, non-obviousness, and utility. The novelty criteria states that an invention must not have been described or used before the filing of a patent application. The non-obviousness criteria requires an invention to not be an obvious combination of existing processes or entities. The final criteria, the utility criteria, demands that a patent can be built and used in practice.

During examination, the patent examiner will perform "prior art search" with the aim of finding public disclosure of the features of the invention, that were available prior to the filing date of a patent application and may invalidate the novelty of a claim. Simply put, all patent and non patent literature that have been published prior to the filing date of a patent application need to be searched with the goal of finding documents that are similar to the patent application.

Both patent applications and granted patents are structured documents with the following fields:

Abstract. This field represents a brief summary of the invention.

Description. This field provides a detailed description of the invention, including prior work, examples, and related technologies.

Claims. This field presents the legal description of an invention.

Bibliographical data. This field is composed of the title, and the metadata related to the patent document such as the inventors, assignees, agents or applicants, and the relations to other documents.

Patent search is a general term that covers different types of search processes such as *technology survey*, *prior art search*, *freedom to operate*, *validity* and *patent portfolio search*. Different processes in patent search differ in terms of the underlying information need, and the needed search output. The information need can be an idea, a patent application, a claim, or a granted patent. The required output can be one of the following items: a set of patents, a single patent, a set of scientific publications covering a domain, or public documents (i.e. anything that is disclosed to public) [64]. We now present a short description of the different tasks in patent search:

Technology Survey. In this search process, the input information need is an idea and the output is to obtain a general understanding of the innovation by searching in all public documents.

Prior Art Search. The input information need is a patent application and the search is performed over all public documents until the date of the application. The purpose of this search is to identify whether a given patent application satisfies the condition for granting. This task is also referred to as Novelty, or Patentability Search.

Freedom to Operate. The information need is about a product and its related technologies. This search is carried out on a set of patents in force in a specific jurisdiction. This process is also called Infringement, Right-to-Use, and Clearance.

Validity. In this search process, the information need is composed of a granted patent and all the public documents prior to the *priority date* of the patent in question will be searched to identify if a granted patent satisfied the granting criteria at the earliest priority date. Priority date is the moment when a first application was registered for the invention described. Other names for this search process are: Invalidity, Enforcement Readiness, and Opposition.

Patent Portfolio Search. In this search process, the information need is a company, or a technology area. The goal of this search is to obtain a general understanding of the patents, in a specific technology area by looking into all public documents. This search task is referred to as Due Diligence, and Patent Landscape.

Notice, however, that the precise names and definitions of these search processes vary between those who deal with patents, like for example, information specialists, private patent searchers, patent examiners, and patent lawyers. Chapter 2 will describe the field of information retrieval (IR) in more detail and presents how patent retrieval is related to IR.

In the next section we explain our motivations for pursuing this line of research.

1.2 Motivation

Although general information retrieval has advanced immensely in the recent years and Web Search engines were revolutionized (by Google), the systems which are in use by the patent search communities do not yet take advantage fully of the advancement in search technologies. More specifically, the Boolean search is still common instead of the ranked retrieval systems [6].

Considering the increase in the number of patent applications, and the long duration of the application process which leads to the big backlogs of patent offices, there is a visible need to change the way the patent information is searched.

In this thesis we focus our attention on the patent prior art search which is a critical step in the examination and evaluation process of a patent application. Prior art search is a challenging task, with different issues when compared to other search tasks, such as web search.

Here we review some of these challenges:

- The first challenge is that the starting point of the prior art search task is a full patent application. In this task the information need is presented by a patent document (comprised of a hundred words) instead of a short web search query (composed of two or three keywords). Therefore, a major challenge is how to transform the patent application into search queries in order to find similar documents.

- A second challenge is related to the vocabulary usage in patent domain which is very unique and far from everyday speech and writing and often contains highly specialized or technical words not found in everyday language [6]. Writers tend to use many vague terms and expressions along with non-standard terminology in order to avoid narrowing the scope of their claims [9]. This usage of vocabulary by patent writers results in term (vocabulary) mismatch. Term mismatch refers to a situation where two patent documents have few or no keywords in common, while at the same time the idea conveyed in these two patents are similar. In other words, the query words used to search for relevant documents are not exactly the same terms used in the relevant documents. Term mismatch is a common problem in most of information retrieval tasks – due to spelling mistakes, ambiguity of words and different ways of referring to the same concept. However, this problem is exacerbated in patent retrieval because of its exceptional vocabulary.

- The last issue is related to the fact that patent prior art search is a recall oriented application where the goal is to retrieve all relevant documents at early rank positions as opposed to ad hoc and web search, where the goal is to achieve high precision. In prior art search, the searcher needs to ensure s/he is not missing on any relevant document as infringing on some existing patents might result in a multi-million dollar lawsuit; consequently, this search can take a very long time (days or weeks) [54].

We focus on the following properties of patents and propose solutions for integrating them in a retrieval system in order to address the challenges mentioned previously.

- The first property is linked to the structure of patents. Patent documents are structured documents with different fields such as abstract, description, and claims. Patent writers use different style of writing for describing the invention in different fields of patent. For example, the abstract and description use a technical terminology while the claims field uses a legal jargon [107].
- A variety of different techniques have been employed in previous studies for identifying effective query terms, mainly looking into the distribution of term frequency [83, 90]. But, in addition to the textual content of patent documents, there are other rich relevance evidences (e.g. classification information, bibliographical information) which can be utilized for finding documents relevant to a patent. The second property is related to these dependencies which can be employed for minimizing the term mismatch between a query patent and documents relevant to it [6].
- The last property is related to language evolution over time. Terminology of technological domains is changing over time; as a result, identifying the vocabulary of different time intervals is helpful to address the term mismatch.

1.3 Research Outline and Questions

The following are the research questions that we tried to address in this thesis.

RQ1 How can we estimate a query model from a patent application (query patent) using the different textual fields available in it?

This general research question leads to the following detailed sub-questions:

(I) Which field of the patent application serves as a more effective source for extracting query terms?

(II) Are there any advantages in using phrases in the query representation, in addition to terms (uni-grams)?

(III) For what type of queries can we expect the noun phrases to be more effective?

RQ2 How can we leverage the classification tags (associated to patent documents) to construct a domain dependent lexicon?

More detailed sub-questions are:

(I) Is the IPC conceptual lexicon useful for query expansion?

(II) Is the proximity information between query terms and expansion terms, extracted from the IPC conceptual lexicon, helpful in identifying weights for expansion terms? How can we model such proximity?

(III) What are the best parameters of the kernel function used for modeling proximity information?

RQ3 Could we use the citation links, the content, and the temporal features of the cited documents to expand the initial query model built from the query patent?

We break this general research question into following sub-questions:

(I) Do citation links together with the content of the cited documents improve the performance of the initial query built from the query document? Does employing the temporal features of the query and of the collection result in a more precise query?

(II) What type of document prior is more effective in modeling the decay over time?

RQ4 Is combining different dependencies that are available with the initial query patent helpful for improving the query formulation? Assuming that it is helpful, how can we combine different dependencies to formulate a query?

The remainder of this section discusses how the thesis addresses the above objectives.

1.4 Main Contributions

In this section we summarize the main contributions of this thesis:

- We investigate different ways of estimating a query model from a query patent utilizing patent-specific characteristics.
- We present a method for predicting whether query expansion using noun phrases (concepts) will improve the retrieval effectiveness in a selective query expansion framework – we extract different features from each query and its initial rank list (both pre and post retrieval features) to predict the quality of queries. We then make a query dependent decision for expansion using noun phrases.
- We present an approach to construct a domain-dependent lexicon for identifying query expansion concepts. We develop a proximity-based query expansion method for estimating the probability that an expansion term is relevant to a query term. We also investigate different query reformulation strategies for extracting concepts from a domain-dependent lexicon.
- We develop an approach for boosting the initial query model using a topic-sensitive graph built from the citation links. We propose an approach for exploiting the temporal features of documents in the citation graph in order to improve the query representation.
- We present a framework to combine multiple relevance evidences associated with a query patent, namely classification and bibliographic information.
- We perform experimental evaluation and validate our proposed models on two patent retrieval test collections.

1.5 Thesis Overview

Chapter 2 and 3 of this thesis play the role of introductory chapters to familiarize the reader with the field of information retrieval (IR) and the experimental evaluation in the context of IR. This thesis consists of five main research chapters, Chapters 4 – 8, each addressing the set of research questions introduced earlier. Finally we present concluding remarks in Chapter 9.

Chapter 2. We present an introduction to IR in general and patent prior art search in particular as a specific task which we address in this thesis. We explain the related work in the area of patent retrieval and focus on the challenges that were not properly addressed in previous work.

Chapter 3. We introduce the evaluation methodology and the experimental setup that forms the basis of the empirical evaluations throughout this thesis. The test collections, the set of test queries, and evaluation metrics are introduced in this chapter.

Chapter 4. Next, we present a solution to the patent prior art search problem allowing the user to submit a full patent document as a query and the retrieval system identifying related patent documents from a corpus accordingly. We first define the query generation problem and describe three approaches to estimate the topic of a query patent. We also explore generating queries from different fields of the patent documents. Our contribution is to build an effective term selection and weighting technique using a weighted log-likelihood based approach to distinguish words which are both indicative of the topic of the query and are not extensively used in the collection. We also investigate query modeling based on the Parsimonious Language Model for estimating a query model from the query patent. Furthermore, we utilize the knowledge embedded in IPC classes. This addresses the vocabulary mismatch as we include words in the query which are not present in the query topic itself.

This chapter provides answer to sub-question RQ1.(I).

Chapter 5. We propose a technique to process patent documents and extract terms and key phrases in order to form a query to retrieve relevant documents from the patent corpus. This approach refines the initial query by expanding it with selected key concepts (i.e., bi-grams or phrases) from the query patent using the global analysis of the patent collection. Query expansion using noun phrases is not consistently beneficial for all queries, as sometimes the expansion candidates are not associated with the main aspect of the query. For example, for a patent application related to “water filtration”, the selected expansion candidates are “removing filters”, “continuous pores” and “integrating bag” which are focused on the partial aspects (subtopics) of the query while the main aspect of the topic related to “filter material” (such as carbon and/or ceramic) is neglected. We thus decided to perform query expansion in a query dependent manner to guar-

antee the inclusion of expansion candidates related to the main aspect of the topic to guarantee a profitable expansion.

In this chapter we propose a method for distinguishing between queries and deciding when to selectively use the result of a refinement technique that is likely to improve the retrieval performance. Our goal is to identify queries that have highly positive changes in query performance using refinement. To this end, we use query performance predictors (pre and post-retrieval) [4, 26, 48, 104] and patent-specific features in order to find highly performing queries in the expanded retrieval rank list. To the best of our knowledge no previous work has used query performance predictors in the patent domain. To decide when to use the result of the expanded list, we rely on a machine learning approach that tries to predict which one of two competing approaches will offer the best result for a given query.

This chapter provides answer to subquestions RQ1.(II) and RQ1.(III).

Chapter 6. Here, our aim is to address the term mismatch problem by taking advantage of IPC classifications which categorize patent documents by topics. A simple analogy can be made between IPC classes and tags associated to news articles or tweets. We are interested to use IPC classification tags to improve the representation of the query. These IPC classifications are assigned to a patent document in the patent office before the prior art search. Thus, they can be exploited at the time of prior art search.

Definition of IPC classes consists of the explanations regarding each IPC class. This vocabulary serves as an established and accepted terminology among the practitioners in the domain, shared between examiners and inventors. In this chapter, our aim is to address the term mismatch problem by taking advantage of the established terminology of IPC classes and identify related terminology to the important topics and subtopics of the query.

The language of the patent documents uses a less standard terminology, compared to the terminology of IPC definitions. The reason is related to the frequent usage of non-standardized acronyms which are invented by patent applicants, the presence of homonyms (the same word referring to two or more different entities), such as bus[1] and closet[2], and synonyms (i.e. signal and wave). Paraphrasing is another source of term mismatch, where a re-wording is used to express the meaning of a concept with a

[1] i) motor vehicle, ii) an electronic subsystem transferring plurality of digits bits in group.
[2] i) water closet (flush toilet), ii) a small cupboard used for storing things.

greater clarity (for example “a drink sucking hollow plastic tube” is used to refer to “straw”).

We propose a proximity-based query propagation method to calculate the query term density at each point in the document. We then use term proximity information to calculate reliable importance weights for the expansion concepts. Our proximity-based framework incorporates positional information into the estimation of importance of expansion concepts so that we can reward expansion concepts occurring close to query terms. This way we can concentrate on the terms that are associated with the query terms and avoid the topic drift which is caused by taking into account irrelevant terms.

RQ2 is addressed in this chapter.

Chapter 7. Next, we focus on taking advantage of the citation links – similar to hyper links in the web – between patent documents in the collection to improve the term mismatch problem. We look into temporal information to adapt to the change of the language in the cited documents over time.

We first perform a citation link analysis over the patent citation graph using Page Rank method to identify important documents, which could influence their domain terminology. The assumption is that if a patent is cited by a large number of documents, the cited patent is possibly a foundation of the citing patents and is considered influential on other patents. As a result of its impact, its language might be useful to bridge the gap between the query and its relevant documents.

We are interested to improve term mismatch solutions by tapping the power of the community of inventors related to the subject of the invention of the query. In other words, through citation link analysis we identify a set of terms which are relevant to a given query document and can be exploited for improving the original ranking to find documents that do not contain the exact wording of the query patent.

In order to consider the dynamic nature of the patent citation network and recognize the new influential nodes which are added to the network but have not stayed long enough to accumulate sufficient links, we parametrize the random walk with a time factor. We do this by considering the temporal order of the nodes in the citation network. We discount the initial probability of selecting a node as the seed of the Page Rank algorithm according to some temporal decay factor. To the best of our knowledge, there

has been little or no work on using temporal information for modeling the change of the vocabulary over time in the patent domain.

RQ3 is addressed in this chapter.

Chapter 8. We propose a unified framework to take advantage of all dependencies associated to a query patent such as IPC classifications, citation links, the content and the temporal features of the cited documents to enhance the initial query representation estimated from a query patent.

RQ4 is addressed in this chapter.

Chapter 9. In the final chapter, we report the conclusions and list possible future directions.

1.6 Publications

This thesis is based on several published works. Chapter 4 which estimates query models using the query patent document (patent application) is based on the following works:

- Mahdabi, P., Keikha, M., Gerani, S., Landoni, M., and Crestani, F. Building Queries for Prior Art Search. *Proceedings of Information Retrieval Facility Conference*, pp. 3-15, 2011.
- Mahdabi, P., Andersson, L., Hanbury, A., and Crestani, F., Report on the CLEF-IP 2011 Experiments: Exploring Patent Summarization. *In Conference on Multilingual and Multimodal Information Access Evaluation (CLEF 2011)*

Chapter 5 which presents a selective query expansion framework using noun phrases is based on the following studies:

- Mahdabi, P., Andersson, L., Keikha, M., and Crestani, F. Automatic refinement of Patent Queries using Concept Importance Predictors, *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 505-514, 2012.
- Mahdabi, P. and Crestani, F. Learning-Based Pseudo-Relevance Feedback for Patent Retrieval. *Proceedings of Information Retrieval Facility Conference*, pp. 1-11, 2012.

The proximity-based framework for query expansion which uses an IPC lexicon presented in Chapter 6 is described in:

- Mahdabi, P., Gerani, S., Huang, J., and Crestani, F. Leveraging Conceptual Lexicon: Query Disambiguation using Proximity Information for Patent Retrieval, *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 113-122, 2013.

Chapter 7 which presents the work on using citation information (extracted from the top-ranked documents of an initial rank list) for query expansion is based on the following study:

- Mahdabi, P. and Crestani, F. The Effect of Citation Analysis on Query Expansion for Patent Retrieval, *Information Retrieval Journal*.

Chapter 8 which presents the work on combining different sources of relevance evidence for query expansion is based on the following work:

- Mahdabi, P. and Crestani, F. Patent Query Formulation by Synthesizing Multiple Sources of Relevance Evidence, *ACM Transaction on Information Systems*.

This thesis also includes material that is under review and thus we do not mention it here.

2

Related work

"I believe that reading and writing are the most nourishing forms of meditation anyone has so far found. By reading the writings of the most interesting minds in history, we meditate with our own minds and theirs as well. This to me is a miracle."

– Kurt Vonnegut

2.1 Introduction

In this chapter we introduce the previous work on which this thesis is based. The baseline approaches for prior art search, presented in Section 2.6, are most immediately relevant. However, we build related material in several steps. We start this chapter by recalling the basic concepts and terminology of IR in Section 2.2, to which we refer throughout this thesis. Then, we take a closer look at Language Modeling for IR in Section 2.3. Next, we discuss query transformation methods (such as query expansion) and review the importance of query transformation in different IR tasks in Section 2.4. Then, we describe query quality prediction approaches in Section 2.5 and, Finally, in Section 2.6, we describe the existing work on patent prior art search.

2.2 Information Retrieval

Users require information when performing particular tasks. The motivation for a person to use a search engine is described as the *information need*. Information Retrieval (IR) is a branch of computer science whose goal is to provide effective

models for matching user's expression of their information need with *relevant* information in a collection of documents, or other data. The information need is not observed by IR models directly; instead, an IR system receives an expression of the user's information need, called a query. According to Belkin, a query may be a non-specific, imprecise, and incomplete description of the user's information need [10]. We call a document that satisfies an information need, relevant. A retrieval model scores the documents in the collection, providing a list of relevant documents and preferably ranking them according to some measure denoting how well the query is matched to the information in the documents.

The first popular retrieval models were Boolean systems. These did not generate a ranking; instead, they returned a set of documents fulfilling a Boolean query. Later, the Vector Space Model (VSM) [92] was introduced by Salton et. al which was first implemented and used in the SMART system [40]. VSM was the basis of the research in IR in its early days and the research was focused on textual documents, mainly books and journals, for library applications.

In the VSM, queries and documents are represented by vectors, where each dimension in the vector corresponds to a separate term [92]. The similarity between vectors is defined using a distance measure. The most commonly used distance measure is based on the cosine of the angle between vectors. Each component of a vector goes beyond binary values (non-zero values for denoting term occurrence in a document) and can take statistical information such as term frequency (TF) and inverse document frequency (IDF). The TF of a term in a document is defined as the relative frequency of occurrence of that term in the document. IDF is defined as the log of the inverse of the relative frequency of occurrence of a term in the entire collection.

Traditional notion of relevance [22] centered on topicality (aboutness) also relies on TF and IDF; in fact, these term weightings are in common use in most retrieval models nowadays. The idea behind these weight representations is that documents with a high TF for a term are more likely to be relevant to queries containing this term. Moreover, terms that are infrequent in the collection have more discriminative power and are the ones that describe better the information content. Therefore a common weighting scheme called TF/IDF, a simple combination of the two weighting schema, was introduced in [92]. VSM was the first model to provide a rank list of search results for a query. The disadvantage of VSM model was that it could not specifically explain how these weighting schema and ranking algorithm were related to relevance.

The first work that encouraged the development of probabilistic retrieval model was Probability Ranking Principle (PRP) proposed by Robertson and Spärk Jones [86, 87], in which documents are ranked by their decreasing probability

of relevance to the user who submitted the request. They introduce assumptions such as the relevance of a document to a query being independent of other documents. However, they did not explain how to estimate this probability and it is only later works that proposed methods for estimating this probability [88]. The Okapi team developed an extended version of the PRP model, which makes use of term frequency (TF), inverse document frequency (IDF), and document length. This model which became mainstream is now commonly known as (Okapi) BM25 [55]. This model performed very well in TREC[1] retrieval experiments and has influenced the ranking algorithms of both commercial and web search engines.

A relatively new model, known as language modeling (LM), appeared in the late 1990's. The success of statistical language models in improving tasks in a variety of natural language processing and understanding applications such as speech recognition [53] generated considerable interest among IR researchers. They thus borrowed this notion to represent the document retrieval in a generative probabilistic framework. In such a framework documents are represented as generative probabilistic models. This model is now commonly used and in fact, most of the work in this thesis is inspired by LM. We thus will take a closer look at LM in the next section.

2.3 Statistical Language Models for IR

The simplest form of language model, known as *uni-gram* language model, assigns probabilities to every words in the vocabulary for a collection by means of a probability distribution [84]. This model does not capture the influence of the words before or after the target. This leads to a *bag of words* model and turns out to generate a multinomial distribution over words. The multinomial model explicitly captures the frequency of occurrence of a term.

In applications such as speech recognition, word prediction is based on longer sequences, which are called n-gram language models [56]. An n-gram model predicts a word based on the previous $n-1$ words. The most common n-gram models are bi-grams or tri-grams models where prediction is based on the previous word or the two previous words, respectively. N-gram models can be used to compute the probability of observing a sequence of terms, by calculating the product of the probabilities of observing individual terms. That is, they can tell which possible output word sequences are more probable than others. We will focus our discussion on uni-grams as they are simpler (considering the huge size

[1] Available at `http://trec.nist.gov/`

of textual collections) and proven to be very effective as the basis for ranking algorithms.

2.3.1 Query Likelihood Model

In the *query likelihood* retrieval model, retrieved documents are ranked based on the probability that the document language model would generate the terms of the query [24]. Starting with a query Q, we would like to calculate $P(D|Q)$ to rank the documents. Using Bayes' Rule, we can calculate this by

$$Score(Q,D) = P(D|Q) =^{rank} P(Q|D)P(D) \tag{2.1}$$

where D is a document and the symbol $=^{rank}$ means that the right-hand side is rank equivalent to the left-hand side, ignoring the normalization value P(Q) which is the same for all the documents. It is common to assume that the document prior P(D) is uniform, the same for all documents, and therefore it is safely ignored too.

Documents are then ranked by $P(Q|D)$ (the probability that a query is observed as a random sample from the document model). $P(Q|D)$ is calculated using a multinomial uni-gram language model for the document, considering a simplifying independence assumption between terms in the query:

$$P(Q|D) = \prod_{q \in Q} P(q|D)^{f(q,Q)} \tag{2.2}$$

where q denotes a query word and $f(q,Q)$ indicates the frequency of word q in Q. As multiplying small numbers can cause arithmetic underflow, we use logarithms instead.

$$\log P(Q|D) = \sum_{q \in Q} f(q,Q) \log P(q|D) \tag{2.3}$$

Maximum likelihood estimates, for multinomial distributions, are commonly used to estimate a document's generative language model as follows:

$$P(q|D) = \frac{f(q,D)}{|D|} \tag{2.4}$$

where $|D|$ denotes the length of document D and $f(q,D)$ indicates the frequency of term q in D. Maximum likelihood (ML) is the estimate that makes the observed value of $f(q,D)$ most likely. In a maximum likelihood estimate,

unseen events (terms that do not appear in the document) receive zero probability. Smoothing techniques play an indispensable role to avoid data sparsity problem while calculating ML estimate. Smoothing techniques [21, 112] (e.g. Laplace, and Good Turing) accounts for unseen words by discounting the probability mass of seen words either from the document or from the collection. Another type of commonly used smoothing in IR is *Jelink-Mercer* smoothing which is an interpolation-based smoothing. It considers documents to be a mixture of a document-specific model and a more general background model. The latter is usually estimated based on a sufficiently large collection C.

$$P(t|\theta_D) = \lambda P(t|D) + (1-\lambda)P(t|C) \tag{2.5}$$

where λ denotes the interpolation coefficient and $P(t|C) = \frac{f(t,C)}{|C|}$. $|C|$ is the total number of words in the collection. So far, *Bayesian smoothing using a Dirichlet prior* has been shown to be the most effective on different IR tasks [112], this is its formulation:

$$P(t|\theta_D) = \frac{|D|}{|D|+\mu}P(t|D) + \frac{\mu}{|D|+\mu}P(t|C) \tag{2.6}$$

where μ is a hyper parameter that controls the level of smoothing – it is typically set to the average document length of documents in the collection.

Various extensions of language models have been introduced in the literature. We will discuss two of them that will be used later on in this thesis. In [51], Hiemstra *et al.* introduced a variation of language modeling called the Parsimonious Language Model, where they used an Expectation Maximization based algorithm which takes away probability mass from frequent terms in the general English and gives it instead to terms that are rare in a document.

In [66], the authors proposed a unified language modeling framework called Positional Language Model (PLM), which implements two retrieval heuristics, proximity and passage retrieval. The proximity heuristic rewards a document where the matched query terms occur close to each other. The passage retrieval heuristic scores a document mainly based on the best matching passage. They achieve this goal by defining a language model for each position in a document and score a document based on the score of its PLMs. Each PLM is estimated based on a density kernel function [30] which captures proximity heuristics and achieves an effect similar to passage retrieval.

2.3.2 Relevance Models

The basic query likelihood model can be extended to incorporate information about relevant documents while modeling the query and information need. We

call this a query language model *relevance model* as it represents the topic covered in relevant documents. If we can estimate a relevance model from a query, we can directly compare this language model with the document model. We can then rank documents according to the topical similarity between the document model and the relevance model. A (non-symmetric) measure of difference between two probability distributions called, *Kullback-Leibler divergence* (KL divergence) has been borrowed from probability theory and information theory. Given a true (reference) probability distribution P and another distribution Q, which is an approximation of P, KL divergence is defined as follows:

$$KL(P||Q) = \sum_{x} P(x) \log \frac{P(x)}{Q(x)} \tag{2.7}$$

KL-divergence is always a non-negative value. The larger the value of the KL-divergence, the bigger the difference between the two probability distributions, and equal distributions receive zero divergence value. As we would like to assign a high score for highly similar documents and a low score for less similar ones, thus, the *negative* KL-divergence has been used for ranking in LM [24]:

$$\sum_{t \in V} P(t|R) \log P(t|D) - \sum_{t \in V} P(t|R) \log P(t|R) \tag{2.8}$$

where the relevance model for query (R) is the true distribution and the document language model (D) represents the approximation. The summation is done over all terms t in the vocabulary V.

2.4 Query Transformation

As mentioned earlier, query is a vague formulation of the user's underlying information need. In order to improve the query model and better represent the underlying information need, IR researchers proposed a variety of *query transformation* techniques. The primary goal of query transformation approaches is to improve the retrieval performance and the ranking presented to the user as the result of the submitted query to the search system by changing the query representation [23]. Query transformation is also referred to as *query modeling*, *query reformulation*, or *query refinement* in the literature.

Query transformation is composed of two stages: the first processing stage alters the query at a morphological level (e.g., tokenization, spelling corrections, and stemming); the second stage works on the output of the first stage and modifies the query at a structural level [12]. Modifications in the second stage

include, *query expansion* using related terms [103], identifying key concepts and re-weighting query terms [11], rewriting the query and substituting terms [29], to mention a few. Query expansion is a popular method of query transformation which help addressing the term mismatch problem by adding and re-weighting query terms. Term mismatch is a scenario where matching the query with the relevant documents is hampered, as the terms used in the query are different than those used in the relevant documents.

Query expansion approaches are classified into three main groups: *pseudo*, *explicit*, and *implicit* relevance feedback, depending on how the relevant information sources are acquired. A popular form of query expansion is called *pseudo relevance feedback* (PRF) where the top ranked documents in the initial result of a query (called feedback documents) are assumed to be relevant and are used to locate additional query terms [106]. Explicit relevance feedback uses explicit relevance assessment provided by the user [43]. Implicit relevance feedback, on the other hand, uses query logs [28] to infer the behavior of the user from the low-level user interactions with the search interface, which is logged as click data. Another common approach is to use resources such as thesauri and controlled vocabularies [13], or external corpora [33] as another source for query expansion. The latter approach permits the calculation of collection independent statistics.

Recently, machine learning approaches have been employed to improve the selection of terms and documents in query expansion. Cao et al. proposed to refine PRF at a term level [17]. They use a Support Vector Machine (SVM) to select good expansion terms using a set of term-level features such as the proximity of an expansion term and an original query term, or the co-occurrences of an expansion term and an original query term in the collection. He and Ounis proposed a method to improve PRF by choosing the right documents for relevance feedback [50] among top-ranked documents. Their goal is to predict which feedback documents are of better quality for query expansion, instead of assuming all top-ranked documents to be relevant (the simplifying assumption of PRF). They apply Naïve Bayes classification and Logistic Regression to classify feedback documents. They use document-level features such as the distribution of query terms in the feedback document and all the top-ranked documents or the proximity between the expansion terms and the original query terms in the feedback document.

2.5 Query Performance Prediction

Query performance prediction approaches estimate the effectiveness of a search performed in response to a query without any relevance judgements [18]. Such predictors are classified into two groups:

1. Pre-retrieval predictor methods: this category analyzes the query expression before search is performed [26, 45, 48, 77]. This category of methods look into available information at query time including linguistic features, statistical properties of the query-terms distribution, and the collection-based statistics. The Clarity score [26], measures the degree of ambiguity of a query with respect to a collection of documents by computing the relative entropy between a query language model and the collection language model. The resulting score measures the dissimilarity between the language usage associated with the query and the generic language of the collection as a whole.

2. Post-retrieval predictor methods: this group of methods analyzes the top ranked documents from the result list as a response to a query. The Clarity score reappears in this category too, as it can be used to estimate the focus of the result list with respect to the corpus, as measured by the KL-divergence between their induced language models. Variants of the Clarity score are proposed for improving prediction performance [19, 25, 46].

Query performance predictors are usually evaluated by reporting correlation coefficients to denote how well the methods perform at predicting the retrieval performance of a set of queries [47, 48]. Query performance predictors are used to predict queries that have highly negative changes in Average Precision after query expansion, using a score that does not depend on relevance information, in order to improve the retrieval effectiveness in a selective query expansion framework [4, 25, 104]. These predictors can be used alone or in a combination to predict the effectiveness of a rank list (e.g. Average Precision).

2.6 Patent Prior Art Search

Patent prior art search is composed of a search over available patent and non-patent data (prior to the filing date of a patent application) with the goal of retrieving similar documents, which describe the prior art work of a patent application, (henceforth referred to as *query patent*). The challenges of patent

prior art search are different from those of standard ad hoc text and web search. These differences are categorized below:

- The first distinguishing property of prior art search compared to standard information retrieval tasks (such as web search) is that the information need is presented by a patent document rather than short queries [107]. The challenge here is how to reduce a query patent which comes with a rich set of metadata, in order to find a single focused information need and to remove the ambiguous and noisy terms. In previous work, researchers explored different fields of the query patent to perform query reduction [20, 107]. Some of the previous work reported that effective queries were built from the entire query patent [20]; others obtained better results using single fields such as the "background summary" [107]. It is worth mentioning that the "background summary" field is specific to US patents.

- The second distinct property of patent retrieval is related to the terminology of patents which contains highly specialized and/or technical words not found in everyday language [54]. It also contains exceptional (creative) vocabulary, curious grammatical constructions, regulatory, and legal requirements [6]. These inherent properties lead to an overwhelming term (vocabulary) mismatch. For example one patent document may contain few or no keywords in common with the query patent, but the idea conveyed in it might be quite similar or even identical to the query patent – one possible reason is paraphrasing [6]. As a result, the retrieval problem is exacerbated and standard search systems may be confused. One possible solution for this problem is query disambiguation. Previous work used different external resources for query expansion such as Wikipedia [63] and WordNet [69] with the goal of query disambiguation. The goal here is to alleviate the term mismatch problem by expanding the query with topically related words or synonyms of the query terms.

- The third property is related to the fact that patent prior art search is a recall oriented application where the goal is to retrieve all relevant documents at considerably early rank positions. Ad hoc and web search, on the other hand have the goal of retrieving only a few relevant documents at the top of a ranking and thus achieving high precision [8]. In a prior art search scenario even missing one relevant patent can lead to a multi-million Euro law suit due to patent infringement, and so a high recall is demanded in this type of search.

In this chapter, we first survey different approaches for query formulation in Section 2.6.1 and describe how the textual sections of the patent and the metadata associated to the patent document such as classification are used for query reformulation. We then describe different approaches that use external knowledge bases and proximity information in Section 2.6.2. Then, in Section 2.6.3 we present different techniques that consider patent citation information. Finally, in Section 2.6.4, we explain the evaluation metric designed for patent retrieval.

2.6.1 Query Formulation for Patent Retrieval

The main line of research in patent retrieval started after the third NTCIR workshop in 2003 [52], where a few test collections were released. Starting from the fourth NTCIR workshop in 2004 [36], a search task was presented called "invalidity search run". The goal was to find documents prior to the filing date of a particular granted patent which conflict with the claimed invention. The citation parts of the applications were removed and counted as ground truth. Participants used different term weighting methods for query generation from the claims field.

In [95] the authors studied the rhetorical structure of a claim (an item in the claims field). They segmented a claim into multiple components, each of which was used to produce an initial query. They then searched for candidate documents on a component by component basis. Similar work was introduced in [74] where the authors analyzed the structure of claims field to enhance retrieval effectiveness. The structure of each item of claims usually consists of the *premise* and *invention* parts, which describes existing and new technologies, respectively. The authors proposed a two stage process where they first extract a query from the premise to increase the recall. They then aim to increase the precision by extracting another query from the invention part. The final relevance score of each document was calculated by merging the scores of both stages.

A recent line of work advocated the use of the full patent application as the query to reduce the burden on patent examiners. This direction was initiated by Xue and Croft [107], who conducted a series of experiments in order to examine the effect of different patent fields on the query formulation and concluded with the observation that the best Mean Average Precision (MAP) is achieved using the text from the "background summary" field of the query patent.

Current developments in patent search have been driven by the Intellectual

Property task within the CLEF[2] initiative. Several teams participated in prior art search task of the CLEF-IP 2010 and proposed approaches to reduce the query patent by extracting a set of key terms from it. Different participating teams experimented with term distribution analysis in a language modeling framework and employed the document structure of the patent documents in various ways [83]. Here, we only discuss in detail the two best performing approaches in the CLEF-IP 2010. Lopez et al. [63] constructed a small corpus by exploiting the citation structure and IPC metadata. They then performed retrieval over this initial corpus. In [70] Magdy et al. generated the query out of the most frequent uni-grams and bi-grams. In this work the effect of using bi-grams in query generation was studied but the improvement was not significant. This is perhaps because of the unusual vocabulary usage in the patent domain.

So far, one of the most comprehensive descriptions of the problems and possible solutions for prior art search is presented by Magdy and Lopez [72]. The authors showed that the best performing run of CLEF-IP 2010 [63] used citations extracted by training a Conditional Random Field (CRF). The second best run [70] used a list of citations extracted from the patent numbers within the description field of patent queries. They also showed that the best run employed sophisticated retrieval methods using two complementary indices, one constructed by extracting terms from the patent collection and the other built from external resources such as Wikipedia. They compared these approaches and concluded that the second best run achieves a statistically indistinguishable performance compared to the best run when initial citations are provided with the query patent.

Many CLEF-IP and NTCIR participants used classification information as an extra feature besides the content of the patent. Thus, a different range of methods for combining text content and classification information were proposed. A standard way of combining the classification information is to consider it as a metadata and use it to filter the search results [41, 44, 73, 95, 99, 102]. This helps to filter out classifications that are too general or not related to the subject area of the query patent. Conclusive results are reported with respect to the usefulness of filtering using classification information. In [37] the authors integrate IPC codes into a probabilistic retrieval model, employing the IPC codes for estimating the document prior. A different usage of IPC classification has been performed in [31]. They used the classification information to extract query terms from triples specific to an IPC class. To do this, they used LCS software [59] which builds class profiles representing the term distribution (word and depen-

[2]See http://ifs.tuwien.ac.at/ clef-ip/

dency triples) per IPC class. They created a sub-corpus per query document that contains documents with at least one IPC class in common with the query document. Classification information has been successfully used by [91] in a different manner. They used classification information to partition the collection into different subject areas and with this partitioning they simulate a federated search for patent documents.

2.6.2 Leveraging Knowledge Bases and Using Proximity Heuristics

Previous research [63, 69] tackled the term mismatch problem in patent retrieval by first forming a keyword query from the query patent based on the frequency information. The initial query is then expanded using a knowledge base such as Wikipedia or WordNet, exploiting this enhanced query to disambiguate the occurrences of query terms. The use of external resources has shown to be more effective compared than the use of the initial query and pseudo relevance feedback (PRF). In fact, the retrieval effectiveness of PRF in patent retrieval has been shown to be disappointing mainly due to the low MAP of the initial rank list [38].

Patent examiners use term proximity heuristics in their searches using the Boolean retrieval model in order to reward a document where the matched query terms occur close to each other. Two forms of adjacency operators are used in Boolean retrieval to address proximity: the "ADJn" operator, which searches for terms within a window of n words in the order specified, and the "NEARn" operator, which searches for the terms within a window of n words, in whatever order. This usage shows that proximity information plays an important role in patent searching.

Lv and Zhai's works on positional language model and positional relevance model [66, 67] capture passage level evidence in a "soft" way by modeling proximity information via density functions. Their experiments confirmed that using density kernel functions to model the proximity information works better than applying a "hard" boundary of passages. Proximity information has shown to be useful in different IR tasks such as, for example, opinion mining [39], where authors investigated proximity information for capturing the opinion density at each point in the document.

Term position and proximity cues have mostly been ignored in previous work in patent retrieval. Recently, Ganguly et al.'s work captured term positions and proximity evidences indirectly through the use of passages [38]. Their goal is

to remove non-useful context that is not related to the focus of the query. They hypothesized that the removal of segments most dissimilar to the pseudo feedback documents can increase the precision of retrieval by removing non-useful context. To this end, they decomposed a patent application into constituent text segments and computed the Language Modeling (LM) similarities by calculating the probability of generating each text segment from the top ranked documents. They then reduced the patent query by removing the least similar segments from the query. This work proposes an innovative usage of feedback documents employing them for query reduction. This is in contrast to the traditional approaches that employ feedback documents for query expansion.

A different approach has been proposed by [31] that rewrites the query using Natural Language Processing (NLP) techniques. They extracted textual relations as triple dependencies from the title, abstract and the first 400 words of the description field to enhance the query. Such dependencies are representations of grammatical relations between words in a sentence. They observed that adding triples to the query did not improve MAP scores, in comparison to a bag-of-word baseline but had a positive effect on recall scores.

Another recent study on improving retrievability of patent documents [9] combined term proximity heuristics with other features to select good query expansion terms in the context of PRF. In this work different distance functions were considered from different windows surrounding query term occurrences. They reported an increase in terms of retrievability [7] of individual patents using proximity heuristics compared to standard PRF. However, they did not evaluate directly the performance of their approach in terms of retrieval effectiveness.

A different approach is introduced by [16] which addresses the patent retrieval as an XML retrieval task. The authors encapsulated proximity information by introducing flexible constraints (*near* and *below*) on the document structure which produce a numerical score based on tag positions in the XML structure of patent documents. They calculate the similarity of a document to a query by taking advantage of the XML structure of patent documents together with document content. They showed that their approach achieved high recall and high precision by employing structure-based constraints, as opposed to most of the existing patent retrieval approaches which have a good recall but suffer from low precision.

2.6.3 Citation Analysis

We now review the prior research that takes advantage of citation information, in particular, for improving the query representation through citation link analysis, and for patent citation/partner recommendation.

In [35], the author applied Page Rank algorithm [14] on a graph created based on the citation link structure of patent documents. He developed two distinct methods for measuring the influence of a patent document on the citation graph. In the first method he calculated the Page Rank score for each document by considering a graph structure composed of all documents in the collection. This method is not specific to the query submitted to the system. In the second method, he computed the Page Rank score for a query-specific citation graph, which is composed of the top-k documents initially retrieved for a given query topic and their cited documents. His experimental results on the NTCIR-6 test collection demonstrated that query-specific Page Rank score is more effective than traditional Page Rank score.

Lopez and Romary used references in a patent document as a starting point for prior art search [63]. They showed that extracting patent references using regular expression patterns resulted in missing at least 40% of references. In order to increase the accuracy of the extraction module, they identified patent reference blocks in the text of the patent using a Linear Chain CRF (Conditional Random Field) model. The reference block is then parsed to obtain a set of bibliographical attributes. They also used online bibliographical services to enrich the identified references. In order to extract characterizing key terms from a document to formulate a synthetic query, they extracted candidate phrases up to 5-grams from the text of the patent documents. They estimated the potential of each phrase to serve as a key term with a bagged decision tree. This model is trained on the key terms annotated by authors and readers from a set of training documents.

It is worth mentioning that in the prior art search task of CLEF-IP, citation information of query patents (topics in the test set) was removed and used for building the relevance judgement (ground truth). However, references to cited patent documents in the text of the query patent were not removed; as a result, the usage of these references in the text of the query patent was not recommended by organizers – unless participants explicitly mention such usage.

Recently, a few researchers [78, 96, 105, 109] studied the heterogeneous network of US patents derived from interacting patent companies and inventors to perform link analysis and prediction of the network structure. They leveraged the relation information among different types of objects on the network

in addition to the textual content of patents to mine the network.

The work reported in [105] studied patent collaboration patterns on an enterprise social network for recommending patent partners. They used a ranking factor graph model to predict future collaborators according to a user's profile and provide a recommendation list. They found that factors like, complementary research interests, and geographical proximity have positive effects in forming collaborations among inventors. Patent collaboration finding can also be seen as an instance of expert finding, similar to the problem of paper reviewers recommendation [76] and [97].

Recent work [96], studied a heterogeneous network of patents including different type of objects such as companies, inventors and the technical content of patent documents. They used topic modeling to discover latent topics associated with each objects on the network. After associating each object with a topic distribution, they identified the topical evolution of such objects on the patent network using temporal information and provided a variety of micro level analytics to simplify the decision making of the user. For example, their approach can provide a co-ranking of multiple objects such as companies and inventors in addition to patent documents on the patent network. Also, their system is able to identify active competitor companies based on their technology development trends.

Different work [78] used bibliographical attributes and citation links on a heterogeneous patent network for citation recommendation. This network is comprised of multi-typed objects including patent textual contents and patent contexts such as patent classifications, assignees, and inventors. The patents in the network are related if one patent is citing another patent or if they share bibliographical attributes. They used a supervised ranking algorithm, RankSVM, to provide a rank list of citations for a query patent application. According to their experiments, the citation information is useful and increases the accuracy of the system while the textual similarity-based approach increases the recall performance.

Other related work [109] identifies competitors of a given company by learning across multiple heterogeneous networks. They studied competitive relationship patterns on a company network, derived from a patent dataset and augmented by social networking information extracted from Twitter. They used topic modeling and built the topic model of each company, associating each company to a topic distribution. Their intuition is that entities with similar topic distributions are more likely to be competitors. They modeled the competitive relationship as a latent topic and used a factor graph model to infer the competitive label of each relationship among companies on the network. Experimental

results showed that their model was able to extract complementary competition patterns over these two sources, namely, the patent data set and the social network of Twitter.

2.6.4 Evaluation Metrics for Patent Retrieval

As mentioned earlier, in the beginning of Section 2.6, prior art search is a recall oriented application. In [68], Magdy and Jones showed that Mean Average Precision (MAP) can be a misleading metric for evaluating the performance of patent prior art search because of its inherent characteristic of favoring precision over recall. To address this problem, the authors proposed a metric called Patent Retrieval Evaluation Score (PRES) which takes into account the system recall and the user's search effort. We will use this metric for evaluating the retrieval effectiveness of our proposed methods later on throughout this thesis. Therefore, we explain this metric in detail in this section.

PRES is a modification over one of the well known IR evaluation metric called Normalized recall (R_{norm}) [89, 101]. R_{norm} measures the effectiveness in ranking documents relative to the best and worst ranking case, where the best ranking case is the retrieval of all relevant documents at the top of the list, and the worst case is the retrieval of all relevant documents only after retrieving the full collection. R_{norm} is calculated as the area between the actual and worst cases divided by the area between the best and worst cases. Normalized recall is greater when relevant documents are retrieved earlier in the rank list thus it can be seen as a good representative measure for recall-oriented applications. However, the disadvantage of normalized recall is related to the fact that it requires ranking the full collection which will not be feasible for very large collections.

In order to address this problem, authors [68] proposed a modification for the calculation of R_{norm}. They suggested an approximation of the worst case scenario by considering any relevant document not retrieved in the top $N_{\max}$ to be ranked at the end of the collection. The new assumption for the worst case scenario is to retrieve all the relevant documents just after the maximum number of documents to be checked by the user, denoted by $N_{\max}$. PRES uses this new assumption for the worst case scenario. The following equation shows how PRES is calculated.

$$PRES = 1 - \frac{\frac{\sum r_i}{n} - \frac{n+1}{2}}{N_{\max}} \tag{2.9}$$

where $N_{\max}$ is the number of documents to be checked by the user (cut-off value),

n is the number of relevant documents, and $\sum r_i$ is the summation of ranks of relevant documents, which is shown in the following equation:

$$\sum r_i = \sum_{i=1}^{nR} r_i + nR(N_{\max} + n) - \frac{nR(nR-1)}{2} \tag{2.10}$$

where R denotes the recall value defined as the number of relevant and retrieved documents in the first $N_{\max}$ documents.

2.7 Conclusions

This chapter reviewed the past work in IR and in patent information retrieval, in particular. This chapter provided background on patent query estimation and evaluation metrics related to patent prior art search. After reviewing different approaches related to patent information retrieval, we think some aspects of patent information retrieval require further research. We think one of the aspects that needs to be studied further is related to the way the textual sections of a query patent are used for query estimation. We also think proximity information in the form of phrases or in the form of vicinity of expansion terms from query terms can help improve the retrieval effectiveness. Besides this, other metadata associated with patent documents such as classification tags, citation links, and temporal features can be used to address the term mismatch problem. It is worth investigating whether these resources provide complementary vocabulary to each other and if utilizing these sources can help improve retrieval performance in patent information retrieval. We pursue our research following these directions. We discuss the methods and models developed from our considerations of these topics in the following chapters of this thesis. In the next chapter we discuss the experimental methodology that we will follow in the rest of this thesis.

3

Experimental Setup

"The only source of knowledge is experience."
– Albert Einstein

3.1 Introduction

In this chapter we describe the evaluation methodology employed in IR in general and in patent prior art search in particular to evaluate retrieval effectiveness. In order to evaluate the effectiveness of an information retrieval method in a standard setting, it is necessary to have a benchmark dataset. This dataset is composed of a document collection, a test set of query topics, and a corresponding set of relevance judgements to each query. The second component necessary in evaluating the effectiveness of an IR system is an evaluation measure.

This chapter is organized as follows: we first explain the benchmark dataset in Section 3.2 and we describe the test collections that will be used in later experiments. We then discuss the evaluation metrics used for evaluating the performance of the methods proposed in this thesis in Section 3.3. The notions introduced in this chapter will be used in various places throughout this thesis.

3.2 Benchmark Dataset

In this section we present a detailed description of document collections, a test set of queries/topics and a set of relevance judgements/assessments.

3.2.1 Test Collections

We conducted our experiments using datasets from two years of the CLEF Intellectual Property (CLEF-IP) task, namely CLEF-IP 2010 and CLEF-IP 2011 datasets. CLEF-IP documents are presented in XML format with annotations about different textual fields and metadata such as inventors, assignees, and priority dates. These documents are obtained from European Patent Office and have mixed content in English, German and French. Table 3.1 reports the number of documents and queries of CLEF-IP test collections.

Table 3.1. Specifications of test collections.

Collection Name	CLEF-IP 2010	CLEF-IP 2011
# documents	2.6 Millions	3 Millions
# queries in the test set	1348	1351

The Terrier toolkit[1] is used to index the collection. Terrier is an information retrieval system written in Java which implements the state-of-the-art indexing and retrieval functionalities, and is developed at the School of Computing Science, at University of Glasgow.

During the indexing, we used the default stemming method (Porter stemmer) and the default stop-word list of Terrier. We worked with the English subset of both collections. We considered the textual fields of entire patent documents while indexing. We then removed patent-specific stop-words such as “device” and “method”. The list of patent specific stop-words is built as follows. We calculated document frequencies for each term in the collection. We then selected terms with top 10% highest document frequency and considered them as patent specific stop-words. The threshold value of 10% is experimentally set. In addition to that, we removed all the formulas and numeric references to improve the retrieval effectiveness.

Each patent document in the dataset has different kind-codes (versions) which are used to denote its level of publication (e.g., first publication, second publication, or corrected publication). We merged the documents related to the same patent by taking each field from the latest publication and created a virtual document according to the guidelines[2].

[1] Available at `http://ir.dcs.gla.ac.uk/terrier/`

[2] Available at `http://www.ifs.tuwien.ac.at/~clef-ip/download/2009/topics/finalset/CLEFIP09_TopicGuidelines.pdf`

3.2.2 Test sets

In the training and test sets of the CLEF-IP data sets, a topic is represented by a full patent application rather than a keyword query as used in standard adhoc IR, hereafter referred to as a query patent. An example is shown in Figure 3.1.

```
<topic>
<num>PACt-1</num>
<narr>Find all patents in the collection that potentially invalidate
patent application EP-1752549-A1.</narr>
<file>PACt-1_EP-1752549-A1.xml</file>
</topic>
```

Figure 3.1. A query topic selected from the training set of CLEF-IP 2010.

A query patent is a structured document which is composed of the following fields: *title*, *abstract*, *description*, and *claims*. The claims field comprises of multiple claims that are numbered. A claim which does not refer to any other claim is called an *independent claim* while others are called *dependent claims* [64]. The independent items in the claims field of the patent comprise the kernel of the technical innovation of the patent. Among the claims the most important one is the first independent claim (the first item in the claims), which represents the essence of the technology of the patent document. The other parts of the patent document illustrate the reason, background, implementation and advantages, of the invention being described [65].

An example of a patent application is shown in Figure 3.2. According to this example, claim 1 is an independent claim while claims 2-5 are dependent claims.

One metadata field associated with patent documents is the International Patent Classification (IPC). The IPC system is mainly designed to help classify patent documents. It provides a hierarchical categorization over different technological fields such as computer science, electronics, mechanics, and biochemistry. These IPC classes can be seen as conceptual tags assigned to the patent documents. They categorize the content of a patent document and describe the field of technology that a patent document belongs to. Note that in general there are about 70,000 classes in the most fine grained level of the IPC hierarchy[3]. Table 3.2 shows some statistics about IPC classes in the CLEF-IP test

[3]According to `http://www.wipo.int/classifications/ipc/en/general/statistics.html`

Application Number	EP-1832953-A2
Title	Method and apparatus for managing a peer-to-peer collaboration system
IPC Classes	G06F1/00, G06F15/00, G06F21/00, G06F21/24, H04L29/06, H04L29/08
Abstract	Users and devices in a peer-to-peer collaboration system can join a management domain in which members are administered as a group by a centralized management server operated by an enterprise. In response to a administrator request to join the management domain, the user downloads an injectible identity file containing a definition of the managed user/device into the user system. The user then joins the managed domain by associating the injected identity with their actual identity. Once a user or device is part of a management domain, that user or device receives license rights and policy restrictions that are associated with the domain. In return, the management server interacts with the individual peer-to-peer collaboration systems to enable the enterprise to monitor the enterprise to monitor the usage of, and control the behavior of, that specific identity within the peer-to-peer collaboration system.
Description	This invention relates to peer-to-peer collaboration systems and, in particular to methods and apparatus for gathering usage statistics for managing such systems. New collaboration models have been developed which operate in a “peer-to-peer” fashion without the intervention of a central authority. One of these latter models is built upon direct connections between users in a shared private “space”. In accordance with this model, users can be invited into, enter and leave a shared space during an ongoing collaboration session between other users. Each user has an application program called an “activity”, which is operable in his or her personal computer system, communication appliance or other network-capable device which generates a shared “space” in that user’s computer. The activity responds to user interactions within the shared space by generating data change requests, called “deltas”. The activity also has a data-change engine component that maintains a local data copy and performs the changes to the data requested by the deltas. The deltas are distributed from one user to another over a network, such as the Internet, by a dynamics manager component. When the deltas are received by another user activity in the shared space, the local data copy maintained by that activity is also updated...
Claims	**1.** A method for managing a peer-to-peer collaboration system in which users having identities are directly connected to each other in a shared private space by client software operating in devices and wherein the users communicate with a management server using the client software, the method comprising: (a) sending a request from the management server to the user to become a managed entity; (b) downloading from the management server to the client software a definition file containing a definition of the managed entity; and (c) associating information in the definition file with user identities and device in the client software in order to create a managed entity. **2.** The method of claim 1 wherein the managed entity is a managed user and the definition information file is an injectible identity file. **3.** The method of claim 1 wherein the managed entity is a managed device and the definition information file is a device information file. **4.** The method of claim 3 wherein the device information file is a Windows REG file. **5.** The method of claim 1 further comprising: (d) sending at least one license file from the management server to the managed user; and (e) in response to information in the license file, enabling at least one function in the client software....

Figure 3.2. An example query topic in CLEF-IP 2011 test set (an excerpt)

collections. Table 3.3 represents more fine grained information about the field of technology of test topics. As shown in Table 3.3, IPC divides technology into eight sections.

Table 3.2. IPC classes in CLEF-IP collections.

Collection Name	CLEF-IP 2010	CLEF-IP 2011
# distinct IPC classes	62183	63495
Avg # IPC classes per document	3.4	3.9

Table 3.3. IPC section distribution over English test set of CLEF-IP 2010 and CLEF-IP 2011.

Category	Description	# of topics in CLEF-IP 2010	# of topics in CLEF-IP 2011
A	Human Necessities	154	250
B	Performing Operations and Transporting	307	213
C	Chemistry and Metallurgy	255	150
D	Textiles and Papers	10	23
E	Fixed Constructions	7	18
F	Mechanical Engineering, Heating, Weapons, and Blasting	90	143
G	Physics	289	263
H	Electricity	236	291
	total number of queries	1348	1351

3.2.3 Assessment

IR evaluation requires ground truth data to evaluate the performance of retrieval systems. The common procedure is to collect manual assessment from voluntary assessors for all evaluation topics. However, this approach is labor-intensive and finding voluntary assessors is not easy. Finding voluntary assessors is even more difficult in patent prior art search as expert knowledge is required. An alternative assessment approach is to use patent citations which provide partial ground truth and are easily accessible [90]. The organizers of CLEF-IP used this approach and built the query relevance judgements (qrels) for the CLEF-IP test collection from citations.

Citations are extracted from several sources:

- Disclosed by applicant: some patent offices (e.g. United States Patents and Trademark Office (USPTO)) require applicants to disclose all known relevant publications when applying for a patent.

- Patent examiner search report: each patent examiner does a prior art search to judge the novelty of a patent application.

- Opposition procedure: this happens when a company monitors granted patents by its competitors and files an opposition procedure to claim that a granted patent belonging to its competitor is not actually novel.

Figure 3.3 shows the list of qrels for a topic selected from the training set of CLEF-IP 2010. Scale of relevancy denotes the source of the extracted citations. The scale of relevancy 1 denotes that the citation is either disclosed by the patent applicant or mentioned in the search report (provided by the patent examiner). Relevance scale 2 indicates that the citation is obtained through an opposition procedure.

Topic number	Relevant document	Scale of relevancy
PACt-1	EP-1473371-B1	1
PACt-1	EP-1473371-A3	1
PACt-1	EP-1473371-A2	2
PACt-1	EP-1356126-B1	2
PACt-1	EP-1356126-A2	1
PACt-1	EP-0484904-B1	1
PACt-1	EP-0484904-A3	1

Figure 3.3. The relevance judgements for the query topic "PACt-1" selected from the training set of CLEF-IP 2010.

3.3 Evaluation

We now explain some of the common metrics in IR that are used to evaluate the performance of IR systems. Using these metrics allows us to compare the performance of our methods with the state of the art patent retrieval systems.

Precision. The fraction of retrieved documents in response to a query that are relevant [24].

$$Precision = \frac{\texttt{\{relevant documents\}} \bigcap \texttt{\{retrieved documents\}}}{\texttt{\{retrieved documents\}}} \tag{3.1}$$

Recall. The fraction of relevant documents available in response to a query that are actually retrieved [24].

$$Recall = \frac{\texttt{\{relevant documents\}} \bigcap \texttt{\{retrieved documents\}}}{\texttt{\{relevant documents\}}} \tag{3.2}$$

Average Precision (AP). The Precision score is calculated at each position in the rank list where a relevant document is retrieved, and then these precision scores are averaged [15].

$$AP_i = \frac{1}{|R_i|} \sum_{r \in R_i} P@\texttt{rank}(q_i, r) \tag{3.3}$$

where AP_i denotes the average precision for the ith query, R denotes a ranked list, r denotes a relevant document, and P denotes the precision.

Mean Average Precision (MAP). MAP is the mean of AP_i over all topics in topic set Q.

$$MAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q} \tag{3.4}$$

this averaging over all the queries in the test set is performed to allow the reporting of the performance of a retrieval system over the full test set.

PRES. Patent Retrieval Evaluation Score (PRES) [68] combines recall and the user's search effort in one single score. PRES metric is specially designed for recall-oriented applications such as patent retrieval. We previously explained this metric, please find more detailed information in Section 2.6.4.

We used the relevance judgement of the test topics in English provided by CLEF-IP for evaluation purposes.

3.3.1 Best Official Results of CLEF-IP Challenge

Table 3.4 shows the performance of the best official results of CLEF-IP 2010 and CLEF-IP 2011 [82, 83] on the English subset of the test set. We will compare our

methods with these approaches throughout the thesis. We took the best official results of CLEF-IP 2010 from the evaluation report[4].

For CLEF-IP 2011 we took the best official results from the evaluation summary released to participants. Note that PRES values were not reported for the best results of CLEF-IP 2011.

Table 3.4. The performance of the best official results on English Test Set.

Official best results of CLEF-IP 2010				
Method	Run description	MAP	recall	PRES
humb	rank 1	0.2264	0.6946	0.6149
dcu	rank 2	0.1807	0.6160	0.5167
Official best results of CLEF-IP 2011				
Method	Run description	MAP	recall	PRES
nijm	rank 1	0.0582	0.6303	NA
hyder	rank 2	0.0593	0.5713	NA

3.3.2 Statistical Significance

In the remainder of this thesis, in our experiments, we compare the proposed models with some baseline systems or we compare two different versions of a single model. In order to check for statistical significant difference between the two runs, we use the randomization (permutation) test with a confidence level of 0.05. This test has been shown to be more reliable than Wilcoxon and t-test [94].

3.4 Conclusions

In this chapter we introduced the evaluation methodology including the CLEF-IP 2010 and CLEF-IP 2011 test collections, the set of training and test queries, relevance assessments, evaluation metrics, and significance tests.

CLEF-IP test collections are standard benchmark collections for patent retrieval. A topic in the CLEF-IP test set is a full patent application instead of a keyword query.

[4]http://www.ifs.tuwien.ac.at/ clef-ip/pubs/CLEF-IP-2010-IRF-TR-2010-00003.pdf

The evaluation methodology described in this chapter will be used for evaluating the retrieval effectiveness of our proposed methods in the next chapters of this thesis.

4

Query Reduction

"A little knowledge that acts is worth infinitely more than much knowledge that is idle."
– Kahlil Gibran

4.1 Introduction

A query patent is a full patent application composed of hundreds of terms which does not represent a single focused information need. Our goal is to reduce the query patent and select representative terms to form an effective query. In this context, query A is more effective than query B if it can better distinguish relevant patents from non relevant patents. The effectiveness of a query is evaluated according to the performance of the final rank list. We implemented three different approaches to estimate the query model of a patent document. The first two approaches are based on weighted log-likelihood [75], and the third approach is based on Parsimonious language modeling [51]. The goal of these approaches is to select the most informative terms for representing the topic of the query patent. These approaches will be discussed in more detail in this chapter.

We utilize the structural information of a patent document in our model by estimating a query model for each field separately. A patent document in the CLEF-IP 2010 collection contains the following fields: the title (*ttl*), the abstract (*abs*), the description (*desc*), and the claims (*clm*). Our aim is to investigate and compare the quality of extracted terms according to the query model of each field. In an attempt to take into account the full structure of the document, we also explore merging rank lists generated from different fields.

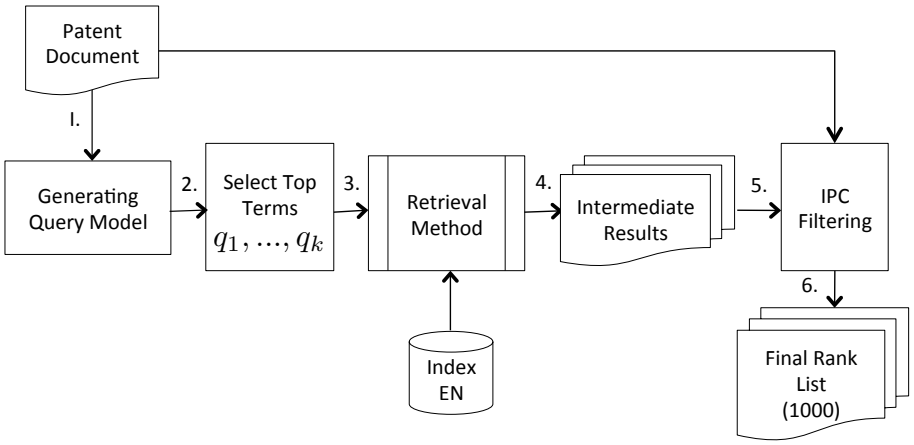


Figure 4.1. Overall architecture of the proposed system.

4.2 System Architecture

Figure 4.1 shows the overall architecture of our retrieval system for prior art search. In the first step we estimate a query model from the query patent document. In the second step we formulate a query by selecting the top k terms from the query model. Next, we retrieve documents relevant to the query using a retrieval method (step 3) and return an intermediate rank list (step 4). We filter this rank list by excluding documents which do not have any IPC class in common with the query patent document (step 5) and generate a final rank list (step 6).

In the next sections we focus on the query reduction problem and propose three methods for query model estimation.

4.3 Query Model based on Weighted Log-Likelihood

In our first approach we build a query model (denoted θ_{Q_f}) for the field f of the patent document, where f belongs to {*title, abstract, description, claims*}. We estimate the query model θ_{Q_f} by calculating the relative frequencies for terms in the field f of the query document. To create a better representation, we smooth the θ_{Q_f} estimate with the topic model of the relevant cluster of documents. This cluster consists of documents with at least one IPC class in common with the query document (denoted *RIPC*). The intuition is that patent documents with similar IPC classes have similar topics [42]. This smoothing of the parameters

away from their maximum likelihood estimates helps us to exploit the knowledge embedded in the IPC hierarchy. Because of the smoothing, non zero probabilities are assigned to words which are associated with the topic of a document and are not mentioned in the document itself. This can be seen as expanding the document model with the IPC metadata. The query model θ_{Q_f} is estimated as follows:

$$P(w|\theta_{Q_f}) = \lambda \frac{f(w,Q_f)}{|Q_f|} + \frac{(1-\lambda)}{N} \sum_{d \in RIPC} \frac{f(w,D_f)}{|D_f|} \tag{4.1}$$

where $f(w,Q_f)$ denotes the term frequency of the word w in the field f of the patent document, $|Q_f|$ is the length of the field f of the patent document, N denotes the size of the relevant cluster *RIPC*, and λ denotes the smoothing parameter. In order to estimate a query model for the patent in question, it is necessary to highlight words from the term distribution of θ_{Q_f} which are rare in the collection. To this end, we weight term probabilities in θ_{Q_f} using the following formula:

$$P(w|LLQM_f) = Z_w \, P(w|\theta_{Q_f}) \, \log \frac{P(w|\theta_{Q_f})}{P(w|\theta_{C_f})} \tag{4.2}$$

where $P(w|\theta_{C_f})$ shows the probability of a word in the collection and is estimated as follows:

$$P(w|\theta_{C_f}) = \frac{f(w,C_f)}{\sum_{D \in C} |D_f|} \tag{4.3}$$

where $f(w,C_f)$ denotes the collection term frequency for the field f of the query patent and $Z_w = \frac{1}{\sum_{w \in V} P(w|\theta_{Q_f}) \log \frac{P(w|\theta_{Q_f})}{P(w|\theta_{C_f})}}$ is a normalization factor. What we have in the denominator is the Kullback-Leibler divergence between $P(w|\theta_{Q_f})$ and $P(w|\theta_{C_f})$, as it is summed over all the terms in the vocabulary. Thus, the normalization factor can be written as $Z_w = \frac{1}{D_{KL}(P(w|\theta_{Q_f})||P(w|\theta_{C_f}))}$. We refer to this model as the Log-Likelihood Query Model ($LLQM_f$). This model is similar to the approach introduced in [75].

This measure quantifies the similarity of the query document with the topical model of relevance and the dissimilarity between the query document and the collection model. Terms with high divergence are good indicators of the patent document and show the specific terminology of the patent document.

4.4 Cluster-based Query Modeling

In the second approach, we attempt to incorporate the knowledge of the hierarchical classifications of IPC [1] into our model. We estimate a slightly different formulation of the query model, referred to as Cluster Based Query Modeling ($CBQM_f$), by weighting term probabilities in θ_{Q_f} by their relative information in the cluster language model θ_{Cl_f} and the collection language model θ_{C_f}. This model assigns a high score to query terms which are similar to the cluster language model but dissimilar to the collection language model. We base this estimate on the divergence between θ_{Q_f} and the cluster language model, measuring this divergence by determining the log-likelihood ratio between θ_{Q_f} and θ_{Cl_f}, divided by θ_{C_f}. This formulation gives another way of estimating the query model based on the relevant cluster derived from IPC classes:

$$P(w|CBQM_f) = Z_w \, P(w|\theta_{Q_f}) \, \log \frac{P(w|\theta_{Cl_f})}{P(w|\theta_{C_f})} \tag{4.4}$$

4.5 Parsimonious Query Modeling

In our third approach we estimate a query model that differentiates the language used by the query patent from the collection model. Following the work of Hiemstra *et al.* [51], we estimate the topic of the query patent using parsimonious language modeling, by concentrating the probability mass on terms that are indicative of the topic of the query patent but are dissimilar from the collection model. We use the Expectation-Maximization (EM) algorithm for estimating the query model of different fields of a patent document. The Parsimonious Query Model (PQM_f) is estimated according to the following iterative algorithm:

E-step:

$$e_w = f(w, Q_f) \, \frac{\lambda P(w|PQM_f)}{(1-\lambda) \, P(w|C_f) + \lambda \, P(w|PQM_f)} \tag{4.5}$$

M-step:

$$P(w|PQM_f) = \frac{e_w}{\sum_w e_w}, \text{ i.e. normalize the model} \tag{4.6}$$

[1] Available at `http://www.wipo.int/classifications/ipc/en/general/preface.html`

where $P(w|C_f)$ is the maximum likelihood estimate for the collection and is calculated according to Equation 4.3. The initial value for $P(w|PQM_f)$ is based on the maximum likelihood estimate for the query as in Equation 4.1, skipping the smoothing step. The advantage of this estimation model is that it discards field-specific stop-words automatically. This is because we estimate the query model for each field separately. For example, for the abstract field the set of words "system", "device", "apparatus", and "invention" are identified as stop-words.

4.6 Experiments

The proposed models have two parameters: the field f of query patent used for building the query model, and the query length parameter k which denotes the maximum number of selected terms to be used. In the rest of this section we study the effect of these parameters on the effectiveness of the final rank lists. We carried out our experiments with BM25 retrieval function and the smoothing parameter λ in $LLQM_f$ and PQM_f was experimentally set to 0.9.

4.6.1 Effect of Query Length and Field

Tables 4.1 – 4.4 show the results of our experiments with the query estimation approaches introduced previously. To study the effect of the query length we vary the number of query terms selected from different fields of the query document. Results are reported over the training set of CLEF-IP 2010. Note that the query model is built for each field separately. However, in the retrieval step, all the fields of the patent documents are considered for similarity score calculation. In other words, the query model built from field f is applied on all the fields of the patent documents. In the tables, for the sake of readability, the field f used for query estimation is denoted within parentheses.

The results of all four tables show that increasing the query length improves the evaluation scores. However, when the query length exceeds some limit, adding more candidate query terms does not further improve the performance. This is true for all the three query estimation methods. Based on these experiments we limit the length of the generated queries from description, claims, abstract, and title to 100, 100, 50, 10, respectively.

Table 4.5 reports the performance of the three term selection techniques on the training set over different fields, using the optimized query length. Furthermore, Table 4.5 shows the effect of merging multiple search results of the differ-

Table 4.1. Evaluation scores of the different query estimation methods using the description field on the training set for the English subset.

PQM(desc)	25	50	75	100	125	150
MAP	0.08	0.09	0.09	**0.10**	0.10	0.09
Recall	0.56	0.57	0.59	**0.59**	0.58	0.57
CBQM(desc)	25	50	75	100	125	150
MAP	0.08	0.09	0.10	**0.11**	0.10	0.09
Recall	0.58	0.59	0.60	**0.60**	0.59	0.59
LLQM(desc)	25	50	75	100	125	150
MAP	0.08	0.08	0.11	**0.12**	0.12	0.11
Recall	0.59	0.62	0.62	**0.63**	0.61	0.60

Table 4.2. Evaluation scores of the different query estimation methods using the claims field on the training set for the English subset.

PQM(clm)	25	50	75	100	125	150
MAP	0.04	0.05	0.06	**0.07**	0.07	0.07
Recall	0.48	0.50	0.52	**0.54**	0.53	0.52
CBQM(clm)	25	50	75	100	125	150
MAP	0.05	0.06	0.06	**0.07**	0.06	0.06
Recall	0.49	0.52	0.53	**0.56**	0.54	0.52
LLQM(clm)	25	50	75	100	125	150
MAP	0.06	0.08	0.10	**0.10**	0.09	0.09
Recall	0.51	0.53	0.56	**0.57**	0.56	0.55

ent algorithms using CombSUM and CombMNZ [93]. The CombSUM combination method calculates the sum of the set of relative ranks, or similarity values, retrieved by multiple search runs. CombMNZ, performs similar to CombSUM by calculating the average of the set of similarity values and it also provides higher weights to documents retrieved by multiple retrieval methods.

Experiments show that extracting terms from the description field has the best performance over all other fields. The reason for this can be related to the technical language used in description as opposed to the legal jargon which is the characteristic of the claims field. We believe the short length of titles is

Table 4.3. Evaluation scores of the different query estimation methods using the abstract field on the training set for the English subset.

PQM(abs)	10	20	30	40	50
MAP	0.05	0.05	0.06	0.06	**0.07**
Recall	0.47	0.48	0.50	0.52	**0.54**
CBQM(abs)	10	20	30	40	50
MAP	0.05	0.05	0.06	0.06	**0.07**
Recall	0.48	0.52	0.54	0.56	**0.56**
LLQM(abs)	10	20	30	40	50
MAP	0.05	0.05	0.06	0.07	**0.07**
Recall	0.50	0.52	0.54	0.55	**0.56**

Table 4.4. Evaluation scores of the different query estimation methods using the title field on the training set for the English subset.

PQM(tit)	5	10
MAP	0.03	**0.03**
Recall	0.48	**0.50**
CBQM(tit)	5	10
MAP	0.04	**0.04**
Recall	0.52	**0.53**
LLQM(tit)	5	10
MAP	0.04	**0.05**
Recall	0.52	**0.53**

the reason why selecting terms from them performs poorly when compared to other fields. Prior work [108] suggests that both the abstract and description use technical terminology, but our results show that using the abstract field is less effective. Further investigation is needed to understand why query terms extracted from the abstract field are not as effective as those extracted from the description.

Another observation is that $LLQM_f$ outperforms $CBQM_f$ and PQM_f in terms of both MAP and recall. The reason that $CBQM_f$ performed slightly worse than $LLQM_f$, is perhaps due to the fact that in $CBQM_f$ we consider all documents that have IPC classes in common with the query as feedback documents. This cluster of relevant documents is very big, therefore we lose the specific terms which are representative of the topic of the query document.

Our attempt to merge results of different fields using CombSUM and CombMNZ did not improve the performance over the best setting. Since similar results were found when building a single query by combining the selected query terms from different fields, we did not report the results.

4.6.2 Comparison with the CLEF-IP 2010 participants

We fix our two parameters for the estimation method of the query model, namely the query length and the query field, to the values which have been shown to achieve the best performance on the training set. Now we present our results with this setting on the test set. Our results on the training set show that $LLQM_f$ and $CBQM_f$ perform better than PQM_f. Thus we only present the results of these two approaches on the test set. If we would have submitted the results of $LLQM_f$, it would have been positioned among the top-3 for the prior art task in terms of recall and PRES. In terms of MAP it would have been placed at rank 4, while $CBQM_f$ would have been placed two ranks below $LLQM_f$.

Table 4.6 shows our position with respect to other CLEF-IP 2010 participants according to the official results on the English subset of the test set[2]. In our approaches, we did not look into citations proposed by applicants. Among the top ranked participants, only two other approaches, that is *dcu-nc* by Magdy and Jones [70] and *spq* by Alink et al. [1], were similar to ours in this respect. Our two approaches are shown in bold face.

Although previous work mainly uses claims for query formulation [58, 95], our results suggest that building queries from the description field can be more useful. This result is in agreement with [108], in which query generation for

[2]http://www.ifs.tuwien.ac.at/∼clef-ip/pubs/CLEF-IP-2010-IRF-TR-2010-00003.pdf

Table 4.5. Comparison of performance of the different query estimation methods using the different fields of a patent document.

Run	MAP	Recall
PQM(tit)	0.03	0.50
PQM(abs)	0.07	0.54
PQM(desc)	**0.10**	**0.59**
PQM(clm)	0.07	0.54
CombSUM(all)	0.05	0.55
CombMNZ(all)	0.04	0.54
CBQM(tit)	0.04	0.53
CBQM(abs)	0.07	0.56
CBQM(desc)	**0.11**	**0.60**
CBQM(clm)	0.07	0.56
CombSUM(all)	0.09	0.57
CombMNZ(all)	0.07	0.56
LLQM(tit)	0.05	0.53
LLQM(abs)	0.07	0.56
LLQM(desc)	**0.12**	**0.63**
LLQM(clm)	0.10	0.57
CombSUM(all)	0.09	0.57
CombMNZ(all)	0.08	0.56

Table 4.6. Prior art results for best runs in CLEF-IP 2010, ranked by PRES, using the large topic set for the English subset.

Run	MAP	Recall	PRES
humb[63]	0.2264	0.6946	0.6149
dcu-wc[70]	0.1807	0.616	0.5167
LLQM	**0.124**	**0.60**	**0.485**
dcu-nc[70]	0.1386	0.5886	0.483
CBQM	**0.124**	**0.589**	**0.477**
spq[1]	0.1108	0.5762	0.4626
bibtem[98]	0.1226	0.4869	0.3187

Field	MAP	Recall	PRES	Retrieval Method
Description	0.136	0.621	0.537	BM25
Claims	0.129	0.607	0.514	LM
Entire Patent	0.130	0.611	0.515	LM

Table 4.7. Results of CLEF-IP 2010 extracted from different fields.

US patents were explored, and the "background summary" field was shown to be the best source for extracting terms. Since the background summary in US patents uses a technical terminology for explaining the invention, it is considered equivalent to the description field in European patents.

4.6.3 Removing Patent Specific Stop-words

By removing patent specific stopwords, as explained in Section 3.2.1, we improved the performance of the system. The obtained results are presented in Table 4.7. These results are obtained using LLQM as query estimation model. These query models generated from different fields of the query patent will be used in the next chapters.

4.7 Conclusions

Prior art task is one of the most performed search tasks in the patent domain. The information need in this task is presented by a query document (a patent application). Therefore, converting the document into effective search queries is the necessary step for finding relevant documents.

In this work, we presented three query modeling methods for estimating the topic of the patent application. We integrate the structural information of a patent document and the IPC classification (for filtering) into our model. Our results suggest that the "description" is the best field for extracting terms for estimating query models and LLQM is the best query estimation model among the three.

Based on our experiments, combining different fields while estimating the query model or merging the results afterwards, achieves lower performance compared to that of the "description". In the next section we study the advan-

tage of using noun phrases for query expansion and other metadata associated with the query patent to improve the query model

5

Query Quality Prediction

"Absence of evidence is not evidence of absence."
– Carl Sagan

5.1 Introduction

In the previous chapter we presented an investigation into patent prior art search aimed at achieving maximum retrieval effectiveness when the user submits a full patent document as a query. To provide such functionality, we propose techniques to process patent documents and extract terms and key phrases in order to form a query to retrieve relevant documents from the patent corpus. The technique proposed in the previous chapter extracts single terms from the query patent using the KL-divergence between the query patent and the collection. In this chapter, we extend the previous approach by extracting key phrases with similar semantics to the query patent. Such phrases will be used to expand and disambiguate the initial uni-gram query, (estimated in Chapter 4). We then expand the original query using noun phrases and retrieve relevant documents from the patent corpus. We use both corpus statistics and linguistic heuristics for finding meaningful phrases. These two approaches are complementary to each other: the first approach extracts generic terms, favoring recall, while the second aims at finding a clear focus for the query by providing more specific phrases, thus increasing precision.

This chapter is organized as follows: we start by explaining our motivation to use noun phrases and to perform selective query expansion in Section 5.2. Then, we estimate different uni-gram query models in Section 5.3. We identify the set of all candidate key phrases for the query document d in Section 5.4. We

evaluate the significance of each candidate phrase by assigning a score between 0 and 1 to each phrase as shown in Section 5.5. We predict the effectiveness of the noun phrases in Section 5.6 and select the effective phrases to construct an expanded query. In the evaluation section (Section 5.7), we compare the quality of the expanded query to the uni-gram query by reporting the document retrieval results. Finally, Section 5.8 concludes the chapter.

5.2 Using Noun Phrases for Query Expansion

Our motivation to use noun phrases is originated by the fact that often technical terms in English are represented with more than one word. Mostly noun phrases are used, where a general word is modified by more specific words, for describing the topic of the invention – such as in "gambling or entertainment machine".

Our goal is to refine the original query by expanding it with selected key concepts (i.e., bi-grams or phrases) from the query patent using the global analysis of the patent collection. After performing the expansion for all the queries in the topic set and evaluating the mean average precision (MAP) over the topic set, we observed that the performance of the expanded rank list (using noun phrases) is not statistically different from the unexpanded rank list (using single terms (uni-grams)). After performing failure analysis on a per query basis, we detected a large variation in performance for different queries. We found that while indeed the expanded rank list improves the quality of results for many queries considerably, the quality of results is poor for some other queries. One of the reasons explaining the low retrieval performance is attributed to the topics for which the main aspect of the query is not considered during expansion. In such cases, only important concepts describing the partial aspects of the query are extracted. This observation shows that expansion using noun phrases was not consistently beneficial for all queries. This suggests that the decision about query expansion using concepts should be taken in a query dependent way.

In this chapter, we propose a method for distinguishing between queries and deciding when to selectively use the result of a refinement technique that is likely to improve the retrieval performance. Our goal is to find queries that have highly positive changes in query performance using refinement. To this end, we use query performance predictors (pre and post-retrieval) [4, 26, 48, 104] and patent-specific features in order to find highly performing queries in the expanded retrieval rank list. In order to decide when to use the result of the expanded list, we rely on a machine learning approach that tries to predict which

one of the two competing approaches will offer the best result for a given query.

This prediction is performed in a selective query refinement framework [4, 27, 47, 49, 111]. It is necessary for us to build a robust patent retrieval system which can be used in an operational setting. We aim at quantifying the performance of the queries in order to build a robust system which can invoke different retrieval strategies in a query dependent way according to the estimated performance of a query. To the best of our knowledge no previous work has used the query performance predictors in the patent domain.

In this chapter we explore extracting concepts which explicitly occur in the query patent itself. We also study extracting important concepts associated with the underlying information need of the query patent by building a relevance model through the process of query expansion: 1) via pseudo relevance feedback; 2) using sample relevant documents.

5.3 Establishing a Uni-gram Baseline

In this section we estimate the query model for a query patent in a language modeling framework. This estimation enables us to identify the importance of terms and assign weights to them accordingly. By modeling the term distribution of the query patent we get a detailed representation of the query patent which allows us to expand the query, and to refine the query model by considering relationships between terms. This approach is used to bridge the vocabulary gap between the underlying information need of the query patent and the collection.

In this section, we first describe how we create a language model Θ_Q for the query patent. We use the maximum likelihood estimate smoothed by the background language model, as expressed in Equation 5.1 to avoid sparseness issues.

$$P(t|\Theta_Q) = (1-\lambda) \cdot P_{ML}(t|D) + \lambda \cdot P_{ML}(t|C) \tag{5.1}$$

where maximum likelihood estimate P_{ML} is calculated as follows:

$$P_{ML}(t|D) = \frac{n(t,D)}{\sum_{t'} n(t',D)} \tag{5.2}$$

We introduce a uni-gram query model by estimating the importance of each term according to a weighted log-likelihood based approach as expressed below:

$$P(t|Q_{orig}) = Z_t \; P(t|\Theta_Q) \; \log\left(\frac{P(t|\Theta_Q)}{P(t|\Theta_C)}\right) \tag{5.3}$$

where $Z_t = 1/\sum_{t \in V} P(t|Q_{orig})$ is the normalization factor. This approach favors terms that have high similarity to the document language model Θ_Q and low similarity to the collection language model Θ_C. For the rest of this chapter Q_{orig} serves as our uni-gram baseline. Q_{orig} is equal to LLQM query model which was calculated in Chapter 4.

In order to model the query patent more precisely we need a source of additional knowledge about the information need. Patent documents are annotated with International Patent Classifications[1] (IPC). Such classes are language independent keywords assigned as metadata to the patent documents. They are categorizing the content of a patent document and reflecting the field of technology of a patent. These IPC classes resemble tags assigned to documents (henceforth referred to as *conceptual tags* in this chapter).

Our goal is to build a relevance model by employing documents that have at least one conceptual tag in common with the query topic. Each relevant document from this sample is assumed to serve as evidence towards the estimation of the relevance model. Note that the relevant samples are not part of the relevance information.

Our approach to construct the relevance model Θ_{IPC} is the following. First, we estimate the level of relevance of a document D with $P(D|\Theta_{IPC})$. Then the top-k terms with the highest probability $P(t|D)$ are picked and used to build Θ_{IPC}. The sample distribution $P(t|\Theta_{IPC})$ is calculated according to Equation 5.4. This sampling is dependent on the original query patent as it utilizes documents with similar conceptual tags to the query patent.

$$P(t|\Theta_{IPC}) = \sum_{D \in IPC} P(t|D) \cdot P(D|\Theta_{IPC}) \tag{5.4}$$

Now we explain how the level of relevance of a sample document D is estimated. We can not assume documents in the relevance set have equal importance. The reason is that documents in the relevance set can be multi-faceted and therefore not entirely relevant to the information need represented by the query patent. So we need to assign importance to the documents according to their level of relevance. We approximate the relevance of a sample document D, denoted by $P(D|\Theta_{IPC})$, based on the divergence between D and Θ_{IPC}. We measure this divergence by calculating the log-likelihood ratio between D and Θ_{IPC}, normalized by the collection C as defined below:

[1] http://www.wipo.int/classifications/ipc/en/

$$P(D|\Theta_{IPC}) \propto H(\Theta_D, \Theta_C) - H(\Theta_D, \Theta_{IPC})$$
$$= Z_D \sum_{t \in V} P(t|\Theta_D) \log \frac{P(t|\Theta_{IPC})}{P(t|\Theta_C)}$$

where $H(\Theta_D, \Theta_C)$ represents the cross entropy between the sample document D and the collection and $H(\Theta_D, \Theta_{IPC})$ represents the cross entropy between the sample document D and the topical model of relevance Θ_{IPC}. We define $Z_D = 1/\sum_{D \in IPC} P(D|\Theta_{IPC})$ as a document-specific normalization factor. This approach assigns higher scores to documents which contain specific terminology and are more similar to Θ_{IPC} and less similar to the language model of the collection Θ_C. Θ_{IPC} is similar to CBQM query model calculated in Chapter 4. For estimating the term importance $P(t|D)$ in Equation 5.4, we consider the smoothed maximum likelihood estimate of a term to avoid sparseness issues as shown in Equation 5.1.

We then mix the estimated relevance model using the conceptual tags and the original query in order to build an expanded query. To do this, we use a linear combination as expressed in the following:

$$P(t|Q_{expand}) = (1 - \mu) \cdot P(t|\theta_{IPC}) + \mu \cdot P(t|Q_{orig}) \tag{5.5}$$

where $P(t|Q_{orig})$ and $P(t|\theta_{IPC})$ show the probability of term t given the original query model and the estimated relevance model, respectively. We refer to this expanded query model as EX-RM. The expanded query model EX-RM is similar to a linear interpolation between LLQM and CBQM query models which were introduced in Chapter 4.

The performance of different uni-gram query models presented in this section is compared to each other in the experimental section. For comparison purposes, we also show the performance of Pseudo Relevance Feedback (PRF), as a reference baseline, and we compare this to the query models estimated in this section. In the experiment section we show that the relevance model constructed based on the conceptual tags (EX-RM) outperforms the result of PRF. To generate a query we pick the top-k terms with higher weights from each query model.

5.4 Extracting Candidate Key Phrases

We recognized and extracted candidate noun phrases with length at most 5 from the query patent, by using the Stanford *part of speech tagger* [100]. The part-of-

speech tagger assigns part-of-speech tags (e.g., noun (NN), verb (VB), adjective (JJ), etc.) to each term w in document d. The part-of-speech tagger applies a pre-trained classifier on w and it's surrounding terms in d. We consider all noun phrases as candidate phrases, and compute score s_p by extracting all such phrases from d. We are interested to find ordinary phrases rather than extracting named entities. Some example noun phrase patterns[2] that we used are listed in Table 5.1.

Table 5.1. Examples of extracted noun phrases and corresponding patterns

Pattern	**Instance**
NN	leukocyte
JJ NN	miniature column
NN NN	blood filtration
JJ JJ NN	hydrophobic polymerizable monomer
NN NN NN	leukocyte removal performance
JJ NN NN	nonwoven polyester fabric
JJ JJ JJ NN	protonic neutral hydrophilic part
NN NN NN NN	blood transfusion side effect
...	...
NN NN NN NN NN	coating leukocyte removal filter material

5.5 Scoring Key Phrases

We used the two methods proposed in [110] for scoring phrases. In the following, we briefly revisit the two scoring approaches. The first approach employs TF/IDF information for evaluating the importance of each phrase, while the second calculates a weight for each phrase using mutual information.

5.5.1 Scoring Phrases based on TF/IDF

The first scoring technique assigns a score $s_t(p)$ to a phrase p which is based on a linear combination of the total TF/IDF score of all terms in p and the degree of *coherence* of p. Coherence quantifies the likelihood of the constituting terms in forming a single concept and is a measure of the stability of a phrase

[2]Presented instances belong to query PAC-433 in the topic set.

in the corpus. Formally, let $|p|$ denote the number of terms in phrase p; we use $w_1, w_2, ..., w_{|p|}$ to refer to the actual terms. $s_t(p)$ is formally defined as:

$$s_t(p) = \sum_{i=1}^{|p|} tf.idf(w_i) + \alpha \cdot coherence(p) \tag{5.6}$$

where $idf(w_i)$ is the inverse document frequency of w_i and α is a tunable parameter. The first component in $s_t(p)$ captures the importance of each term in p by using the TF/IDF value. A rare term that occurs frequently in d is more important than a common term frequently appearing in d (i.e. with low idf). This component will reward rare phrases. The second component in $s_t(p)$ represents how coherent the phrase p is. The coherence of p is defined as:

$$coherence(p) = \frac{tf(p) \cdot (1 + \log\ tf(p))}{\frac{1}{|p|} \cdot \sum_{i=1}^{|p|} tf(w_i)} \tag{5.7}$$

where $tf(p)$ is the number of times the phrase p appears in the document d. Equation 5.7 compares the frequency of p with the average tf of its terms. The additional logarithmic component gives importance to phrases appearing frequently in the input document.

We expand the uni-gram query model with the top-k concepts selected by the TF/IDF scoring method and refer to it as QM-NP1 in the rest of this chapter.

5.5.2 Scoring Phrases based on Mutual Information

The second scoring technique assigns $s_m(p)$ to a phrase p. The score is based on the mutual information (MI) between the terms of phrase p and the idf values from the background corpus. $s_m(p)$ is a linear combination of idf values of terms in p, frequency of p, and the point-wise mutual information among them. $s_m(p)$ is formally defined as:

$$s_m(p) = \sum_{i=1}^{|p|} idf(w_i) + \log \frac{tf(p)}{tf(POS_p)} + PMI(p) \tag{5.8}$$

where $tf(p)$ and $tf(POS_p)$ are the number of times p and its part-of-speech tag sequence POS_p appear in d and in its part-of-speech tag sequence POS_d, respectively. The first part represents how descriptive each term in phrase p is. The second part identifies how frequent the phrase p is at the corresponding POS tag sequence in the document. The third part captures how likely are the terms

to from a phrase together. Mutual information compares the probability of observing the constituting terms in phrase p together (the joint probability) with the probabilities of observing them independently. The $PMI(p)$ for a phrase p is defined as:

$$PMI(p) = \log(\frac{P(p)}{\prod_{i=1}^{|p|} P(w_i)}) \tag{5.9}$$

where $P(w_i)$ and $P(p)$ denote the probability of occurrence of w_i and phrase p respectively at the appropriate part-of-speech tag sequence. They are formally defined as:

$$P(p) = \frac{tf(p)}{tf(POS_p)},\ P(w_i) = \frac{tf(w_i)}{tf(POS_{w_i})} \tag{5.10}$$

In order to emphasize the importance of occurrence frequency of phrase p in document d we weight Equation 5.8 by $\frac{tf(p)}{tf(POS_p)}$ as shown below:

$$s'_m(p) = \frac{tf(p)}{tf(POS_p)} \cdot \left(\sum_{i=1}^{|p|} idf(w_i) + \log \frac{tf(p)}{tf(POS_p)} + PMI(p) \right)$$

In the rest of this chapter we combine the uni-gram query model from mutual information-based scoring method and refer to it as QM-NP2. We analyze the role of the number of selected phrases on the final performance of the system in the Section 5.7.

5.6 Predicting Noun Phrase effectiveness

Our goal is to predict whether an expanded query using noun phrases will be more effective for retrieval than an unexpanded query. We evaluate the effectiveness of the expanded query by estimating the change in the average precision (AP) for each query.

Let $AP(Q_{orig})$ and $AP(Q_{expand})$ be the AP of the original uni-gram query and of the expanded query using noun phrases, respectively. We measure the performance change due to the Q_{expand} as:

$$chg(AP, Q) = \frac{AP(Q_{expand}) - AP(Q_{orig})}{AP(Q_{orig})} \tag{5.11}$$

We set a threshold at 10% for this change in AP to distinguish a good expanded query from a bad one and this indicates an effective noun phrase expansion.

After identifying a good expanded query according to Equation 5.11, we use this estimate to decide whether the original query should be expanded or not. We then perform a selective query expansion (SQE) where we only expand effective noun phrase queries.

Before trying to estimate the effectiveness of the noun phrase query, it is interesting to know how well SQE will perform if the true effectiveness value is used. To this end, we use the average precision of Q_{orig} and Q_{expand} to decide whether to expand a query or not. We refer to this approach as $oracle_{np}$ showing the potential upper bound of what can be achieved by combining the two rank lists based on true effectiveness. Table 5.2 shows the MAP for the top 1000 results with $oracle_{np}$ in comparison to the original uni-gram query model (baseline) and the expanded query model using noun phrases (QM-NP2). The † and ‡ symbols indicate that the improvement over the uni-gram baseline and QM-NP2 is statistically significant at $p<0.01$.

Table 5.2. Performance results using the true noun phrase effectiveness

model	MAP
baseline	0.1366
QM-NP2	0.1380
$oracle_{np}$	0.1649 † ‡

As the result of Table 5.2 suggests, we can achieve a 20% improvement over both uni-gram baseline and QM-NP2, by employing the true effectiveness of the noun phrases. We seek to reach this upper bound by a reasonable estimation of the correct AP values and the change of AP for each query according to Equation 5.11.

5.6.1 Features

In order to predict the effectiveness of an expanded query we use a set of features related to the query to estimate AP of both rank lists of Q_{orig} and Q_{expand}. We describe these features in this Section.

The Query Clarity (QC) measure [26] quantifies the level of effectiveness of a query at retrieving a specific topic. The clarity measure is the Kullback-Leibler (KL) divergence between the query language model $P(w|Q)$ and the collection

language model $P(w|C)$. Formally, the clarity score is defined as:

$$D_{KL}(Q||C) = \sum_{w \in V} P(w|Q) \log \frac{P(w|Q)}{P(w|C)} \tag{5.12}$$

A higher clarity score indicates a clearer query with specialized vocabulary and a lower clarity score indicates a more ambiguous query with a very generic language. To calculate a clarity score in a given collection, a *relevance model* is constructed. This model captures the language usage of documents related to the query and therefore it is a collection-dependent query model.

We propose two measures inspired by Query Clarity using patent-specific characteristics. Let IPC_Q be the set of documents with similar topics to Q represented by conceptual tags. The first measure, called *Topical Clarity*, is defined as KL-divergence between the language model of Q and the language model of IPC_Q. Formally, the Topical Clarity (TC) measure is defined as:

$$D_{KL}(Q||IPC_Q) = \sum_{w \in V} P(w|Q) \log \frac{P(w|Q)}{P(w|IPC_Q)} \tag{5.13}$$

where $P(w|IPC_Q)$ is the relative frequency of term w in documents with similar conceptual tags to Q. Larger KL-divergence indicates a query with fewer topics and therefore more focused, while a smaller KL-divergence indicates a query with a broader language use.

The second measure called *IPC-based Clarity* captures the similarity between the language usage of IPC_Q and the collection language model. This measure is defined as:

$$D_{KL}(IPC_Q||C) = \sum_{w \in V} P(w|IPC_Q) \log \frac{P(w|IPC_Q)}{P(w|C)} \tag{5.14}$$

An alternative indication of the specificity of a query is to consider the distribution of the informative amount in the query terms [48]. This measure is defined by:

$$\gamma_1 = \sigma_{idf} \tag{5.15}$$

where σ represents the standard deviation of the *idf* of the terms in Q. Each query term can be associated with an inverse document frequency ($idf(w)$) describing the informative amount that a query term q carries. The $idf(w)$ is defined by:

$$idf(w) = \log \frac{N - N_w + 0.5}{N_w + 0.5} \tag{5.16}$$

where N_w is the number of documents in which the query term w appears and N is the number of documents in the whole collection.

Table 5.3. Features used in the regression model for query Q

Features	
QC	Query Clarity
TC	Topical Clarity
tag-clarity	IPC-based Clarity
γ_1	Informative Amount in the Query
QS	Query Scope

Another measure that can be used to predict query performance is the Query Scope (QS) [48]. This measure uses the size of the document set containing at least one of the query terms to infer the query performance. Formally, the query scope is defined as:

$$QS = -\log(n_Q/N) \tag{5.17}$$

where n_Q is the number of documents containing at least one of the query terms, and N is the number of documents in the whole collection.

These features are summarized in Table 5.3. Note that the length of generated queries is similar, thus we did not consider this property as a feature.

To learn a performance prediction model using these features we define the following regression problem.

$$\underset{\Phi}{argmin} \sum_{Q \in T} ||\Phi(F(Q)) - AP(Q)||^2 \tag{5.18}$$

where T is a set of training topics and F is a mapping from query to feature space. F also defines a mapping from the respective rank list of query to feature space.

5.6.2 Evaluating the Dependence between the Predictors and Average Precision

In this section, we will examine the correlations of the predictors with the query performance. We use AP as the focus measure indicating the query performance in our experiments. To investigate the effectiveness of the predictors, we check the Spearman rank correlation and linear regression because of their power in showing correlation between predictors and AP as suggested by previous studies [32, 48].

Table 5.4. Linear Regression and Spearman rank correlation coefficient of the query performance predictors with Average Precision

	LR		Spearman	
Features	r	p-value	rs	p-value
QC	0.2180 †	0.05	0.3645 †	0.01
TC	0.2466 †	0.05	0.3170 †	0.01
tag-clarity	0.0943	0.28	0.1812 †	0.05
γ_1	0.0491	0.61	0.1100 †	0.05
QS	0.1956 †	0.05	0.2278 †	0.01

The linear regression assumes a linear distribution of the involved variables, which is not necessarily valid in our case. As the distribution of the involved variables is unknown, a non-parametric measure such as the Spearman rank correlation (which does not assume any particular structure for the relationship) can find stronger relationships. However, the Spearman rank correlation can not find relationships between the combinations of predictors and AP.

Table 5.4 summarizes the results of the linear correlation of each predictor (individually) with AP on the training data. We know that the relationship between predictors and AP may be nonlinear, but this allows us to compare the importance of the features by examining their coefficients. We also examine the importance of the features by examining the significance of their correlation with AP. A † symbol denotes a statistically significant correlation with AP at the reported level of p-value using paired t-test.

In order to model the complex nonlinear relationships between combinations of predictor variables, we use Stochastic Gradient Boosting Tree (SGBT) [34]. This model produces an ensemble of weak prediction learners, i.e., decision trees. It builds additive regression models in a stage-wise manner and it generalizes them by allowing optimization of an arbitrary differentiable loss function. For the SGBT, we used the gbm2 package implemented in R[3]. SGBT can find a sub-combination of features that may aid with the prediction of AP. With this model we can get a prediction of AP for any input. Notice that Φ in Equation 5.18 represents an additive model of multiple decision trees which is learned by SGBT.

[3]available at `http://cran.r-project.org/web/packages/gbm/`

5.7 Experiments

In this section, we present the experimental evaluation results of our proposed method for refining patent queries using concept importance predictors.

First, we describe our experimental setup and the three experimental settings used in our study. In the first setting, we compare different uni-gram query models estimated from the query patent and show their retrieval effectiveness on CLEF-IP 2010 dataset. In the second setting, in order to find out whether we can find a clearer focus of the query patent, we expand the uni-gram query with extracted important key concepts (e.g., bi-grams or phrases). We determine the optimal parameter settings for each query model using training data and we compare the effectiveness of expansion using noun phrases with the baseline uni-gram queries. In the third setting, in order to find out whether query performance predictors can indicate a successful application of phrases, we conduct an experiment where we estimate the effectiveness of using noun phrases based on the feature set proposed in Section 5.6. We then combine the result of the uni-gram query and the expanded query using the outcome of the prediction model. We show that the best performance is achieved by expansion using noun phrases in a query dependent manner.

Please note that we used the text of the *description* section for building the query model. We used BM25 for retrieving and scoring documents.

5.7.1 Uni-gram Query Models

In this section the performance comparison of different uni-gram query models is presented. The result of our baseline method, Q_{orig}, is comparable to the second best result of the CLEF-IP 2010 [72]. In our experiments, we set the smoothing parameter λ to 0.5 while calculating the original query model. Table 5.5 reports a comparison of two query expansion models EX-RM and PRF against the baseline.

The expanded query model PRF is formed based on Pseudo Relevance Feedback. We combined the original query with the expanded query, where the parameter μ controls the weight of the uni-gram query. We used the training data for tuning this parameter and the optimal value for μ is set to 0.6. The result of Table 5.5 are obtained using 10 expansion terms extracted from the top 10 documents and the number of terms used for building the original query is set to 30. Results marked with † achieved statistically significant improvement over the baseline at p-value of 0.01 using randomization test.

According to the results presented in Table 5.5, the PRF method is not able

Table 5.5. Performance comparison of the uni-gram query models, the baseline run, relevance models using pseudo feedback documents and sample relevant documents

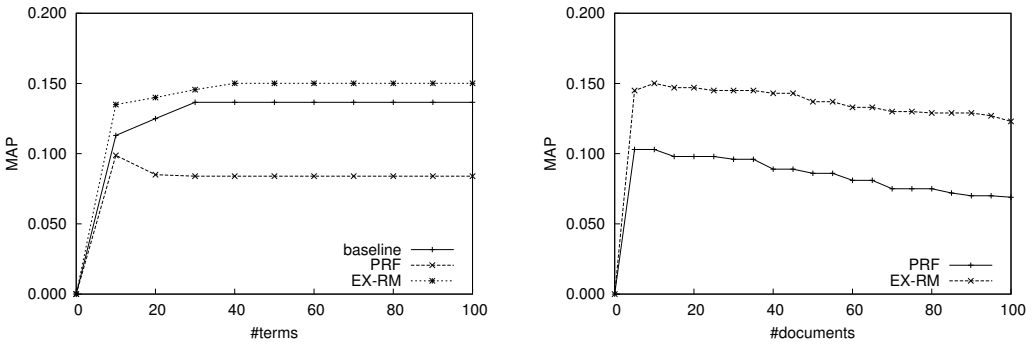
model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
PRF	0.103	0.590	0.481
EX-RM	0.150 †	0.643	0.553

to select the best terms for query generation and all three reported performance measurements decrease compared to the baseline. This is due to the poor quality of search results. However, the relevance model using the sample documents, EX-RM, significantly outperforms the baseline run. This suggests that using the sample documents was beneficial for building the expanded query model and EX-RM on average achieved 13% improvement over the baseline in terms of MAP.

We explore the sensitivity of each of the uni-gram query models, baseline run, EX-RM and PRF, to the number of query terms that needed to be taken into account. We also look into the number of feedback documents that needed to be taken into account for both expanded uni-gram query models EX-RM and PRF. Figure 5.1 presents the MAP of our techniques, for varying values of number of feedback terms, and number of feedback documents. We can see that the number of terms is not highly influential and any value higher than 30 produces the same results. However, the system is more sensitive to the number of feedback documents. In fact, it can be seen that values higher than 10 hurts the performance.

5.7.2 Combining Uni-gram and Phrase Query

We wish to examine the quality of the phrases obtained by the two different techniques explained in Section 5.5 in the task of prior art search. Our goal is to utilize such phrases to identify documents relevant to the query patent. We first combine the uni-gram query model from the query patent with the top-k concepts selected by two scoring methods: a) the TF/IDF scoring method, denoted by QM-NP1; b) the mutual information-based scoring method, denoted by QM-NP2. We further examine expanded queries in which we select the top-k concepts from the pseudo feedback documents using two scoring methods:



a) Number of feedback terms b) Number of feedback documents

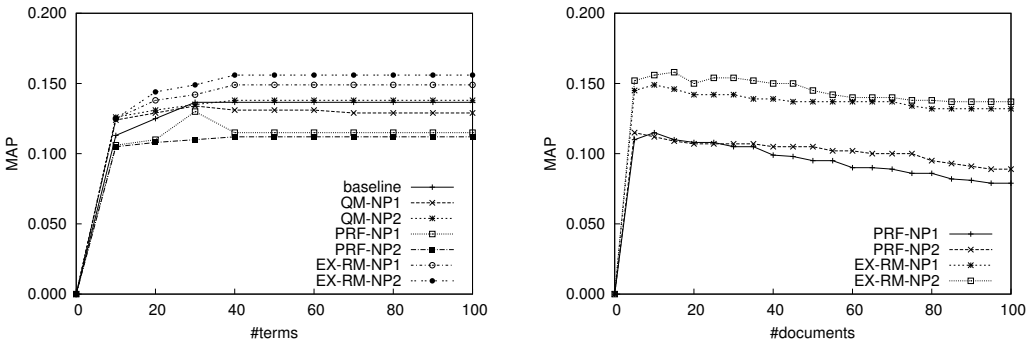
Figure 5.1. Sensitivity of uni-gram query models against (a) the number of terms and (b) the number of feedback documents used for query model construction

PRF-NP1 and PRF-NP2. Finally, in order to use the evidence from the relevance set (i.e. documents with similar conceptual tags), we selected the top-k scoring noun phrases from the relevance set using the two scoring methods. We refer to these methods as EX-RM-NP1 and EX-RM-NP2.

The retrieval results of various combinations of uni-gram queries with phrases are reported in Table 5.6. Results marked with † are significantly better than the baseline and ‡ represents the significant improvement achieved by EX-RM-NP2 against EX-RM-NP1.

Table 5.6. Performance of the expanded query models using phrases

model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
QM-NP1	0.131	0.600	0.521
QM-NP2	0.138	0.621	0.539
PRF-NP1	0.115	0.592	0.494
PRF-NP2	0.112	0.603	0.493
EX-RM-NP1	0.149 †	0.646 †	0.552
EX-RM-NP2	0.156 †‡	0.650 †	0.567



a) Number of feedback terms b) Number of feedback documents

Figure 5.2. Sensitivity of the expanded query models using noun phrases against (a) the number of terms and (b) the number of feedback documents used for expanded query model construction

Our experiments indicate that expansion based on phrases extracted by the mutual information-based scoring technique outperforms TF/IDF based scoring most of the time. This suggests that using co-occurrence information is more helpful in identifying key concepts of a query patent compared to using frequency information alone.

As follows from Table 5.6, extracting concepts from the query patent, as done for QM-NP1 and QM-NP2, does not improve the results over the uni-gram baseline. As we expected, the PRF based expansion decreases the result in terms of MAP, recall, and PRES. It is clear that both relevance models built using similar conceptual tags, EX-RM-NP1 and EX-RM-NP2, outperform our uni-gram baseline significantly. This result demonstrates a positive effect of expansion using both scoring methods. In both cases these improvements hold for MAP, recall and PRES.

A very interesting conclusion which can be made by comparing the results of Table 5.5 and Table 5.6 is that despite the significant improvement of EX-RM-NP1 and EX-RM-NP2 over the baseline, the improvement over EX-RM is not significant. We performed an analysis on the query set and we found that almost 600 queries out of 1348 queries were hurt by the expansion using phrases compared to using uni-grams. We therefore decided to estimate an upper bound of performance by combining these two approaches in a query dependent manner. As we already saw in Section 5.6, we found that by using the true effectiveness of the noun phrase queries we can achieve an increase in performance of 20% in

terms of MAP. In the next section, we show how we can estimate the importance of a noun phrase query in order to decide whether to expand a given query using noun phrases or not.

In our experiments we considered proximity matches rather than exact phrases. This is due to the fact that using proximity matches gave us a consistent gain in retrieval effectiveness in comparison to using exact phrases. We use a window of size 8, as suggested by previous work on proximity matching [81]. We perform a sweep (grid search) on μ to determine the optimal mixture of the original query and the expanded query according to Equation 7. The optimal value is set to 0.6 for all expansion methods.

We selected the top 10 phrases and added them to our uni-gram queries. We studied the sensitivity of each approach against the number of feedback documents and the number of feedback phrases that need to be taken into account. The results are shown in Figure 5.2. An important observation to be made from Figure 5.1 and Figure 5.2 is that using 10 pseudo relevant documents and around 40 feedback terms resulted in the best performance for all expansion methods.

5.7.3 Selective Query Expansion Using Key Concepts

So far we estimated uni-gram and expanded query models using three different sources: 1) the query patent; 2) the pseudo relevant documents retrieved from PRF; 3) the relevance set which is composed of documents with similar conceptual tags to the query.

In this section, our goal is to predict whether query expansion using phrases is effective. We first predict the AP of each query in both rank lists of the expanded and unexpanded query using the features described in Section 5.6. We then calculate the change in AP after expansion based on the predicted values. A positive change in AP after expansion indicates an effective expansion. In the experiments, we considered a change bigger than 10% to be an effective expansion. We use this prediction value to decide which of the two competing methods will offer the best result for a given query.

We used a five-fold cross validation for our experiments. We divided the query topics into five equal parts. We trained the estimator using four out of five parts and applied the training model to estimate the AP of the remaining queries. We repeated the same test process on each of the five parts and report the results on average over all five parts. The same procedure was performed for the expanded and unexpanded lists.

Table 5.7 shows the result of our method for selective query expansion (SQE).

Table 5.7. Retrieval results on CLEF-IP 2010 using selective query expansion

model	MAP	Recall	PRES
baseline	0.136	0.619	0.535
QM-NP2	0.138	0.621	0.539
SQE_Q	0.152 $*$ $\star$	0.617	0.543
PRF	0.103	0.590	0.481
PRF-NP2	0.112	0.603	0.493
SQE_{PRF}	0.122 $\uparrow$ $\Uparrow$	0.609	0.509
EX-RM	0.150	0.643	0.553
EX-RM-NP2	0.156	0.650	0.567
SQE_{EX-RM}	0.168 † ‡	0.668	0.580

The $*$ and $\star$ symbols indicate that the improvement achieved by SQE_Q over the expanded rank list (QM-NP2) and unexpanded rank list (baseline) is statistically significant at $p<0.01$. The $\uparrow$ and $\Uparrow$ symbols indicate that the improvement achieved by SQE_{PRF} over the expanded rank list (PRF-NP2) and unexpanded rank list (PRF) is statistically significant at $p<0.01$. The † and ‡ symbols indicate that the improvement achieved by SQE_{EX-RM} over the expanded rank list (EX-RM-NP2) and unexpanded rank list (EX-RM) is statistically significant at $p<0.01$.

As follows from Table 5.7, for all the three settings of our experiments, selective query expansion achieved statistically significant improvement in terms of MAP over automatic query expansion (using expansion on all queries). This indicates that the chosen features were able to accurately predict the AP for the expanded and unexpanded lists of most of the queries. This also suggests that the predicted change in AP was overall a good indicator of an effective expansion. A per-query analysis showed that the result of SQE method was able to detect more than half of the queries which performed well using the expansion and therefore SQE was able to effectively improve the retrieval effectiveness of those queries. However, the SQE method did not achieve the upper bound performance shown in Table 5.2, which is due to the error made by the prediction model. Despite the achieved increase in terms of MAP, there is still room for improvement which requires the choice of better features.

We calculated the influential features from the learnt SGBT model [34]. Query clarity, Topical clarity and IPC-based clarity are the most influential features.

5.8 Conclusions

In this chapter, we presented several versions of uni-gram and noun phrase queries for prior art search. By evaluating these query models we found that more advanced IR techniques increase the performance of specific queries but the aggregated result may degrade against the baseline. To achieve consistent improvement in all queries we used a selective query expansion framework. The main contribution of this chapter is devising a method for predicting whether expansion using noun phrases improves the retrieval effectiveness of a query.

We experimentally determined the upper bound of what can be achieved by looking into the true effectiveness using a noun phrase query. We used a few common used features for predicting AP and we proposed some features using patent-specific characteristics. Our selective query expansion method using noun phrases obtained a statistically significant improvement over the expanded and unexpanded rank lists. Better features still need to be extracted which can better capture the quality of the results.

We experimented with two different scoring methods for selecting noun phrases. The scoring based on mutual information achieved better results over TF/IDF scoring. Another interesting conclusion can be made by comparing the two relevance models which were used in this study. The first relevance model was built based on PRF and the second was built by employing documents with similar conceptual tags to the query. The retrieval effectiveness of the rank list after performing PRF was lower than the retrieval effectiveness of the initial rank list. The reason can be attributed to the low MAP of the initial rank list which led to the selection of poor quality pseudo feedback documents. However, the relevance model built based on the conceptual tags obtained a better retrieval effectiveness compared to the initial rank list.

In the next chapter, we will take a closer look at the metadata associated with the patent application and study how to employ this information to improve the query model estimated from the patent application.

6

IPC-based Conceptual Lexicon for Query Disambiguation

"On the road from the City of Skepticism, I had to pass through the Valley of Ambiguity."
– Adam Smith

6.1 Introduction

In this chapter, our aim is to investigate the term mismatch problem in patent retrieval by leveraging a domain-dependent resource via query expansion. To do this, we first construct a lexicon from IPC definition pages[1]. Definition of IPC classes consists of explanations of each IPC class which can be used to identify the important topics (concepts) and subtopics of the query. We extract expansion concepts specific to each query from this lexicon for query expansion. We then use term proximity information to calculate reliable importance weights for the expansion concepts. To this end, we propose a proximity-based query propagation method to calculate the query term density at each point in the document. Our proximity-based framework incorporates positional information into the estimation of the importance of expansion concepts so that we can reward expansion concepts occurring close to query terms. This way we can concentrate our attention on the terms that are associated with the query terms and avoid the topic drift which is caused by taking into account irrelevant terms.

We start this chapter by reviewing different relevance evidences in Section 6.2. A schematic architecture of our proposed solution is presented in Section 6.3. We

[1]Available at http://web2.wipo.int/ipcpub/

then discuss the procedure to build a query-specific lexicon in Section 6.4. We perform query expansion by deriving expansion concepts from the query-specific lexicon and we use positional information to calculate weights for insuring high quality expansions in Section 6.5. To this end, we utilize kernel functions to keep track of the distance of expansion concepts from query terms. Consequently, words appearing within the neighborhood of a given query term are more likely to be associated with that query term. Finally, in Section 6.6, we evaluate the performance of the proposed query expansion methods and we analyze the quality of the extracted expansion terms from the IPC conceptual lexicon in terms of their impact on the final performance of the expanded rank list.

6.2 Potential Relevance Evidences for Query Reformulation

In this section we categorize different information sources that can be used as additional information for query reformulation in patent retrieval.

IPC Classification. The International Patent Classification (IPC classification) provides a hierarchical categorization over different technological fields such as computer science, electronics, mechanics, and bio-chemistry. Such classes provide language independent symbols assigned as metadata to the patent documents. They categorize the content of a patent document and describe the field of technology a patent document belongs to. These IPC classes can be seen as conceptual tags assigned to the documents [65]. For each conceptual tag there are textual descriptions available (IPC definition pages) that provide contextual cues about different technical fields.

Citation Chain. Granted patents are published with a list of other patent and non-patent documents that were cited during the processing of the patent application either by the patent examiner or the inventor. A patent searcher has access to these cited documents but also to documents that cite each patent. The process of searching these two sets of documents is referred to as *backward* and *forward*[2] citation searching, respectively.

[2]A "backward citation" is the term used for a traditional citation: it is the document that was published earlier, and which appears on the newer document's front page. In turn, the newer document is called the "forward citation" or "citing document". Obviously forward citations cannot appear on a document's front page, but they can easily appear on the patent record in an

The above sources have different vocabulary usage compared to the initial query patent. The query patent itself has an obscure style of writing, (so called "patentese") [65]. This characteristic might create a term mismatch problem in finding relevant documents for a given patent. However, the other two resources provide a more established vocabulary usage. The descriptions of IPC classes represent the standard vocabulary usage related to different domains. The citation chain contains the language used by the community of inventors related to the subject of the invention of the query. Thus, the vocabulary usage of the two latter sources is complementary to the query itself. In this chapter, we will use IPC classes to build an IPC lexicon which will later be used for query expansion. The citation information will be used in the next chapters to reduce the term mismatch between the initial query patent and the relevant documents.

6.3 The Architecture of our Proposed Model

Figure 6.1 illustrates the general scheme of our proposed method for proximity-based query expansion using the IPC lexicon. The system receives a full patent application (query patent) consisting of textual fields and classification information.

In the first step, we estimate a query model from the textual fields of the patent and in parallel we build a query-specific lexicon from IPC definition pages. In step II we perform a lookup in the lexicon using the IPC classes of the query document. In step III we extract the terms related to the IPC classes of the query from the IPC lexicon. In step IV, we expand the initial query model with expansion concepts extracted from the lexicon. In this step a query expansion is performed and positional information between query terms and expansion terms is used to calculate weights for ensuring high quality expansion. Finally, in step IV, the final rank list is generated, as a result of the previous step.

6.4 IPC Conceptual Lexicon

We now explain the process of building a lexicon from IPC definition pages. We refer to this lexicon as a *conceptual lexicon*. We consider the description of an IPC subgroup[3] as a text segment. We performed stop-word removal on these text segments. We then filtered out the patent specific stop-words. The list

electronic database.

[3]IPC classification scheme is arranged in a hierarchical, tree-like structure. Subgroup is the lowest hierarchical level in the IPC hierarchy.

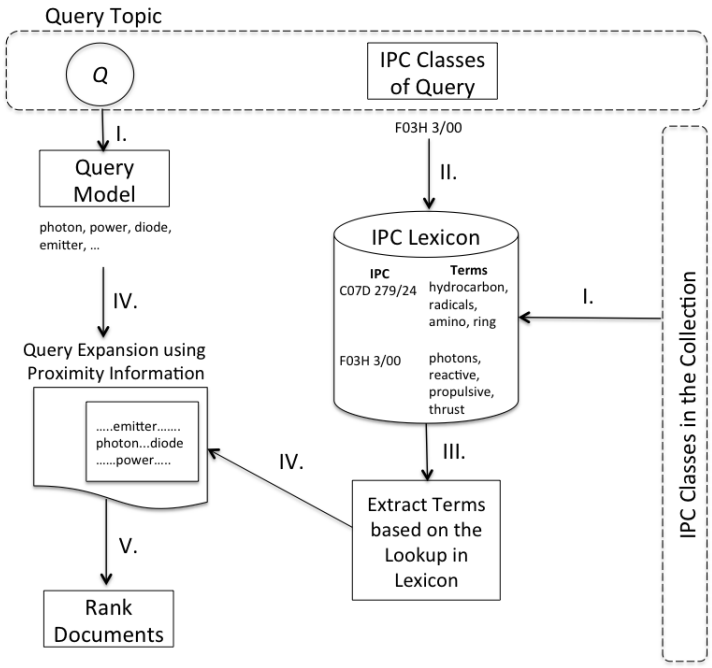


Figure 6.1. The general scheme of our proposed method for query expansion using IPC lexicon. Numbers indicate the sequence flow of operations.

for patent specific stop-words is built as follows. We first calculated document frequencies for each term in the collection. We selected terms with top 10% highest document frequency and considered them as patent specific stop-words. The threshold of 10% was set experimentally. We then filter these terms out to increase the accuracy of our lexicon. Examples of these patent specific stop-words are “method”, “device”, “apparatus”, and “process”.

Each entry in our lexicon is composed of a key and a value. The key is an IPC class and the value is a set of terms representing the mentioned class. An example of an entry in the conceptual lexicon is presented in Table 6.1.

Table 6.1. An entry in the conceptual lexicon.

IPC Class	Representing Terms
C07D 279/24	hydrocarbon, radicals, amino, ring, nitrogen, atom

The lexicon can be used to extract expansion concepts related to the information need of a given query patent. To this end, the IPC classes of the query

patent are searched in the lexicon and the matching terms (corresponding to the searched IPC class) are considered as expansion terms.

Query expansion using the lexicon will help us solve the two following problems. The first is related to the fact that the usage of words is sensitive to the topic domain; in different domains the same word may have different meanings. We aim at finding the correct sense of a word, by associating relevant terms from the topic domain to the given query terms for each query patent.

The second problem is related to the term mismatch. The vocabulary of the query patent is tailored by the language usage of the author (who often uses a non-standard terminology), while conceptual lexicon provides a standard terminology. We try to combine these two terminologies, as we think this might alleviate the term mismatch.

6.5 A Proximity-based Framework for Query Expansion

We now explain how the IPC lexicon is used for query expansion. To do this, we first describe strategies to identify expansion terms that refer to query terms in Section 6.5.1. Then in Section 6.5.2 we explain how to estimate the probability that an expansion term refers to a query term. Finally in Section 6.5.3 and 6.5.4 we discuss calculating relevance scores for documents.

6.5.1 Query Reformulation

In general terms, let $Q = \{q_1, q_2, \ldots, q_k\}$ be a query composed of top-k query terms with highest weights according to a query model estimated from the query patent document D_Q (as explained in Equation 6.3). Given the IPC classes assigned to D_Q, we select a set of concepts $C_E = \{e_1, e_2, \ldots, e_m\}$ from the conceptual lexicon (as explained in Section 6.4). The set C_E is associated to the query Q since the IPC lexicon contains explanations about the IPC classes of D_Q. Once the set of concepts C_E is identified, we determine the importance weights according to their distance from the query terms based on the intuition that concepts closer to query terms are more related to the query. Equation 6.1 shows the process of calculating importance weights for expansion concepts. We can then re-rank documents in the initial rank list $\mathbb{R}$ using a weighted combination of matches of concepts in C_E and our initial keyword query Q based on Equation 6.3.

We explain four specific strategies for selecting expansion concepts in the following.

Explicit Expansion Concepts. In this strategy we use the concepts in our conceptual lexicon that match against the IPC classes of D_Q. However, we restrict our attention to concepts that are present in D_Q. This provides a set of explicit expansion concepts (a subset of C_E) which serves as candidate expansion terms. We refer to this set as X_E. We use the proximity of query terms and expansion terms inside D_Q to assign importance weights to items in X_E. These weights are then used to re-rank documents in the list $\mathbb{R}$.

Implicit Expansion Concepts. In this strategy the expansion terms are not limited to the set of explicit expansion concepts X_E which were defined previously. Instead, our query expansion method includes all expansion concepts in C_E. In this setting we extract proximity information from documents inside $\mathbb{R}$ to compute importance weights for expansion terms. This strategy is able to use of all terms available in C_E and is not limited to the concepts that appear in D_Q.

Combining Search Strategies. In this strategy, instead of expanding the initial query, we calculate an IPC score based on the expansion concepts in C_E. We linearly combine this score with the initial scores calculated in $\mathbb{R}$. Our goal is to compare whether having a unified query, as it exists in the query expansion, is better than constructing two separate queries and combining their results at the end. We introduce this setting for the experiments in order to simulate the specific search strategies taken by searchers for retrieving relevant documents. In such a search strategy searchers perform separate searches based on different information sources, such as the query patent document and IPC classes, and then merge the results of the runs together to produce a unique rank list [65].

Proximity-based Pseudo Relevance Feedback. As a comparison baseline we use the retrieval corpus as a source for PRF and we use the feedback set for selecting expansion terms. The distance between query terms and expansion terms is used to calculate the weight for expansion terms.

As an example, Table 6.2 shows the terms selected from different information resources for the query patent "EP-1783182-A1" selected from CLEF-IP 2010 test topics. The terms from the retrieval corpus are selected via the PRF procedure.

6.5.2 Estimating Query Relatedness

In this section, we explain our method for estimating the probability that expansion term e at position i is related to query term q. We calculate this probability

Table 6.2. Comparison between the list of expansion terms derived from three information sources for the query with title "ink-jet recording ink".

Query Document	Conceptual Lexicon	Retrieval Corpus
acrylate, ink, jet, acid, polymer, pigment, record, ...	light-sensitive, duplicate, printer, ink, sheet, mark, ...	record, liquid, surface, composition, polymer, cartridge, ...

as:

$$P(q|i,D) = \sum_{j} P(q|j)P(j|i,D) \tag{6.1}$$

where D denotes a document, i denotes an expansion term position, and $j = \{1,2,\ldots,k\}$ denotes a set of query term positions. $P(q|i,D)$ indicates the probability that the expansion term at position i in D is about query term q. We refer to this probability as *query relatedness probability*. To find the query relatedness at position i, we calculate the propagated probability from all query positions at position i. For every position j in D, we consider the weight of query term at position j, denoted by $P(q|j)$, and weight it by the probability that the term at position j is about the expansion term at position i, denoted by $P(j|i,D)$. This probability is estimated as follows:

$$P(j|i,D) = \frac{k(j,i)}{\sum_{j'=1}^{|D|} k(j',i)} \tag{6.2}$$

where $k(i,j)$ is the kernel function determining the weight of the propagated query relatedness from j to i. We model query relatedness by placing a density kernel function around query terms.

We study three different density kernel functions, namely Gaussian, Laplace, and Rectangle. We selected Gaussian and Laplace kernels as they have been shown to be the best performing kernels among the kernel functions tested in previous work [39, 66]. We also chose Rectangle kernel to simulate the effect of imposing a hard boundary over passages in contrast to the soft boundary introduced by other kernels. The parameter σ controls the spread of kernel curves and restricts the propagation scope of each term.

- **Gaussian Kernel**

$$k(i,j) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\frac{-(i-j)^2)}{2\sigma^2}\right]$$

- **Laplace Kernel**

$$k(i,j) = \frac{1}{2b} \exp\left[\frac{-|i-j|}{b}\right]$$
$$\text{where } b = \frac{\sqrt{2}}{2}\sigma$$

- **Rectangle Kernel**

$$k(i,j) = \begin{cases} \frac{1}{2a} & \text{if } |i-j| \leqslant a \\ 0 & \text{otherwise} \end{cases}$$
$$\text{where } a = \sqrt{3}\,\sigma$$

Our aim is to investigate whether it is better to use kernel functions which favor expansion term occurrence in close proximity to query terms or not.

6.5.3 Calculating Document Relevance Scores

In this section, we intend to calculate the overall probability that relevant expansion concepts (inside the document) are related to the technical concept of the query. This probability is denoted by $P(q|D,e)$, which is defined as:

$$P(q|D,e) = \sum_{i=1}^{|D|} P(q,i|D,e) = \sum_{i=1}^{|D|} P(q|i,D,e)P(i|D,e) \tag{6.3}$$

We assume e and q to be conditionally independent given their positions in document D. Thus, $P(q|i,D,e)$ reduces to $P(q|i,D)$ which can be estimated using the query relatedness probability. We now need to estimate $P(i|D,e)$. To do this, we suggest two different methods.

- **Avg Position Strategy**: all positions of expansion concepts are equally important:

$$\begin{cases} P(i|D,e) = 1/|pos(e)| & \text{if } t_i \in e \\ 0 & \text{otherwise} \end{cases}$$

by substituting this in Equation 6.3 we have:

$$P(q|D,e) = 1/|pos(e)| \sum_{i \in pos(e)} P(q|i,D) \tag{6.4}$$

where $|pos(e)|$ denotes the number of occurrences of expansion term e in document D.

- **Max Position Strategy**: as an alternative, we only consider the expansion term position with the highest $P(q|i,D)$ as important, so:

$$P(q|D,e) = \max_{i \in pos(e)} P(q|i,D) \tag{6.5}$$

6.5.4 Normalization

We compare the effect of different normalization methods prior to linear combination using two score normalization methods: MinMax [60] and HIS normalization [5]. These methods are often used in distributed information retrieval. MinMax normalization method shifts and scales scores to be between zero and one. On the other hand, HIS normalization estimates a single cumulative density function (CDF) for every search engine based on historical queries.

We also experimented with a variation of score normalization where we first applied MinMax and then HIS normalization. We refer to this method as MinMax-HIS throughout the experiments.

6.6 Experimental Results

6.6.1 Building the Initial Query

We built keyword queries by extracting distinguishing terms from the query patent document. To this end, we estimated the importance of each term according to a weighted log-likelihood based approach, as explained in the previous chapter. We selected initial query terms from two different sources: a) all of the claims; b) only the first independent claim. Table 6.3 summarizes the results we obtained for the topics in the training and test set of CLEF-IP 2010 and the test set of CLEF-IP 2011. We used top-10 query terms with higher weights from the estimated query model in our experiments. Results marked with † and ‡ achieve statistically significant improvement in terms of MAP and recall, respectively. Note that this comparison is performed among runs belonging to the same experimental settings.

The results of Table 6.3 demonstrate that the performance of the runs obtained by issuing the query built from the first-claim is always stronger than the performance of the runs where the query is built from the claims in terms of MAP. However, the opposite holds for recall. The reason for the high MAP is because the first independent item of claims is focused on the core invention of

CLEF-IP 2010 (training topics)				
Method	Run description	MAP	Recall	PRES
C10TR	claims	0.1211	0.6302 ‡	0.5492
FC10TR	first-claim	0.1530 †	0.6015	0.5479
CLEF-IP 2010 (test topics)				
Method	Run description	MAP	Recall	PRES
C10TE	claims	0.1293	0.6067 ‡	0.5140
FC10TE	first-claim	0.1445 †	0.5624	0.4911
CLEF-IP 2011 (test topics)				
Method	Run description	MAP	Recall	PRES
C11TE	claims	0.0823	0.5905 ‡	0.4850
FC11TE	first-claim	0.1198 †	0.5360	0.4538

Table 6.3. Choosing baseline on the two retrieval collections.

the patent document. However, we are losing some information by ignoring the text of the rest of the claims and this explains the low recall of this setting.

To guarantee the assignment of reliable importance weights to the expansion concepts in our proximity-based framework, we need to start with a set of precise query terms. This is because we rely on the distance between query terms and expansion concepts to calculate importance weights for expansion concepts. Obviously starting with focused and less noisy query terms has a direct effect on the quality of calculated importance weights. Thus, in the remainder of this chapter, we focus on selecting query terms from the first independent item of the claims.

We used the Language Modeling approach with Dirichlet smoothing [112] to score documents from both collections and to build the initial rank lists. We empirically set the value for the smoothing parameter μ to 1500. We also used Language Modeling for the re-ranking of the results. We note that we do not use citation information in our experiments.

6.6.2 Choosing the Baseline

In terms of our comparison baseline, we chose the strongest configuration in terms of PRES from Table 6.3, the retrieval run where query terms are selected from the claims section of the patent document. C10TR is chosen as the baseline for the training topics of CLEF-IP 2010. C10TE is chosen as the baseline for the test topics of CLEF-IP 2010 and C11TE is chosen as the baseline for test topics of

CLEF-IP 2011. Note that the training set of CLEF-IP 2010 is only used for tuning the parameters of the model, thus we will refer to C10TR in such comparisons.

Table 6.4 shows the performance of strong baselines from previous work using external resources for query expansion [38, 69] over CLEF-IP 2010. We presented their performance evaluation in terms of MAP and PRES as the recall values were not reported for the two baselines.

Table 6.4. Strong baselines of the previous work

method	MAP	PRES
baseline1 [38]	0.1278	0.4604
baseline2 [69]	0.1399	0.4860

As we can see from the results of Table 6.4, C10TE is as strong as baseline1 and baseline2 in terms of PRES. This ensures the selection of a strong baseline which will be used in evaluating the performance of our proposed model in the rest of the chapter.

6.6.3 Motivation for Using Proximity Information

In order to test if closeness of expansion concepts to query terms is correlated with relevance, we carry out preliminary experiments on the CLEF-IP 2010 col-

Table 6.5. Recall results of different settings of the kernel functions using IEC query reformulation methods on the training topics of CLEF-IP 2010.

IEC				
kernel \ σ	25	75	125	150
Gaussian	**0.6443**	**0.6561** †	**0.6676** †	**0.6795** †
Laplace	0.6422	0.6556 †	0.6588 †	0.6709 †
Rectangle	0.6398	0.6523	0.6559 †	0.6678 †

Table 6.6. Recall results of different settings of the kernel functions using EEC query reformulation methods on the training topics of CLEF-IP 2010.

EEC				
kernel \ σ	25	75	125	150
Gaussian	**0.6388**	**0.6418**	0.6669 †	**0.6637** †
Laplace	0.6362	0.6390	**0.6685** †	0.6516
Rectangle	0.6339	0.6375	0.6642 †	0.6497

lection.

In these experiments, we selected 100 random queries. For each query, we first retrieved the top 100 documents using a Language Modeling retrieval method. We separated relevant and non-relevant documents according to relevance judgements (qrels). We then looked at the average distance between query terms and expansion concepts inside the set of relevant documents, denoted by $\mathbb{R}$, and the set of non-relevant documents, denoted by $\overline{\mathbb{R}}$. The distance in each of the two mentioned sets is calculated as follows:

$$DIS(Q,\mathbb{R}) = \sum_{D\in\mathbb{R}} \frac{\sum_{q\in Q} min_{e\in E}(Distance(q,e))}{|Q||\mathbb{R}|}$$

$$DIS(Q,\overline{\mathbb{R}}) = \sum_{D\in\overline{\mathbb{R}}} \frac{\sum_{q\in Q} min_{e\in E}(Distance(q,e))}{|Q||\overline{\mathbb{R}}|}$$

where q denotes a query term drawn from the set of query terms Q, e denotes an expansion term, and E the set of expansion concepts selected from the conceptual lexicon. $Distance(q,e)$, the distance between q and e, is calculated according to the positional difference of q and e in document D – this distance is calculated according to the number of terms between q and e. $DIS(Q,\mathbb{R})$ denotes the average distance between query terms and expansion concepts inside the set of relevant documents. While $DIS(Q,\overline{\mathbb{R}})$ denotes the average distance between query terms and expansion concepts inside the set of non-relevant documents.

Figure 6.2 shows the average distance in relevant and non-relevant document sets for each query topic. For clarity purposes, topics are sorted according to $DIS(Q,\mathbb{R})$ value.

It can be seen from Figure 6.2 that the minimum distance between an expansion term and a query term in relevant documents is less than their respective distance in non-relevant documents. Therefore we can use this proximity information to differentiate the relevant documents from non-relevant documents and to improve the ranking of relevant documents.

6.6.4 Effect of Density Kernels

We investigated the effectiveness of different query reformulation methods proposed in Section 6.5.1 for scoring documents in our proximity-based framework. The results of this comparison are summarized in Tables 6.5 and 6.6.

In all the comparisons, our query expansion method which uses *explicit expansion concept* is denoted as EEC. The query expansion method which uses

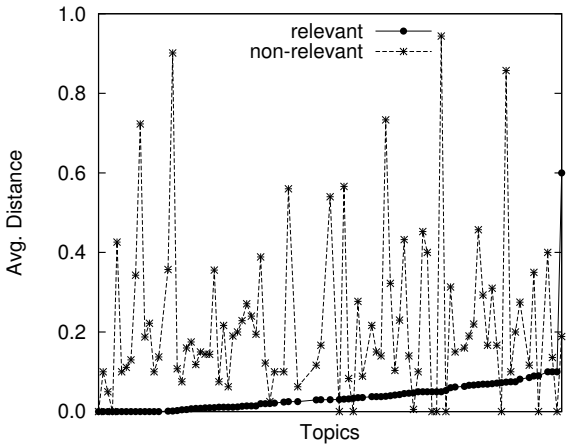


Figure 6.2. *DIS* value for CLEF-IP 2010 query topics in the relevant and non-relevant document sets.

implicit expansion concept is referred to as IEC. Since the performance of these methods is directly determined by the effectiveness of the kernel function used to estimate the propagated query relatedness probabilities for the expansion concepts, we first need to compare three different proximity-based kernel functions to see which one performs the best.

We place a density kernel around each occurrence of query terms in the document, as previously explained in Section 6.5. The query relatedness at each expansion term position is then calculated based on the accumulated query relatedness density from different query terms at that position. Therefore, an expansion term which occurs at a position close to many query terms will receive high query relatedness and thus will obtain a higher importance weight.

Our proximity-based framework has two parameters: the type of kernel function and its bandwidth parameter σ, which controls the degree of query relatedness propagation throughout the entire document. To tune the parameters of our model we used the training topics of CLEF-IP 2010.

The results of comparing different kernel functions on the training topics of CLEF-IP 2010 are shown in Tables 6.5 and 6.6. A † denotes statistical significant improvement over C10TR and the best result for each kernel type is highlighted. The results show that the performance of EEC and IEC with all kernel functions improves over C10TR.

It is also clear that among all the kernel functions, the Gaussian kernel outperforms other types of kernels in most cases. Since the Gaussian kernel performed the best in most of the experiments carried, we use this kernel function

Table 6.7. The performance results of query reformulation approaches on two patent retrieval datasets on the test topics of CLEF-IP 2010 and CLEF-IP 2011.

Collection	metric	IEC	EEC	CSS	PPRF
CLEF-IP 2010	MAP	0.1050	0.1026	0.0982	0.0705
	Recall	0.6595 †	0.6437 †	0.6241	0.5877
	PRES	0.5540	0.5498	0.5354	0.5023
CLEF-IP 2011	MAP	0.0772	0.0761	0.0738	0.0629
	Recall	0.6371 ‡	0.6254 ‡	0.6088	0.5632
	PRES	0.5288	0.5249	0.5127	0.4945

for our system evaluation in the rest of our experiments.

In order to find the best value for the parameter σ we tried a set of fixed values in the range [25, 200] with a step of 25, similar to what done in previous work [66, 67]. Tables 6.5 and 6.6 reports the performance of different kernel functions using varying values of σ. The results show that selecting a value of 125 or 150 usually gives the best retrieval performance.

Overall, Tables 6.5 and 6.6 clearly demonstrate that the results obtained with the σ value of 150 achieved better performance in most cases, although the difference among different settings was not significant. We thus use the σ value of 150 in the rest of our experiments. In Section 6.6.5 we further study the performance of the query reformulation methods.

Comparison of Max and Avg Strategy We are interested to evaluate the two strategies for calculating the probability of relevance of a document as proposed in Section 6.5.3. Table 6.8 shows the result of using avg and max strategies for different sigma values on the training topics of CLEF-IP 2010 using the IEC reformulation method.

The results show that the max strategy is statistically better than the avg strategy. Thus, we use the max strategy in all configurations of our experiments throughout this chapter. A † denotes the statistical significant improvement over the avg method.

6.6.5 Effect of Query Reformulation

In this section, we present the evaluation results of our proposed approaches on the topics in the test set of CLEF-IP 2010 and CLEF-IP 2011. Table 6.7 re-

Table 6.8. Recall of the Max and Avg method using Gaussian kernel with IEC reformulation method on training topics of CLEF-IP 2010.

method \ σ	25	75	125	150
max	0.6443 †	0.6561 †	0.6676 †	0.6795 †
avg	0.6164	0.6198	0.6207	0.6238

ports the retrieval performance of query reformulation methods described in Section 6.5.1. The symbols † and ‡ denote statistical significant improvements over C10TE and C11TE, respectively.

We now compare the performance of our query formulation methods. In addition to EEC and IEC which were introduced earlier, the results of the two other query reformulation methods (described in Section 6.5.1) are presented in Table 6.7. Our method that presents a *combination of search strategies* is referred to as CSS. The last method in our comparisons is the *positional-based pseudo relevance feedback*, which is denoted by PPRF. The number of feedback documents used in both PPRF and PRF is set to 10.

The main observation from Table 6.7 is that IEC is always more effective than the other three methods. In addition, IEC improves the baseline in terms of recall on both collections significantly.

Table 6.7 shows that a method which uses a conceptual lexicon for selecting expansion terms outperforms a method which uses feedback documents for identifying expansion terms. This is evident by comparing the performance of EEC, IEC and CSS to the performance of PPRF, as the first three methods use the conceptual lexicon for query expansion. This result is consistent on both corpora used for evaluation.

In addition, the results of Table 6.7 demonstrate that IEC obtains improvement over EEC. In contrast to IEC, EEC extracts a limited set of expansion terms from the conceptual lexicon, the ones which are present in the query document itself. This diminishes the power of EEC in contrast to IEC, and explains the advantage of IEC. Results confirm that an unlimited usage of the conceptual lexicon is superior to a limited usage.

Another observation which can be made from Table 6.7 is that CSS achieves worst results compared to both EEC and IEC. This is perhaps due to the fact that some information is lost during the combination of two separate runs made from the query terms and expansion terms. While, in EEC and IEC we use a unified query which is composed of query terms and expansion terms.

Overall, the results of Table 6.7 show that using the conceptual lexicon as a

domain-dependent external resource is effective in terms of recall, although this improvement does not hold for precision. Table 6.9 shows some examples of queries for which using IEC reformulation method led to improvement in terms of recall over the initial query.

We used 40 expansion terms (based on initial experiments) in each of the query reformulation methods. We studied the effect of the number of expansion terms on the performance of each method. The result of this study is reported in Section 6.6.7.

Table 6.9. Examples of queries for which IEC reformulation method improved recall.

Example 1: (Topic ID: pac-1474) Patent title: "Optical information recording medium" Query terms extracted from the first independent item of claims: optical, layer, record, lens, light, interlay, irradiation, wavelength Expansion concepts selected from the conceptual lexicon related to the query: organic, dielectric, sensitizing, record, reproduction
Retrieved docs for example 1 Number of retrieved relevant documents in the baseline run: 15/42 Number of retrieved relevant documents after using IEC method: 24/42
Example 2: (Topic ID: pac-552) Patent title: "Power supplying apparatus, design method of the same, and power generation apparatus" Query terms extracted from the first independent item of claims: power, supply, boost, transformer, switch, resonance Expansion concepts selected from the conceptual lexicon related to the query: conversion, semiconductor, electrode, light, push-pull
Retrieved docs for example 2 Number of retrieved relevant documents in the baseline run: 6/15 Number of retrieved relevant documents after using IEC method: 12/15

6.6.6 Comparison to Standard PRF

Table 6.10 reports the retrieval performance of PPRF compared to PRF. A ‡ indicates a statistical significant improvement over the baseline which is built from the first-claim presented in Table 6.3. A † denotes the statistical significant improvement over standard PRF in terms of recall.

As previously explained in Section 6.5.1, PPRF is similar to PRF since they both use the feedback set as a source for selecting expansion terms. However, PPRF uses proximity information inside the feedback set to calculate weights for expansion terms in contrast to standard PRF.

The results show that PPRF performs significantly better than standard PRF. This result confirms the usefulness of proximity information for identifying importance weights for expansion terms, as previously was shown in [67].

Note that PPRF and PRF do not achieve improvement over the baseline, but a fair comparison is to compare the retrieval effectiveness before and after query expansion. We thus need to compare the results of Table 6.10 with the results of FC10TE and FC11TE which correspond to the performance of the initial query built from the first claim.

Our results show that the performance obtained with the PPRF method achieves statistical significant improvements in terms of recall over the initial query (before expansion). This comparison demonstrates the usefulness of aggregating proximity information in the calculation of the expansion weights, as performed in our proximity-based framework.

6.6.7 Influence of Different Parameter Settings

Finally, we are interested to study the influence of different parameters on the effectiveness of our proposed methods. We used the test topics of both test collections (CLEF-IP 2010 and CLEF-IP 2011) in this study.

Table 6.10. The comparison of performance results of PRF and PPRF.

Collection	metric	PPRF	PRF
CLEF-IP 2010	MAP	0.0705	0.0650
	Recall	0.5877 ‡†	0.5630
	PRES	0.5023	0.4961
CLEF-IP 2011	MAP	0.0629	0.0617
	Recall	0.5632 ‡†	0.5346
	PRES	0.4945	0.4792

Number of Expansion Terms. To see the effect of the number of expansion terms on the effectiveness of our proposed methods we plot the sensitivity of different query reformulation methods to the number of expansion terms over CLEF-IP 2010 test topics. We vary the number of expansion terms from 1 to 50.

The recall results are shown in Figure 6.3. We observe that all four methods achieve effective performance using around 40 expansion terms.

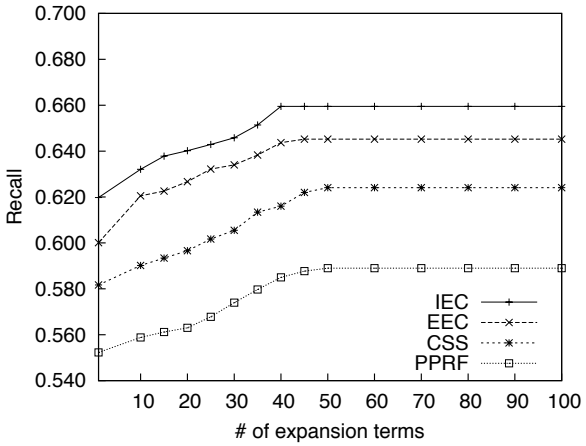


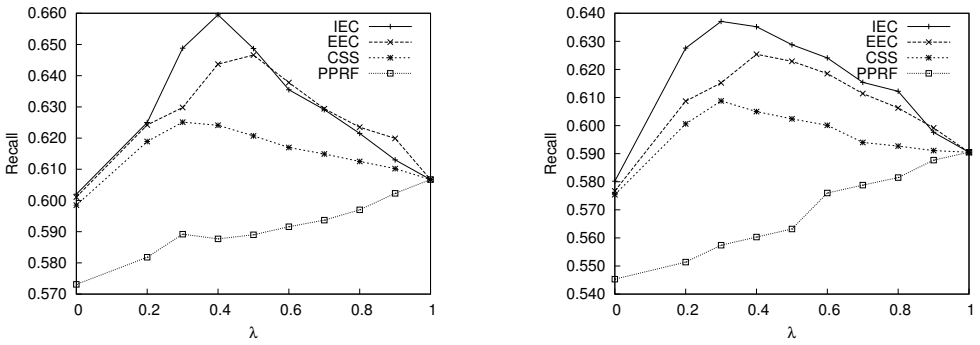
Figure 6.3. Sensitivity to the effect of number of expansion terms on CLEF-IP 2010

Effect of Rank list Combination. In all configurations of our experiments we linearly combined the results we got from each of the reformulation methods with the initial query. The weight of the interpolation λ controls the weight of the initial query. When $\lambda = 0$, the query expansion model is used and when $\lambda = 1$ the initial query is used. λ was tuned based on the training topics of CLEF-IP 2010.

Figure 6.4 shows the results of the sensitivity analysis over the coefficient λ on the test topics of CLEF-IP 2010 and CLEF-IP 2011. We notice that IEC is more effective than other query reformulation methods for different λ values. The optimal value for the parameter λ is around 0.4.

Effect of Normalization. We now compare the effect of different normalization methods prior to linear combination using two score normalization methods, MinMax [60] and HIS [5], which are used in distributed information retrieval or meta-search. MinMax method shifts and scales scores to be between zero and one. HIS method estimates a single cumulative density function (CDF) for every search engine based on historical queries.

We also experimented with a hybrid score normalization technique, in which the scores from each rank list are first normalized using MinMax and then



a) CLEF-IP 2010 b) CLEF-IP 2011

Figure 6.4. Sensitivity to the λ coefficient in the linear combination of results made from the initial and the expanded query.

re-normalized a second time using HIS. We refer to this method as MinMax-HIS throughout the experiments. According to our experiments, the result of the hybrid approach MinMax-HIS was more effective than either MinMax or HIS alone. We thus presented the results of normalization using MinMax-HIS method throughout the chapter.

Table 6.11 shows a comparison among different normalization methods. These results correspond to the final performance of each run after the combination over test topics of CLEF-IP 2010. The results are obtained with the IEC method. The best results are highlighted although the difference is not statistically significant.

Table 6.11. Comparison of different normalization methods over CLEF-IP 2010 using IEC method

metric	MinMax	HIS	MinMax-HIS
MAP	0.0924	0.0991	**0.1050**
Recall	0.6520	0.6568	**0.6595**
PRES	0.5473	0.5522	**0.5540**

We observe that IEC achieves the best performance using MinMax-HIS normalization. The results of other methods were also confirming that applying normalization using MinMax-HIS was better compared to either MinMax or HIS alone, although not significantly.

6.7 Conclusions

In this chapter we introduced a proximity based framework for query expansion which utilizes a conceptual lexicon for patent retrieval. To this end, we constructed a domain-dependent conceptual lexicon which can be used as an external resource for query expansion. Our proximity-based retrieval framework provides a principled way to calculate the importance weights for expansion terms selected from the conceptual lexicon. We showed that proximity of expansion terms to query terms is a good indicator of the importance of the expansion terms. In this chapter we focused on performing query expansion with single terms to ensure the efficiency of the expansion concept selection process.

We have evaluated our proposed method on two patent retrieval corpora, namely CLEF-IP 2010 and CLEF-IP 2011. Our query formulation method, IEC, was shown to outperform the strong baselines of CLEF-IP and the standard pseudo relevance feedback method in terms of recall. Further analysis of the performance of the query reformulation methods proposed in this chapter showed the high quality of expansion terms extracted from the conceptual lexicon.

In the next chapter we will take a closer look into selecting expansion terms from the citation chain of a patent application. Furthermore, we will study whether the vocabulary extracted from citations is complementary to the vocabulary extracted from the query document and the conceptual lexicon.

7

Citation Analysis

"If I have seen further it is by standing on the shoulders of giants."
– Isaac Newton

7.1 Introduction

Behavioral studies of patent examiners in patent offices show that, besides keyword based query and classification based query, other sources that are influencing the most the searching practice of patent examiners are bibliographic information [65]. This includes both backward and forward citations. The question is how can building queries from different information sources such as classifications and citations lend additional power to the initial query itself. Patent authors use an inventive terminology; as a result, there is often a gap between the terms in the query document and the documents relevant to that query [9, 71]. We must cope with the fact that documents relevant to a given (patent) query may not contain the exact terms used by the author of the query patent.

We are interested in overcoming this gap by tapping the power of the community of inventors related to the subject of the invention of the query. To this end, we want to boost the initial query with terms used in the cited documents. In other words, through citation link analysis we want to identify a set of terms which are relevant to a given query document and appear in the cited documents. These terms can be exploited for improving the initial ranking.

In the work presented in this chapter we capture the influence of the citation links in the graph structure of patent documents in two scenarios and compare them. We first use a link-based measure to compute the importance of each doc-

ument in the graph in a topic-sensitive manner. We then use the term distribution of cited documents to estimate a query model from the cited documents by identifying distinguishing terms and their corresponding weights. We perform query expansion using the estimated query model from the cited documents to improve the representation of the initial query.

The rest of this chapter is organized as follows: Section 7.2 explains the construction of a citation graph for a given patent application and describes the citation analysis over the graph. Section 7.3 describes a method to estimate an expanded query model from the cited documents exploiting the citation-based measures. Section 7.4 and 7.5 report how temporal features of documents in the citation graph can be used together with the citation links and the content of cited documents to improve the citation query model. Sections 7.6 presents the results of the experiments aimed at proving the validity of the approach. We next describe the analysis carried out to study the influence of different parameters on the performance of the proposed method. We then present the results of improving the citation query model using temporal features. Finally, Section 7.7 reports the conclusions of the work.

7.2 Query-Specific Citation Graph

In this section we present the basics of representing the patent collection as a directed unweighted graph. Our goal is to find the important documents in the citation graph that could influence their domain terminology.

In the CLEF-IP collections, the citations of query topics (query patents) are removed by the organizers and used for building the query relevance judgements (qrels) which are later used for automatic evaluation of topics. However, we have access to the citations of all other documents apart from the query topics in the collection. A recent work [62] used a web service offered by the European Patent Office[1] to retrieve the citations of documents in the collection. We also used this web service to extract all the citations of the documents in the collection with the exception of the query documents. With these, we can use the citation links and build a graph from the documents in the collection. The assumption is that if a patent is cited by a large number of documents, the cited patent is possibly a foundation of the citing patents and thus is considered important. Therefore, its language might be useful to bridge the gap between the query and its relevant documents.

[1]Available at http://www.epo.org/searching/free/ops.html

As previous work suggested [35], computing Page Rank values as a measure of static quality of the patent documents in the collection (calculated independently of any query a system might receive) has a clear disadvantage compared to conditioning the computation of Page Rank values on the query being served. Thus, we will focus on how to assemble a subset of patent documents around the topic of the query from the graph induced by their citation links. By doing so, we are able to derive Page Rank values relative to particular queries.

To gather a subset of documents that forms the vertices of the topic-specific citation graph we carry out the two following steps.

1. Given a query patent, we perform the search and retrieve an initial rank list of documents. We take the top-k documents from this list and call it the *root set*.

2. We construct the *base set* by expanding the root set with any document that either cites or is cited by a document in the root set.

The subset of selected documents in the root set and base set are considered as a directed unweighted graph $G = (V, E)$, where V is a set of $|V| = N$ patent documents and $E \in V \times V$ is a set of citation relationships between patent documents. Each citation link from document A to document B can be seen as an endorsement of document B. Figure 7.1 illustrates the general scheme of our proposed method of query expansion.

In the first step, we estimate a uni-gram query model from the query patent document using its entire textual content. We create this according to the weighted log-likelihood query model (LLQM) which was previously presented in Chapter 4. In step 2, this query will be used to retrieve an initial set of documents to form the root set. In step 3, we take the initial rank list and build a set from the top-k ranked documents for each query, to which we refer as root set. In step 4, we look up the citation links of documents in the root set, identifying documents that both cite and are cited by the documents in the root set. In step 5, as a result of the expansion of the root set, we obtain a base set. The topic-specific citation graph is constructed by collecting documents in both the root set and the base set. In step 6, we perform influence analysis on the citation graph incorporating the temporal features of the cited documents into our model and we build a citation query model. In step 7, we expand the initial query model with the citation query model. In step 8, the final rank list is generated as a result of the previous step.

We now compute the topic-specific Page Rank values for all nodes in the citation graph.

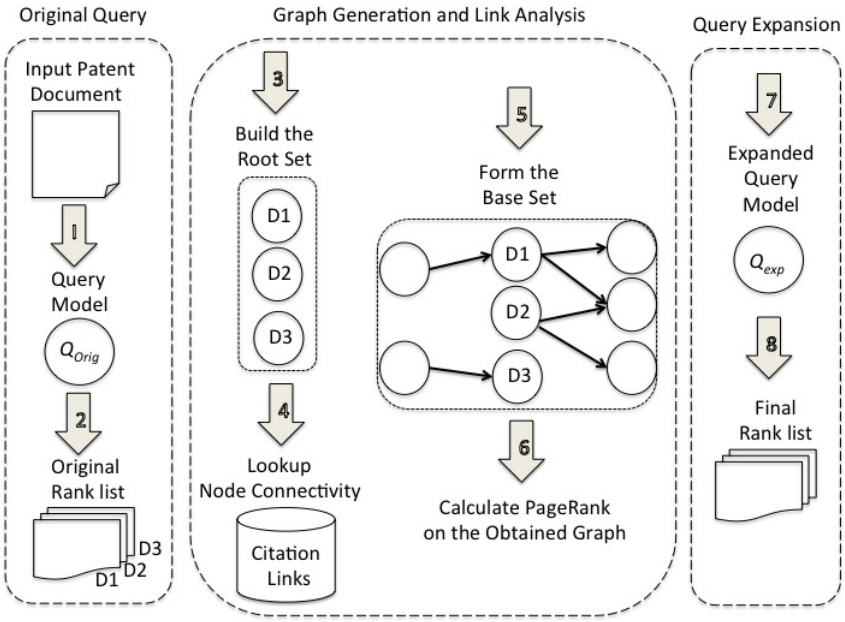


Figure 7.1. The general scheme of our proposed method for query expansion using citation information. Numbers indicate the sequence flow of operations.

7.3 Query Expansion Guided by Page Rank Scores

The computation of Page Rank value for a document D is performed as follows:

$$PR(D) = \sum_{x \in d_{*\to D}} \frac{PR(x)}{d_{x\to *}} \tag{7.1}$$

where $d_{*\to D}$ is a set of patent documents that cites D, and $d_{x\to *}$ is a set of patent documents cited by D. If D is cited by a large number of documents, a high score is given to D. However, if a document cites n documents, the value for each cited document is divided by n [14].

We calculated the Page Rank values for all the documents in the topic-specific citation graph. We now explain how this value is used to guide the priority assignment to documents while estimating a query model from citation graph.

Our approach for query expansion aims to improve the language model of the initial query model by using the term distribution of documents in the citation graph. The key assumption of this approach is that the term mismatch can be

alleviated by using the term distribution of documents with higher Page Rank scores.

We identify and weigh the most distinguishing terms in the documents belonging to the citation graph and we use the calculated Page Rank values as document prior in a language modeling framework. The term sampling is performed as follows:

$$P(t|Q_{cit}) = Z_t \sum_{D \in G_{cit}} P(t|D)P(D) \tag{7.2}$$

where G_{cit} denotes the citation graph and $P(D)$ indicates the Page Rank score of document D calculated according to Equation 7.1 after normalization. Z_t is the normalization factor.

7.4 Temporal Analysis of the Citation Graph

After conducting our citation analysis using Page Rank scores, we noticed that Page Rank is assigning a higher score to older documents. To investigate whether the language of the query is more susceptible to the terminology of older documents, we looked into the relationship between relevance and time and studied how relevance changes over time using time series.

For this analysis we focused only on the result set of a query and not all the documents in the collection. We derived time series from the result set (which could be relevant to the query, thus referred to as pseudo relevant documents) and in parallel from the set of relevant documents (qrels). We then compared these two time series. We consider the publication date of the first kind-code of a patent application as the time tag. A kind-code is a version used to denote the publication level of a patent (e.g., first publication, second publication, or corrected publication). The unit of time granularity considered in our analysis is a year. We thus aggregated in one bin documents with publication dates in the same year.

After performing this analysis, we observed that for the majority of queries, the temporal distribution of true relevant documents (qrels) has a higher density of documents with recent publication dates, while our result set contains a higher number of documents with older publication dates. This means that the pseudo relevant set is lagging behind the qrels in the time dimension. Likewise, citation influence analysis using Page Rank scores is biased towards the terminology of older documents.

We are thus interested to take into account the time dimension in order to improve the effectiveness of the retrieval. To do this, we need a query model that captures the established terminology (derived from older documents) but at the same time encodes the new vocabulary of the field (which is led by recent documents). The challenge is how to balance these two distinct terminologies and build a query model that combines both of these terminologies at once.

Modeling decay over time. Our aim is to capture the language change over time. We take into account the patent publication dates and prioritize recent documents while penalizing older documents.

In previous work, different functions have been used to model the decay over time in a retrieval setting [3, 80]. Exponential decay function has been used previously in IR tasks for modeling the time decay [61]. Recently, inspired by cognitive psychology, Weibull function has been introduced as a time-aware prior and has been successfully employed on blog and news collections for improving the query modeling of event-based queries. Weibull function has been shown to be more effective compared to exponential decay according to the retrieval results obtained in [80]. In this work we consider time-aware functions to discount the effect of older documents and capture the terminology of recent documents in the query model.

We describe the two time-aware functions below.

- Exponential Decay

$$f_{Exp-Decay}(D,q,g) = \hat{\mu} e^{-\hat{\mu}\ \delta_g(q,D)} \tag{7.3}$$

- Weibull

$$f_{Weibull}(D,q,g) = e^{-(\frac{\hat{\mu}\ \delta_g(D,q)}{\hat{d}})^{\hat{d}}} \tag{7.4}$$

where $\delta_g(q,D)$ is the difference between the publication date of the query and the publication date of the document D. $\hat{\mu}$ denotes the decay parameter, $\hat{d}$ indicates the steepness of the decay (forgetting) function, and g denotes the time granularity.

We identify and weight the most distinguishing terms in the documents in the citation graph, prioritizing recent documents. We consider a granularity of one year and employ time-aware functions as document priors.

$$P(t|Q_{innov-cit}) = Z_t \sum_{D \in G_{cit}} P(t|D)P(D) \tag{7.5}$$

where G_{cit} is the citation graph. The document prior component in Equation 7.5, $P(D)$, is proportional to the value calculated by the exponential decay function or Weibull function. The weight of each document, using the exponential decay function, is calculated as follows: $P(D) = \frac{f_{Exp-Decay}(D,q,g)}{\sum_{D \in G_{cit}} f_{Exp-Decay}(D,q,g)}$. Z_t is a normalization factor.

Our assumption is that recent documents have an innovative language and using temporal priors allow us to capture the terminology of recent documents.

7.5 Query Expansion using Citation Graph and Temporal Features

We build a query model that has a good coverage over different time intervals, utilizing the established language usage of older documents and the innovative language usage of recently published documents. This query is built from the linear combination of the initial query, the citation query model using the Page Rank scores, and the temporal query model of the citation graph.

We interpolate the temporal query (as estimated in Equation 7.5) with the citation query (as estimated in Equation 7.2) and the initial query (as estimated in Equation 4.2):

$$P(t|Q) = \alpha\, P(t|Q_{Init}) + \beta\, P(t|Q_{cit}) + (1 - \alpha - \beta)\, P(t|Q_{innov-cit}) \tag{7.6}$$

The M highest terms from the updated query model is then used as a query to retrieve a rank list of documents.

Query Document	Citation Graph (Page Rank)	Citation Graph (Temporal)
manage, server, collaborate, client, soap, peer, ...	transact, handle, service, access, command, ...	network, permission, secure, request, collect, ...

Table 7.1. Comparing the query terms selected from topic EP-1832953-A2 and the topic-specific citation graph.

Table 7.1 shows a comparison between a list of terms derived from the patent application “EP-1832953-A2”, terms sampled from documents with high Page Rank scores belonging to the topic-specific citation graph, and terms derived from the temporal query model of the citation graph. This query topic

belongs to CLEF-IP 2011 topic set. The title of this patent topic is “Method and apparatus for managing a peer-to-peer collaboration system”. By looking at this example, we see that we are able to select terms from documents in the citation graph which are relevant to the topic of the query but are not captured by the initial query model.

7.6 Experimental Results

We now describe the structure of the experimental evaluations. We compare the two following methods with the baseline presented in Table 7.2. The first method corresponds to our implementation of the work reported in [35]. This method is focused on computing a composite score using the textual information of the query together with the link-based structure of the query-specific citation graph. This method is referred to as *Score-cit*. The second method is our proposal which estimates a query model from the documents in the citation graph and expands the initial query using the estimated model from the term distribution of the documents in the citation graph. This method is referred to as *QM-cit*. Table 7.3 and 7.4 show the evaluation results of different methods using the CLEF-IP corpora.

Results marked with † represents a statistical significant difference compared to Score-cit1 and Score-cit2. The reported results for QM-cit1 and QM-cit2 are obtained using the top 100 feedback terms selected from the expanded query model. The top 30 feedback documents are selected and used to generate the root set. The number of feedback terms and number of feedback documents are experimentally set with the goal of optimizing the performance of the method.

We study the influence of the size of the citation graph on the effectiveness of query expansion by considering two alternative versions of Score-cit and QM-cit. The first version considers a citation graph exploiting one level depth of citation links, constructed by collecting documents in the root set and base set as explained in Section 7.2. We call these methods Score-cit1 and QM-cit1. The second variation takes into account a citation graph using two levels of citation links. We refer to the methods in this category as Score-cit2 and QM-cit2.

The results of Tables 7.3 and 7.4 suggest that the QM-cit method obtained a better performance compared to Score-cit in terms of both recall and precision. QM-cit2 obtained statistical significant improvement in terms of recall over Score-cit1 and Score-cit2. This observation suggests that using the link-based structure as well as exploiting the term distribution of the citations (through estimation of a query model) is more useful than using the citation links alone.

Table 7.2. Choosing baselines on two retrieval collections.

CLEF-IP 2010 (training topics)			
Run identifier	MAP	recall	PRES
W10TR	0.1219	0.6367	0.5512
CLEF-IP 2010 (test topics)			
Run identifier	MAP	recall	PRES
W10TE	0.1295	0.6105	0.5150
CLEF-IP 2011 (test topics)			
Run identifier	MAP	recall	PRES
W11TE	0.0990	0.5935	0.4859

Table 7.3. Performance of different citation analysis methods with a cut off value of 1000.

CLEF-IP 2010 test set				
Method	Run description	MAP	recall	PRES
Score-cit1	citation depth level 1	0.102	0.567	0.449
Score-cit2	citation depth level 2	0.105	0.574	0.461
QM-cit1	citation depth level 1	0.118	0.580	0.469
QM-cit2	citation depth level 2	0.121	0.585	0.474

We can see from the results of Tables 7.3 and 7.4 that neither of the versions of Score-cit nor QM-cit achieved statistical significance over the baselines W10TE and W11TE.

The results presented in Tables 7.3 and 7.4 show that increasing the depth of the citation graph (from depth 1 to depth 2) has a positive effect on the performance of both Score-cit and QM-cit methods. We also carried out experiments with a citation graph of depth 3, where three consecutive iterations of the steps described in Section 7.2 are considered. The obtained performance is statistically indistinguishable from the results for Score-cit2 and QM-cit2. We therefore did not present these results.

In Figure 7.2, we studied the effect of increasing feedback terms and feedback documents on the performance of QM-cit2. We notice that increasing the number of feedback terms has a consistent positive effect on all evaluation met-

Table 7.4. Performance of different citation analysis methods with a cut off value of 1000.

CLEF-IP 2011 test set				
Method	Run description	MAP	recall	PRES
Score-cit1	citation depth level 1	0.091	0.543	0.453
Score-cit2	citation depth level 2	0.095	0.550	0.459
QM-cit1	citation depth level 1	0.105	0.560	0.465
QM-cit2	citation depth level 2	0.105	0.579 †	0.481

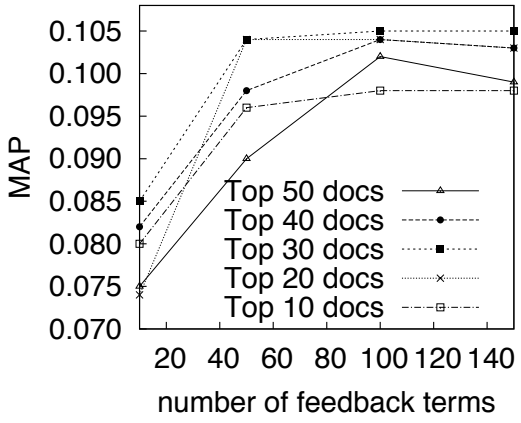
rics. However, when we vary the number of feedback documents, we can see from the curve trends of QM-cit2 that there is a marked drop of performance in terms of MAP for values more than 30. We observe a less severe drop of performance in terms of recall. We can conclude that recall is less susceptible to the number of feedback documents than MAP. By looking at PRES values presented in Figure 7.2 we can see that the best performance of QM-cit2 is obtained with 30 feedback documents and 100 feedback terms.

7.6.1 Sensitivity Analysis of Different Parameter Settings

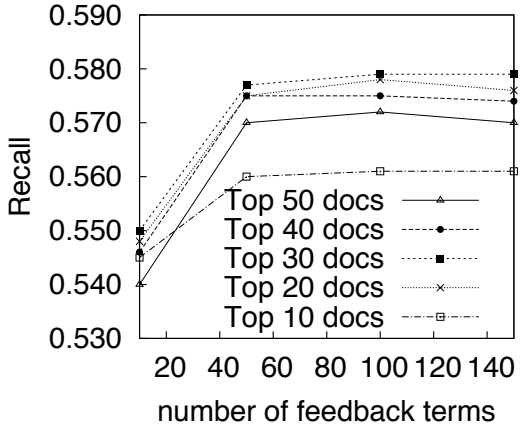
As mentioned before, the reported results in Table 7.4 are obtained using the top 100 terms extracted from the expanded query model using the top 30 feedback documents. In this section, we conduct experiments to study the impact of these parameters on the retrieval effectiveness of our proposed method QM-cit2. We used the test topics of CLEF-IP 2011 during these evaluations.

Effect of the Number of Query Terms. We study the impact of the number of query terms selected from the initial query model on the retrieval effectiveness of the baseline method. Figure 7.3 shows the results of this study. We can observe that by increasing the number of query terms we achieve improvement in terms of recall. In Figure 7.3, the best performance in terms of recall is achieved when the number of query terms is around 100. On the other hand, MAP drops when selecting more than 100 query terms.

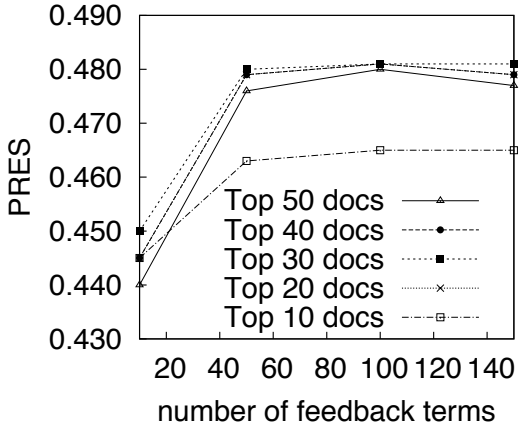
Effect of Number of Feedback Terms. We run QM-cit2 by varying the number of feedback terms from 10 to 150. Table 7.5 shows the effect of these parameters on the performance of the system in terms of MAP, recall, and PRES at cut-off value of 1000. Results marked with † indicate statistically significant improve-



(a) MAP@1000

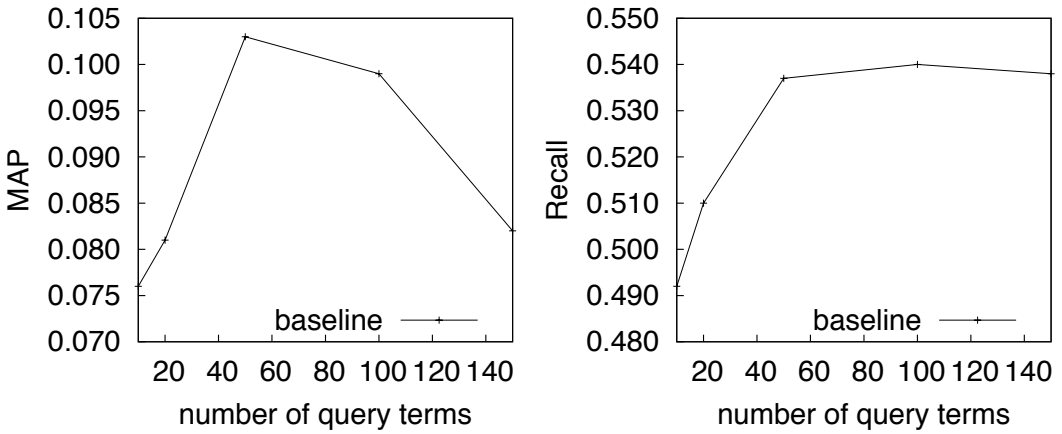


(b) recall@1000



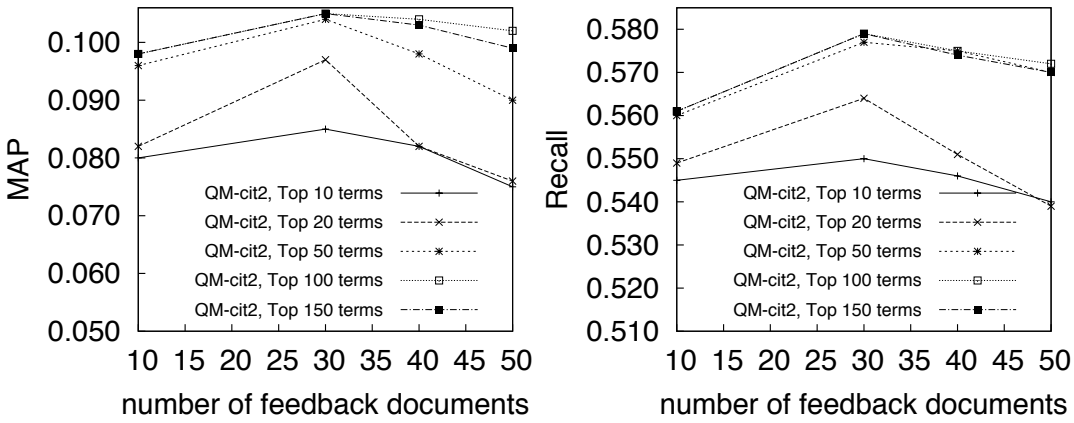
(c) PRES@1000

Figure 7.2. Sensitivity analysis of QM-cit2 to the number of feedback terms on CLEF-IP 2011.



(a) MAP@1000 of baseline (b) Recall@1000 of baseline

Figure 7.3. Sensitivity analysis of the baseline method to the number of query terms selected from the initial query model on CLEF-IP 2011.



(a) MAP@1000 on CLEF-IP 2011 (b) Recall@1000 on CLEF-IP 2011

Figure 7.4. Sensitivity analysis of QM-cit2 to the number of feedback documents on CLEF-IP 2011.

ment over the baseline. We observe that QM-cit2 achieves the best performance, selecting 100 feedback terms, regardless of the number of feedback documents. In Table 7.5, selecting more than 100 query terms does not lead to an improvement. We notice the positive effect of increasing the number of expansion terms

Feedback terms	Metric	Feedback documents				
		10	20	30	40	50
10	MAP	0.080	0.074	0.085	0.082	0.075
	Recall	0.545	0.548	0.550	0.546	0.540
	PRES	0.445	0.445	0.450	0.445	0.440
20	MAP	0.082	0.082	0.097	0.082	0.076
	Recall	0.549	0.551	0.564	0.551	0.539
	PRES	0.447	0.451	0.467	0.451	0.440
50	MAP	0.096	0.104	0.104	0.098	0.090
	Recall	0.560	0.575	0.577	0.575	0.570
	PRES	0.463	0.479	0.480	0.479	0.476
100	MAP	0.098	0.104	0.105	0.104	0.102
	Recall	0.561	0.578 †	0.579 †	0.575	0.572
	PRES	0.465	0.481	0.481	0.481	0.480
150	MAP	0.098	0.103	0.105	0.103	0.099
	Recall	0.561	0.576	0.579 †	0.574	0.570
	PRES	0.465	0.479	0.481	0.479	0.477

Table 7.5. QM-cit2 results over CLEF-IP 2011 dataset with a cut-off value of 1000.

on all the evaluation metrics.

Table 7.6 and 7.7 report the results of QM-cit2 in terms of MAP, recall, PRES at cut-off value of 100 and 500, respectively. The results of Table 7.6 and 7.7 are consistent with the observations made from Table 7.5.

Effect of Number of Feedback Documents We investigate the effect of the number of feedback documents by varying this number from 10 to 50. We plot the sensitivity of QM-cit2 method for varying values of feedback documents in Figure 7.4. We observe the best performance is achieved when the number of feedback documents is around 30. We can see that values higher that 30 hurt the performance.

In the next section we show how we can obtain a better performance by incorporating temporal information into the model when performing citation influence analysis.

Feedback terms	Metric	Feedback documents				
		10	20	30	40	50
10	MAP@100	0.076	0.077	0.078	0.082	0.078
	Recall@100	0.300	0.310	0.310	0.313	0.314
	PRES@100	0.221	0.227	0.227	0.230	0.228
20	MAP@100	0.077	0.080	0.080	0.082	0.078
	Recall@100	0.310	0.324	0.325	0.315	0.328
	PRES@100	0.226	0.245	0.245	0.2327	0.247
50	MAP@100	0.078	0.080	0.080	0.084	0.080
	Recall@100	0.327	0.342	0.342	0.354	0.344
	PRES@100	0.247	0.257	0.257	0.262	0.257
100	MAP@100	0.079	0.081	0.081	0.090	0.082
	Recall@100	0.329	0.344	0.344	0.354	0.342
	PRES@100	0.250	0.258	0.258	0.266	0.258
150	MAP@100	0.079	0.081	0.081	0.090	0.082
	Recall@100	0.329	0.344	0.344	0.353	0.342
	PRES@100	0.250	0.258	0.258	0.266	0.258

Table 7.6. Recall, MAP and PRES results over CLEF-IP 2011 dataset with a cut-off value of 100.

7.6.2 Enhancing Citation Analysis with Temporal Information

We study the impact of the temporal features for improving the citation query model and we look into capturing the language change over time. Table 7.8 shows the results of the temporal query model (according to Equation 7.6). Results marked with † show statistically significant improvement over the baselines W10TE and W11TE. The parameters of the model are tuned using 5 fold cross validation to maximize PRES.

The results reported in Table 7.8 show that including temporal features into the citation query model led to statistically significant improvement in terms of MAP. Furthermore, reported results show that modeling the decay over time using the Weibull prior (Equation 7.4) performed better than using the Exponential decay prior (Equation 7.3).

To further investigate the effect of the temporal query modeling, we presented the evaluation results of methods in different technological fields in Tables 7.9 and 7.10.

The results of Table 7.9 shows that TM-WB obtained a better performance

Feedback terms	Metric	Feedback documents				
		10	20	30	40	50
10	MAP@500	0.079	0.078	0.080	0.081	0.082
	Recall@500	0.476	0.482	0.483	0.489	0.489
	PRES@500	0.374	0.380	0.381	0.389	0.389
20	MAP@500	0.080	0.080	0.082	0.082	0.083
	Recall@500	0.495	0.497	0.497	0.490	0.492
	PRES@500	0.394	0.398	0.398	0.389	0.392
50	MAP@500	0.084	0.084	0.086	0.087	0.087
	Recall@500	0.495	0.505	0.505	0.492	0.494
	PRES@500	0.398	0.401	0.401	0.398	0.398
100	MAP@500	0.960	0.977	0.977	0.960	0.960
	Recall@500	0.494	0.508	0.508	0.494	0.493
	PRES@500	0.405	0.411	0.411	0.405	0.405
150	MAP@500	0.968	0.978	0.978	0.968	0.968
	Recall@500	0.494	0.508	0.508	0.494	0.494
	PRES@500	0.407	0.411	0.411	0.407	0.407

Table 7.7. Recall, MAP and PRES results over CLEF-IP 2011 dataset with a cut-off value of 500.

Table 7.8. Performance of temporal modeling with a cut off value of 1000.

CLEF-IP 2010 test set			
Method	MAP	recall	PRES
TM-ED	0.138	0.587	0.496
TM-WB	0.145 †	0.588	0.503
CLEF-IP 2011 test set			
Method	MAP	recall	PRES
TM-ED	0.124 †	0.580	0.487
TM-WB	0.128 †	0.582	0.490

compared to QM-cit2 in categories F, G and H. These improvements hold for all three reported metrics. However, we observe that the performance of TM-WB is lower than QM-cit2 in category C. Our experiments on CLEF-IP 2010 showed that communities of inventors related to the topics in categories F, G and H are

Table 7.9. Evaluation results over test set of CLEF-IP 2010.

method name	metric	A	B	C	D	E	F	G	H
QM-cit2	MAP	0.138	0.123	0.127	0.070	0.086	0.128	0.110	0.119
	recall	0.518	0.598	0.535	0.620	0.500	0.619	0.600	0.598
	PRES	0.433	0.489	0.448	0.540	0.427	0.517	0.476	0.482
TM-WB	MAP	0.134	0.122	0.121	0.075	0.081	0.148	0.117	0.134
	recall	0.518	0.596	0.533	0.620	0.500	0.638	0.624	0.599
	PRES	0.433	0.489	0.439	0.542	0.427	0.529	0.500	0.495

Table 7.10. Evaluation results over test set of CLEF-IP 2011.

method name	metric	A	B	C	D	E	F	G	H
QM-cit2	MAP	0.129	0.099	0.132	0.195	0.125	0.096	0.103	0.073
	recall	0.603	0.566	0.568	0.676	0.438	0.569	0.584	0.545
	PRES	0.509	0.471	0.485	0.588	0.370	0.490	0.483	0.440
TM-WB	MAP	0.127	0.102	0.130	0.195	0.137	0.120	0.105	0.101
	recall	0.603	0.568	0.565	0.679	0.446	0.579	0.585	0.558
	PRES	0.509	0.472	0.483	0.588	0.376	0.488	0.485	0.455

more receptive to linguistic changes over time while inventions in categories A, B, C, D, and E are more resistant to the changes of the language. The decrease of precision on category C (which categorizes the patent documents related to “Chemistry” and “Metallurgy”) was counter intuitive as we expected the language of this community to be evolving over time. However, we did not capture this effect in our query model. The results of Table 7.10 show the positive effect of our method in capturing the language change in categories E, F and H as opposed to other categories over the topics of CLEF-IP 2011. A concluding remark for these experiments is that we obtained more accurate results when incorporating temporal features into our model.

7.7 Conclusions

Previous work showed that using the link-based structure of citations led to improvements over a strictly textual-based method (using the term distribution of the query document). The question which was not answered is whether the

link-based structure of the citation graph together with the term distribution of cited documents can be effective to improve the ranking. To answer this question in this chapter we used the link-based structure of the citation graph together with the term distribution of cited documents and we built a query model from the citation graph.

In order to address the bias of the Page Rank score against new nodes – which have not stayed long enough in the network to accumulate sufficient links – we incorporated the publication dates of the patent documents in the citation graph in the query modeling process.

We did this by considering the temporal order of the nodes in the citation network. We discounted the initial probability of selecting a node as the seed of the PageRank algorithm according to some temporal decay factor. The results showed the advantage of using the term distribution of the cited documents together with the publication dates. In particular, our citation influence analysis using temporal features improved the precision. It is worth mentioning that the positive effect of capturing the language change using the temporal query was more visible for patents belonging to domains such as “Mechanical Engineering” and “Electricity”, while we observed a decrease in precision for topics belonging to “Chemistry”.

In the next chapter, we investigate combining in a single framework different relevance evidences related to a query patent such as classification (explained in Chapter 6) and citation information (explained in this chapter).

8

Synthesizing Multiple Relevance Evidences for Query Formulation

"Everything should be as simple as it is, but not simpler."
– Albert Einstein

8.1 Introduction

Up to now, we introduced multiple sources of relevance evidence (i.e., classification tags, bibliographical data, and temporal information) and discussed how to incorporate them in the process of query formulation. We saw that such sources of relevance evidence provides additional details about the underlying information need. In this chapter, we take a broader look and propose a unified framework that integrates the previously mentioned relevance evidences for query formulation.

Our proposed model is composed of different stages which were previously introduced. In the first step, we take advantage of the estimated query model from the textual fields of a query patent, according to a weighted log-likelihood based approach (see Section 4.2). In the second step, we employ the topic specific citation graph and use the estimated citation query model (according to Equation 7.6) to expand our initial query model. We then perform a query expansion by deriving expansion concepts from the query-specific lexicon (built in Section 6.4) and use positional information to calculate weights for ensuring high quality expansions. To this end, we utilize kernel functions to keep track of the distance of expansion concepts from query terms (as previously discussed in Section 6.5). To avoid redundancy we do not repeat the explanations of the

previous methods in this chapter and we only present the result of combining different components in the experimental section.

The rest of this chapter is organized as follows. Section 8.2 presents the architecture of our proposed model. Section 8.3 present the experimental results. We conclude in Section 8.4 with a summary.

8.2 The Architecture of the Proposed Model

Figure 8.1 illustrates the general scheme of our proposed method of query expansion. The system receives a full patent application (query patent) consisting of textual fields and classification information. Note that we do not have the citation information associated with the patent application; however, for the rest of the documents in the collection, we have both classification information and citation information.

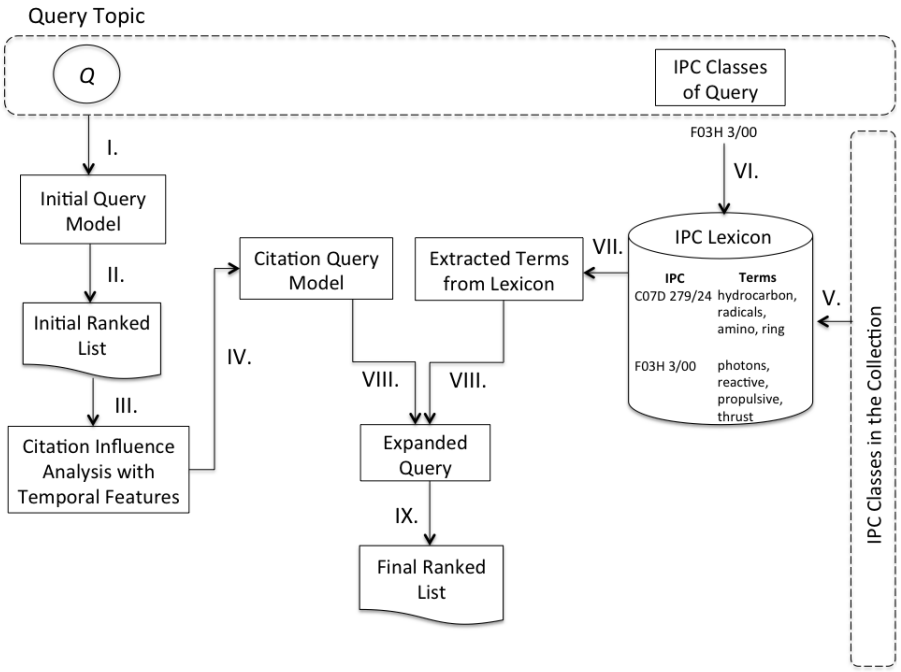


Figure 8.1. The general scheme of our proposed method for query expansion using IPC lexicon and citation information. Numbers indicate the flow of operations.

In step I, we estimate a query model from the textual fields of the patent. In step II we build an initial rank list based on the query model estimated from the

query patent. In step III we take the initial rank list and extract query-dependent citation links from the top-k ranked documents. We then build a query-specific citation graph. We perform influence analysis on the citation graph incorporating the temporal features of the cited documents into our model. In step IV we build a citation query model. In step V we build a query-specific lexicon from the IPC definition pages. In step VI we make a lookup in the lexicon using the IPC classes of the query document. In step VII we extract the terms related to the IPC classes of the query from the IPC lexicon. In step VIII we expand the citation query model with expansion concepts extracted from the lexicon. In step IX, query expansion is performed and the distance between query terms and expansion terms is used to calculate weights for ensuring high quality expansion. The final rank list is generated as the result of this step.

8.3 Experimental Results

To guarantee the assignment of reliable importance weights to the expansion concepts, we start with a set of precise query terms according to the interpolated citation query model (Equation 7.6). Thus, in the remainder of the experiments, we focus on this query model. Note that Equation 7.4 is used as document prior for the temporal component.

In this section, we present the evaluation results of our proposed approaches on the test set of CLEF-IP 2010 and CLEF-IP 2011. Table 8.1 reports the retrieval performance of query reformulation methods described in Section 6.5.1. The symbols † and ‡ denote statistical significant improvements over W10TE and W11TE (presented in Table 6.3), respectively.

Table 8.1. Performance results of query reformulation approaches on two patent retrieval datasets on the test topics of CLEF-IP 2010 and CLEF-IP 2011.

Collection	metric	IEC	EEC	CSS	PPRF
CLEF-IP 2010	MAP	0.1434 †	0.1405 †	0.1301	0.1122
(baseline:	recall	0.6598 †	0.6452 †	0.6243	0.5890
W10TE)	PRES	0.5560 †	0.5510 †	0.5338	0.5029
CLEF-IP 2011	MAP	0.1231 ‡	0.1225 ‡	0.1189	0.1022
(baseline:	recall	0.6369 ‡	0.6268 ‡	0.6094	0.5645
W11TE)	PRES	0.5290 ‡	0.5255 ‡	0.5141	0.4952

We now compare the performance of our query expansion methods which use IPC lexicon for extracting expansion candidates. In addition to EEC and IEC which were introduced earlier, the results of the other two query reformulation methods are presented in Table 8.1. The method that *combines search strategies* is denoted as CSS. The last method in our comparison is the *positional-based pseudo relevance feedback*, which is denoted by PPRF.

The main observation from Table 8.1 is that IEC is always more effective than the other three methods. In addition, IEC improved significantly over the baseline in terms of recall on both collections.

Table 8.1 shows that a method which uses a conceptual lexicon for selecting expansion terms outperforms a method that uses pseudo feedback documents (selected using PRF) for identifying expansion terms. This is evident by comparing the performance of EEC, IEC and CSS to the performance of PPRF, since the first three methods use the conceptual lexicon for query expansion. This result is consistent on both corpora used for evaluation.

In addition, the results of Table 8.1 demonstrate that IEC obtained improvement over EEC. In contrast to IEC, EEC extracts a limited set of expansion terms from the conceptual lexicon, the ones which are present in the query document. This diminishes the power of EEC with respect to IEC. The results confirm that the unlimited usage of the conceptual lexicon is superior to its limited usage.

Another observation which can be made from Table 8.1 is that CSS achieved worse results compared to both EEC and IEC. This is perhaps due to the fact that information is lost during the merging of two separate runs made from the query terms and expansion terms. On the contrary, both EEC and IEC use a unified query which is composed of query terms and expansion terms. Overall, the results of Table 8.1 show that using the conceptual lexicon as a domain-dependent external resource is effective in terms of recall and precision.

We used 40 expansion terms (experimentally set as described in Section 8.3.2) in each of the query reformulation methods. In Section 8.3.2, we studied the effect of varying the number of expansion terms and number of feedback documents on the performance of each method. We also presented the results of normalization using MinMax-HIS throughout this chapter.

Table 8.2 shows the performance of the IEC method along with the best official results of CLEF-IP 2010[1] and CLEF-IP 2011 [82, 83]. We are afraid that PRES values were not reported in the official results of CLEF-IP 2011 and they can't be calculated now. It can be seen that the IEC method performed better than the best official results over CLEF-IP 2011. IEC can also be considered as

[1]http://www.ifs.tuwien.ac.at/∼clef-ip/pubs/CLEF-IP-2010-IRF-TR-2010-00003.pdf

the second best method on CLEF-IP 2010 in terms of recall and PRES.

Table 8.2. Comparison with the best official results on the English subset of the test set.

Official best results of CLEF-IP 2010				
Method	Run description	MAP	recall	PRES
IEC	our method	0.1434	0.6598	0.5560
humb	rank 1	0.2264	0.6946	0.6149
dcu	rank 2	0.1807	0.616	0.5167
Official best results of CLEF-IP 2011				
Method	Run description	MAP	recall	PRES
IEC	our method	0.1231	0.6369	0.5290
nijm	rank 1	0.0582	0.6303	NA
hyder	rank 2	0.0593	0.5713	NA

8.3.1 Comparison with Standard PRF

Table 8.3 reports the retrieval performance of PPRF and PRF. A † denotes statistical significant improvement over standard PRF.

Table 8.3. The comparison of performance results of PRF and PPRF.

Collection	metric	PPRF	PRF
CLEF-IP 2010	MAP	0.1122	0.0880
	recall	0.5890†	0.5630
	PRES	0.5029	0.4962
CLEF-IP 2011	MAP	0.1022	0.0842
	recall	0.5645 †	0.5348
	PRES	0.4952	0.4794

As previously mentioned, PPRF is similar to PRF since they both use a feedback set for selecting expansion terms. However, PPRF uses proximity information inside the feedback set to calculate the weight for expansion terms in

contrast to standard PRF. The results show that PPRF performs significantly better than standard PRF. This result confirms the usefulness of proximity information for identifying importance weights for expansion terms as previously shown in [67]. PPRF and PRF did not achieve improvement over the baseline (presented in Table 6.3). The number of feedback documents is experimentally set (as described in Section 8.3.2) to 10 for both PPRF and PRF methods.

8.3.2 Parameter Study

In this section, we study the influence of different parameters on the effectiveness of our proposed methods.

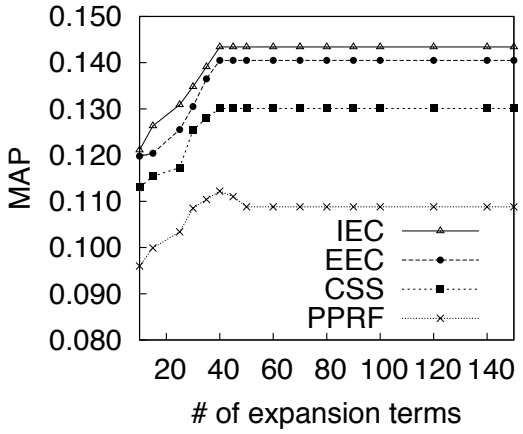
Number of Expansion Terms and Number of Feedback Documents.

We plot the sensitivity of different query reformulation methods in relation to the number of expansion terms over CLEF-IP 2010 test set in Figure 8.2.

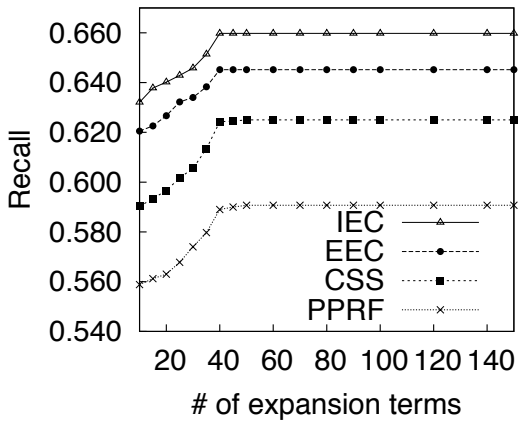
According to Figure 8.2, IEC is a clear winner among the four methods given the three evaluation metrics and PPRF achieved inferior results compared to other methods. We observe some variations in the performance of PPRF with different number of expansion terms. The best performance of PPRF is achieved with 40 expansion terms. Another observation is that IEC, EEC, and CSS seem to be less susceptible to the number of expansion terms. We can see that IEC, EEC, and CSS need 40 expansion terms to exhibit their best performance according to PRES values. IEC, EEC, and CSS continue to maintain a stable performance using higher number of expansion terms.

Since PRF and PPRF share the number of feedback documents, we are interested to understand how this parameter affect the retrieval performance of these two methods. We draw in Figure 8.3 the sensitivity curves of PRF and PPRF with respect to the number of feedback documents and expansion terms on CLEF-IP 2010. Since IEC, EEC and CSS do not share the number of feedback documents as a parameter, we did not include them in this analysis.

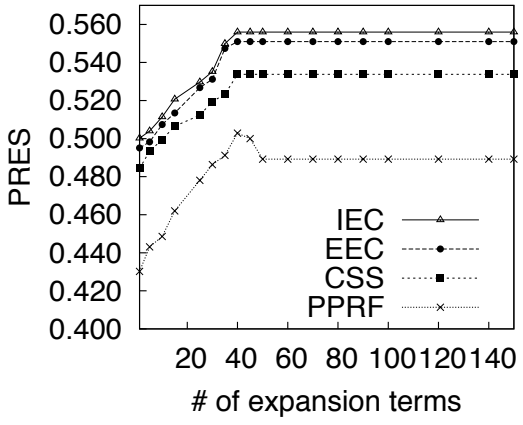
Figure 8.3 shows that PPRF achieved better results compared to PRF. The best performance values for both PRF and PPRF are obtained with 10 feedback documents according to PRES values. The sensitivity curves for both PRF and PPRF show that using more than 10 feedback documents does not improve the performance. We hypothesize that this is because when we select feedback documents with higher rank positions in the rank list, more noisy terms are also selected and this hampers the performance of PRF and PPRF.



(a) MAP@1000

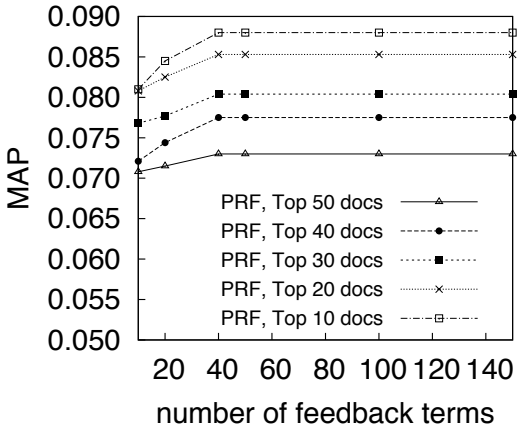


(b) recall@1000

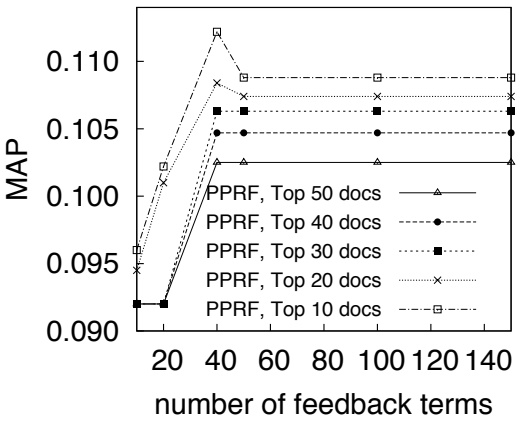


(c) PRES@1000

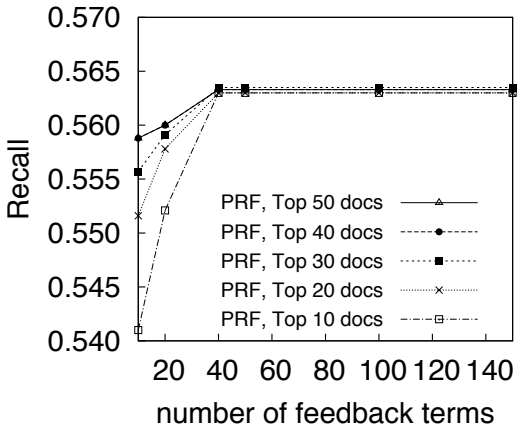
Figure 8.2. Sensitivity to the number of expansion terms and number of feedback documents on CLEF-IP 2010.



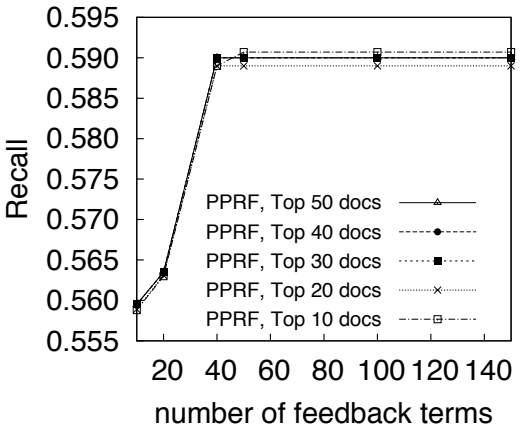
(a) MAP@1000 for PRF method



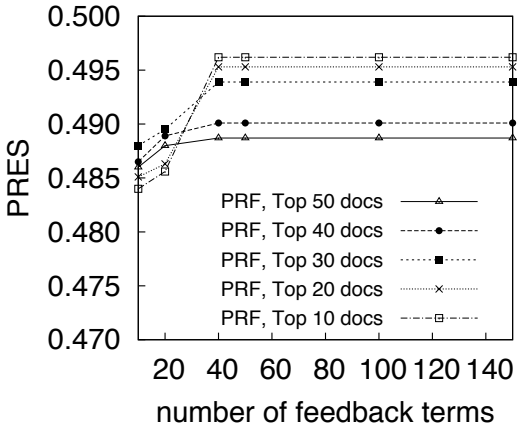
(b) MAP@1000 for PPRF method



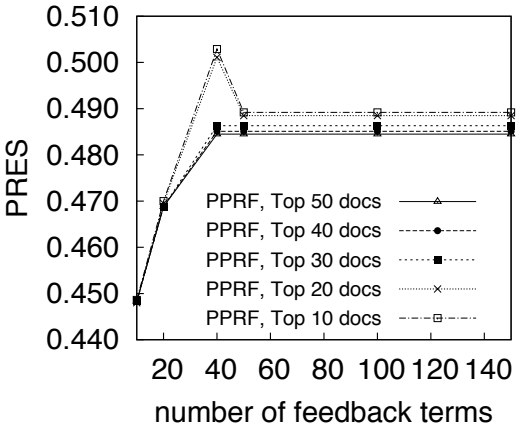
(c) Recall@1000 for PRF method



(d) Recall@1000 for PPRF method



(e) PRES@1000 for PRF method



(f) PRES@1000 for PPRF method

Figure 8.3. Sensitivity analysis of PRF and PPRF on CLEF-IP 2010.

8.4 Conclusions

In this short chapter we presented a unified framework for query expansion which incorporates bibliographic information, IPC classifications, and temporal features to improve the initial query estimated from the query patent. These relevance evidences have been introduced in the previous chapters and were not presented again here. In this chapter, we tried to provide a way to combine these evidences together.

We used the citation query model estimated from the link-based structure of the citation graph together with the term distribution of cited documents. We employed the publication dates associated with the patents to adapt our query model to the change of vocabulary over time. We performed query expansion method leveraging the IPC lexicon which has been previously discussed in Chapter 6. We observed that the query expansion method using IPC lexicon has a recall enhancing effect. We evaluated our proposed method using two patent datasets, namely CLEF-IP 2010 and CLEF-IP 2011. IEC query formulation method achieved similar performance as the state of the art methods on CLEF-IP 2010 and was able to improve over the official best results of CLEF-IP 2011.

9

Conclusions and Future Work

"I was born not knowing and have had only a little time to change that here and there."
– Richard P. Feynman

The central question investigated in this thesis is "How can we improve the information access for patent retrieval?". We proposed different methods and approaches to improve the representation of a query using different dependencies associated with a patent document. We applied our methods to prior art search task, however, they are applicable to other tasks in patent information retrieval such as invalidity search.

This chapter is organized as follows: we conclude our study in Section 9.1 by summarizing the answers to the research questions raised in chapter 1. Ultimately, in Section 9.2, we present potential directions for future work continuing the research theme discussed in this thesis.

9.1 Summary and Contributions

We now provide a summary of different chapters, focusing on addressing the research questions presented in chapter 1.

9.1.1 Reducing the Query Patent Application

RQ1 examines reducing the query patent application and selecting representative terms for formulating a query. In chapter 4, we studied three different methods for estimating a uni-gram query model from a full patent application (extracted from the test set of CLEF-IP collection): i) weighted log-likelihood

based approach, ii) cluster based query model, iii) parsimonious query modeling approach. We evaluated our models experimentally and compared their performance together. The results showed that the query model estimated from the "description" field provides the best query and we noticed that the weighted log-likelihood based approach achieves better performance among the three proposed query estimation approaches.

In chapter 5, we explored expanding the uni-gram query model estimated from a query patent application using noun phrases to enhance the query representation (RQ1.II). We asked ourselves whether noun phrases, extracted form the query document, can be beneficial for query expansion. We limited ourselves to noun phrases, instead of using other N-gram types (e.g verb phrases), as we were interested in finding technical terms which provides description about the topic of the invention.

We extracted noun phrases using the global analysis of the patent collection and calculated two different importance score for noun phrases: i) mutual information based score, ii) TF/IDF based method. We expanded the queries in the topic set using the top-K noun phrases. After evaluating the expanded rank list, we noticed that expansion using noun phrases is not beneficial for all queries. Therefore, we were interested to distinguish between queries and decide when to expand a query in order to obtain a likely improvement in the retrieval performance. We studied different features characterizing each query using query quality predictors such as Clarity measure [26], taking into account both the properties of the query and the properties of the initial result (a rank list) for a query. We introduced other measures using the specific characteristics of patent documents (such as IPC classes) inspired by Clarity measure.

Next, we tried to find a relation between query features and the performance of the final rank list both before and after expansion. We defined this problem as a regression problem and used the Average Precision value as a function of query features indicating the effectiveness of the rank list. We solved this regression problem using a learning algorithm, the Stochastic Gradient Boosting Tree (SGBT). After distinguishing between queries and identifying for which query category noun phrase expansion is effective, we performed a selective query expansion in which we performed a query dependent decision for expansion using noun phrases (RQ1.III).

The results of the experiments showed that the mutual information based scoring method extracted better phrases compared to the TF/IDF based scoring. We found that the selective query expansion approach using noun phrases led to significant improvement over the expanded and unexpanded rank lists. Comparison with the strong baselines of CLEF-IP showed that our selective query

expansion method lead to significant improvement over state of the art methods.

9.1.2 Enhancing Query Representation using Classification Tags

In chapter 6, we were interested to use the International Patent Classification (IPC classification) in order to enhance the representation of the query (RQ2). We investigated the definition of IPC classes – which provides a more established vocabulary usage compared to the query – and built a lexicon from IPC definition pages. This lexicon serves as a source for extracting expansion terms related to the IPC classes of the query document. We explored different strategies for selecting the initial query model built from the query document (as explained in Chapter 4) with expansion concepts. The strategies were: i) Explicit Expansion Concepts (EEC), ii) Implicit Expansion Concepts (IEC), iii) Combining Search Strategies (CSS), iv) Proximity-based Pseudo Relevance Feedback (PPRF).

Assuming that words appearing within the neighborhood of a given query term are more likely to be associated with that query term, we used term proximity information between the query terms and expansion terms to calculate reliable importance weights for the expansion terms. We used density kernel functions to model the relatedness of query and expansion terms. To do this, we used three different density kernel functions, namely Gaussian, Laplace, and Rectangle. The results of comparing different kernel functions showed that Gaussian kernel outperforms other types of kernels in most cases for propagating the query relatedness.

We analyzed the quality of the extracted expansion terms from the IPC conceptual lexicon in terms of their impact on the final performance of the expanded rank list and evaluated the performance of the proposed query expansion strategies. We observed that the expanded query using the IPC lexicon (in both EEC and IEC approaches) achieved statistical significant improvement compared to the initial rank list (before query expansion). These results are encouraging as it shows that the vocabulary of IPC lexicon is complementary to the terminology used in the query patent.

We also evaluated different strategies for query expansion and found that the query formulation using IEC is more effective than the other three methods. This observation confirmed that an unlimited usage of the conceptual lexicon is superior to a limited usage.

9.1.3 Analyzing Patent Citation Network

We hypothesized that the vocabulary of citations could be helpful for minimizing the vocabulary mismatch, as the terminology of cited documents contain the wording conventions used among different inventors related to the subject of the invention of a query. In chapter 7, we studied selecting query expansion terms from the citation information (RQ3).

We first built a directed unweighted graph including documents that refer and are referenced by a patent application (a topic-specific graph). We asked ourselves whether the content of the cited documents can provide additional information to the citation links which have been previously used in the state of the art of prior art search. To study this, we first employed a citation based analysis measure (Page Rank score) to calculate the impact of each document in the citation graph, in terms of the number of citations it has received and the quality (influence) of those citations. We then use the term distribution of cited documents to estimate a query model from the cited documents by identifying key terms and their corresponding weights. We perform a query expansion using the estimated query model from the cited documents to improve the representation of the initial query.

In order to address the bias of the Page Rank score associated with the age of a document – how long it stayed in the graph – we explored two different decay functions for discounting the Page Rank score associated with each document in the citation graph: i) Exponential Decay function, ii) Weibull function.

By employing temporal features associated with the citation information, the query model is adapted to the change of the vocabulary over time and query terms are selected from different time intervals.

9.1.4 Synthesizing Different Relevance Evidences

In chapter 8, we proposed a framework for combining different relevance evidences together – by merging the results of Chapters 6 and 7 – in order to take advantage of the additional information linked with the query patent for minimizing the term mismatch problem. Our proposed framework not only takes advantage of the established language of the IPC definitions to estimate a better representation of the query, but also selects terms from the highly cited documents (the community of influential inventors in the domain).

The results of the experiments showed the advantage of using citation information and temporal features in estimating a query model, leading to a better MAP. We also observed that the query expansion method using IPC lexicon led

to an improvement in terms of recall. This chapter answered RQ4.

9.2 Future Directions

Some possible future work for this line of research are listed below:

Summarizing the results of a topic. Presenting the list of all relevant documents about a topic to a user is not very useful. The reason is that the user has to go through all the documents to find the relevant parts inside each one about the topic. Therefore, it is more useful to choose a set of representative sentences related to the topic from the ranked list and present them as a condensed summary to the user. Such summary could provide a short but informative description of the influential companies and inventors in the field, in addition to the summary of the content of the returned patents. Therefore, it allows the user to get a global picture of other entities (information sources such as companies and inventors) related to a specific topic as well as the relevant textual context.

Passage retrieval. The goal of this task is to find relevant passages inside the relevant patent documents, where the information need is also represented by a passage itself. Retrieving passages from relevant patent documents instead of entire documents is a better simulation of the real task of patent examiners, as patent examiners usually have to provide relevant information at a finer granularity than that of a document. This task has been added to CLEF-IP 2012, replacing the patent document retrieval task.

The objective here is to rank passages in response to a query. The first challenge is related to the information need of the task. The information need is represented by a passage (e.g. a claim of a patent document) and it is composed of only a few sentences. To analyze the information need, we can perform semantic processing at a sentence level to identify the concepts inside the sentence. In order to understand concepts we need to define them and their boundaries and to perform this let us resort to the power of parsers [57, 85]. For the sake of simplicity let us consider that the output of the parser gives us the concepts which are expressed as noun phrases. The next goal is to find relations between concepts in a sentence

and to perform a reasoning over the meaning of the sentence. The parser can mark the relations inside the sentence which are expressed as verbs (the head of the sentence).

We can go one step further and try to categorize paragraphs in a document according to their intent (objective). We can differentiate paragraphs according to the message they are conveying. The following types can be considered as labels for passages: introduction, comparing two methods, describing the advantages or disadvantages of a method, what does a method facilitate, explaining how a method work, the novelty of a method, the usage context of a method, and conclusion. We need annotations for such types of passages and we can use a classification approach to perform such classification. It might be needed to customize the categorization types for different technological domains, thus it might be easier to focus on a specific IPC class in the beginning for which annotations can be obtained easier. A crowd-sourcing toolkit such as Amazon Mechanical Turk (MTurk) [2] can be used for obtaining annotations (at sentence or paragraph level) in the absence of experts.

We can use both the semantic similarity (based on the concepts and relations in a sentence) and intent similarity, in order to find similar paragraphs.

Ranking competitor companies. Another possibility for the future work is to figure out influential companies on a given technology field like "Tablet Computer" using an analysis over the patent citation network. To do this, both textual content and citation network information (relationships between different patents) can be used to obtain accurate ranking results for companies such as Apple, Microsoft, Google, Sony, and Samsung. Studying the evolutionary pattern of competition in a domain is another subject of interest for the future work. For example we can study how the pattern of competition among major components in a field changes over time or how new companies turn into potential competitors in a given domain (companies to watch).

Discovering terminology evolution over time. One interesting work would be to find the terminology used to refer to the same concept in different time intervals, and track down how newly coined terms are gradually replacing other terms. For example, over the years different terms were used for referring to data storage and management techniques, such as data bases,

data warehousing, data mining, Big Data, and Cloud Computing. It is interesting to find the terms related to the same concept and link them to each other over different time periods.

Academic search. The estimated query models in this thesis can be applied to scientific paper search with the goal of finding related work. Academic search shares some common characteristics with patent prior art search. The immediate similarity is related to the query-by-document nature of patent prior art search which can be extended to find the literature related to a scientific paper.

Finding related works for a scientific paper can be done in the two following steps: 1) find the domain experts. 2) generate a summary of the key contributions of the domain experts.

The first challenge is related to building a citation network from the citation information. Each node in the network represents an author and each edge denotes either a citation link or a collaboration between the two authors. This citation network can be analyzed to find domain experts. The social media can be another resource which can help us calculate an expertise score. The professional presence of an expert in the social media (e.g. Twitter and LinkedIn) is an additional factor which helps evaluate their skills in terms of adaptation of novel technologies for disseminating information.

To perform the second stage, we start by identifying the key concepts in the abstract of the scientific paper (the query). Such key concepts are then used to find similar documents. The final rank list presented to the user, includes the list of domain experts where for each expert, a summary of his/her key contributions (extracted form their published work) is generated.

Bibliography

[1] Wouter Alink, Roberto Cornacchia, and Arjen P. de Vries. Building strategies, a year later. *Workshop of the Cross-Language Evaluation Forum, LABs and Workshops, Notebook Papers*, 2010.

[2] Omar Alonso and Matthew Lease. Crowdsourcing for information retrieval: principles, methods, and applications. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 1299–1300, 2011.

[3] Giambattista Amati, Giuseppe Amodeo, and Carlo Gaibisso. Survival analysis for freshness in microblogging search. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2483–2486, 2012.

[4] Giambattista Amati, Claudio Carpineto, and Giovanni Romano. Query difficulty, robustness and selective application of query expansion. In *Proceedings of European Conference on Information Retrieval (ECIR)*, pages 127–137, 2004.

[5] Avi Arampatzis and Jaap Kamps. A signal-to-noise approach to score normalization. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 797–806, 2009.

[6] Kristine H. Atkinson. Toward a more rational patent search paradigm. In *Proceedings of the 1st ACM workshop on Patent Information Retrieval*, pages 37–40, 2008.

[7] Leif Azzopardi and Vishwa Vinay. Retrievability: an evaluation measure for higher order information access tasks. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 561–570, 2008.

[8] Shariq Bashir and Andreas Rauber. Improving retrievability of patents with cluster-based pseudo-relevance feedback documents selection. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1863–1866, 2009.

[9] Shariq Bashir and Andreas Rauber. Improving Retrievability of Patents in Prior-Art Search. In *Proceedings of European Conference on Information Retrieval (ECIR)*, pages 457–470, 2010.

[10] N. J. Belkin. Anomalous states of knowledge as a basis for information retrieval. *The Canadian Journal of Information Science*, 3:133–143, 1980.

[11] Michael Bendersky, Donald Metzler, and W. Bruce Croft. Parameterized concept weighting in verbose queries. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, 2011.

[12] Michael Bendersky, Donald Metzler, and W. Bruce Croft. Effective query formulation with multiple information sources. In *Proceedings of the International ACM Conference on Web Search and Web Data Mining (WSDM)*, pages 443–452, 2012.

[13] J. Bhogal, Andy MacFarlane, and P. Smith. A review of ontology based query expansion. *Information Processing Management*, 43(4):866–886, 2007.

[14] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30(1-7):107–117, 1998.

[15] Chris Buckley and Ellen M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 33–40, 2000.

[16] Silvia Calegari, Emanuele Panzeri, and Gabriella Pasi. Patentlight: a patent search application. In *Proceedings of the second symposium on Information interaction in context (IIiX)*, 2012.

[17] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 243–250, 2008.

[18] D. Carmel and E. Yom-Tov. *Estimating the Query Difficulty for Information Retrieval*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan and Claypool Publishers, 2010.

[19] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg. What makes a query difficult? In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 390–397, 2006.

[20] Suleyman Cetintas and Luo Si. Effective query generation and postprocessing strategies for prior art patent search. *Journal of the American Society for Information Science and Technology (JASIST)*, 63(3):512–527, 2012.

[21] Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of Association for Computational Linguistics (ACL)*, pages 310–318, 1996.

[22] Erica Cosijn and Peter Ingwersen. Dimensions of relevance. *Information Processing and Management*, 36(4):533–550, 2000.

[23] W. Bruce Croft, Michael Bendersky, Hang Li, and Gu Xu:. Query representation and understanding workshop. In *SIGIR Forum*, volume 44, pages 48–53, 2010.

[24] W. Bruce Croft, Donald Metzler, and Trevor Strohman:. *Search Engines - Information Retrieval in Practice*. Pearson Education, 2009.

[25] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. A framework for selective query expansion. Technical Report IR-338, Center for Intelligent Information Retrieval, University of Massachusetts, 2004.

[26] Stephen Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Predicting query performance. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 299–306, 2002.

[27] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. A framework for selective query expansion. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 236 – 237, 2004.

[28] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. Probabilistic query expansion using query logs. In *Proceedings of International Conference on the World-Wide Web (WWW)*, pages 325–332, 2002.

[29] Van Dang and W. Bruce Croft. Query reformulation using anchor text. In *Proceedings of the International ACM Conference on Web Search and Web Data Mining (WSDM)*, pages 41–50, 2010.

[30] Owen de Kretser and Alistair Moffat. Effective document presentation with a locality-based similarity heuristic. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 113–120, 1999.

[31] Eva D'hondt, Suzan Verberne, Wouter Alink, and Roberto Cornacchia. Combining document representations for prior-art retrieval. In *Proceedings of CLEF (Notebook Papers/LABs/Workshops)*, 2011.

[32] Fernando Diaz and Rosie Jones. Using temporal profiles of queries for precision prediction. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 18–24, 2004.

[33] Fernando Diaz and Donald Metzler. Improving the estimation of relevance models using large external corpora. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 154–161, 2006.

[34] J. H. Friedman. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 38:367–378, 1999.

[35] Atsushi Fujii. Enhancing patent retrieval by citation analysis. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 793–794, 2007.

[36] Atsushi Fujii, Makoto Iwayama, and Noriko Kando. Overview of patent retrieval task at NTCIR-4. In *Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization*, 2004.

[37] Sumio Fujita. Revisiting the document length hypotheses- NTCIR-4 CLIR and patent experiments at Patolis. In *Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization*, 2004.

[38] Debasis Ganguly, Johannes Leveling, Walid Magdy, and Gareth J.F. Jones. Patent query reduction based on pseudo-relevant documents. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1953–1956, 2011.

[39] Shima Gerani, Mark James Carman, and Fabio Crestani. Aggregation methods for proximity-based opinion retrieval. *ACM Transactions on Information Systems (TOIS)*, 30(4):26, 2012.

[40] Michael E. Lesk: Commun. Gerard Salton. The smart automatic document retrieval systems - an illustration. *Communications of the ACM*, 8(6):391–398, 1965.

[41] Julien Gobeill, Emilie Pasche, Douglas Teodoro, and Patrick Ruch. Simple pre and post processing strategies for patent searching in clef intellectual property track 2009. In *Proceedings of CLEF (Notebook Papers/LABs/-Workshops)*, pages 444–451, 2009.

[42] Erik Graf, Ingo Frommholz, Mounia Lalmas, and Keith van Rijsbergen. Knowledge modeling in prior art search. In *Advances in Multidisciplinary Retrieval, First Information Retrieval Facility Conference*, pages 31–46, 2010.

[43] Donna Harman. Towards interactive query expansion. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 321–331, 1988.

[44] Christopher G. Harris, Robert Arens, and Padmini Srinivasan. Using classification code hierarchies for patent prior ar searches. In *Current Challenges in Patent Retrieval*, pages 287–304, 2011.

[45] C. Hauff, D. Hiemstra, and F. de Jong. A survey of pre-retrieval query performance predictors. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1419–1420, 2008.

[46] C. Hauff, V. Murdock, and R. Baeza-Yates. Improved query difficulty prediction for the web. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 439–448, 2008.

[47] Claudia Hauff, Leif Azzopardi, Djoerd Hiemstra, and Franciska de Jong. Query performance prediction: Evaluation contrasted with effectiveness. In *Proceedings of European Conference on Information Retrieval (ECIR)*, pages 204–216, 2010.

[48] Ben He and Iadh Ounis. Inferring query performance using pre-retrieval predictors. In *Proceedings of String Processing and Information Retrieval (SPIRE)*, pages 43–54, 2004.

[49] Ben He and Iadh Ounis. Combining fields for query expansion and adaptive query expansion. *Information Processing Management*, 43(5):1294–1307, 2007.

[50] Ben He and Iadh Ounis. Finding good feedback documents. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2011–2014, 2009.

[51] Djoerd Hiemstra, Stephen E. Robertson, and Hugo Zaragoza. Parsimonious language models for information retrieval. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 178–185, 2004.

[52] Makoto Iwayama, Atsushi Fujii, Noriko Kando, and Akihiko Takano. Overview of the third NTCIR workshop. In *Proceedings of the ACL-2003 Workshop on Patent corpus processing*, pages 24–32, 2003.

[53] F. Jelinek, B. Merialdo, S. Roukos, and M. Strauss I. Self-organized language modeling for speech recognition. *Readings in Speech Recognition*, 1990.

[54] Hideo Joho, Leif A. Azzopardi, and Wim Vanderbauwhede. A survey of patent users: an analysis of tasks, behavior, search functionality and system requirements. In *Proceedings of the third symposium on Information interaction in context (IIiX)*, pages 13–24, 2010.

[55] Karen Sparck Jones, Steve Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: development and comparative experiments. *Information Processing and Management*, 36(6):779–808, 2000.

[56] Daniel Jurafsky and James H. Martin. *Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, 2009.

[57] Dan Klein and Christopher D. Manning. 2003. Fast exact inference with a factored model for natural language parsing. In *In Advances in Neural Information Processing Systems (NIPS)*, pages 3–10, 2002.

[58] Kazuya Konishi. Query terms extraction from patent document for invalidity search. In *NTCIR-5*, 2005.

[59] Cornelis H. A. Koster, Marc Seutter, and Jean Beney. Multi-classification of patent applications with winnow. *Ershov Memorial Conference, volume 2890 of Lecture Notes in Computer Science*, pages 546–555, 2003.

[60] Jong-Hak Lee. Analyses of multiple evidence combination. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 267–276, 1997.

[61] Xiaoyan Li and W. Bruce Croft. Time-based language models. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 469–475, 2003.

[62] Patrice Lopez and Laurent Romary. Patatras: Retrieval model combination and regression models for prior art search. In *CLEF (Notebook Papers/LABs/Workshops)*, pages 430–437, 2009.

[63] Patrice Lopez and Laurent Romary. Experiments with citation mining and key-term extraction for prior art search. *Workshop of the Cross-Language Evaluation Forum, LABs and Workshops, Notebook Papers*, 2010.

[64] Mihai Lupu and Allan Hanbury. Patent retrieval. *Foundations and Trends in Information Retrieval*, 2013.

[65] Mihai Lupu, Katja Mayer, John Tait, and Anthony J. Trippe. *Current Challenges in Patent Information Retrieval*. Springer, 2011.

[66] Yuanhua Lv and ChengXiang Zhai. Positional language models for information retrieval. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 299–306, 2009.

[67] Yuanhua Lv and ChengXiang Zhai. Positional relevance model for pseudo-relevance feedback. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 579–586, 2010.

[68] W. Magdy and G. J. F. Jones. PRES: A score metric for evaluating recall-oriented information retrieval applications. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 611–618, 2010.

[69] W. Magdy and G. J. F. Jones. A study on query expansion methods for patent retrieval. In *International Workshop on Patent Information Retrieval (PaIR) in CIKM*, pages 19–24, 2011.

[70] Walid Magdy and Gareth J. F. Jones. Applying the KISS principle for the CLEF-IP 2010 prior art candidate patent search task. *CLEF (Notebook Papers/LABs/Workshops)*, 2010.

[71] Walid Magdy, Johannes Leveling, and Gareth J. F. Jones. Exploring structured documents and query formulation techniques for patent retrieval. In *Proceedings of CLEF (Notebook Papers/LABs/Workshops)*, pages 410–417, 2009.

[72] Walid Magdy, Patrice Lopez, and Gareth J. F. Jones. Simple vs. sophisticated approaches for patent prior-art search. In *Proceedings of European Conference on Information Retrieval (ECIR)*, pages 725–728, 2010.

[73] Parvaz Mahdabi, Mostafa Keikha, Shima Gerani, Monica Landoni, and Fabio Crestani. Building queries for prior-art search. In *Proceedings of Information Retrieval Facility Conference (IRFC)*, pages 3–15, 2011.

[74] Hisao Mase, Tadataka Matsubayashi, Yuichi Ogawa, Makoto Iwayama, and Tadaaki Oshio. proposal of two-stage patent retrieval method considering the claim structure. *ACM Transactions on Asian Language Information Processing*, 4(2):190–206, 2005.

[75] Edgar Meij, Wouter Weerkamp, and Maarten de Rijke. A query model based on normalized log-likelihood. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1903–1906, 2009.

[76] David M. Mimno and Andrew McCallum. Expertise modeling for matching papers with reviewers. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 500–509, 2007.

[77] Josiane Mothe and Ludovic Tanguy. Linguistic features to predict query difficulty. In *In ACM SIGIR 2005 Workshop on Predicting Query Difficulty - Methods and Applications*, 2005.

[78] Sooyoung Oh, Zhen Lei, Wang-Chien Lee, Prasenjit Mitra, and John Yen. CV-PCR: a context-guided value-driven framework for patent citation recommendation. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2291–2296, 2013.

[79] World Intellectual Property Organisation. Glossary of terms concerning industrial property information and documentation, Appendix III to part

10, of the WIPO handbook on industrial property information and documentation. *WIPO Publication No. CD208.*, 2003.

[80] Maria-Hendrike Peetz and Maarten de Rijke. Cognitive temporal document priors. In *Proceedings of European Conference on Information Retrieval (ECIR)*, pages 318–330, 2013.

[81] Jie Peng, Craig Macdonald, Ben He, Vassilis Plachouras, and Iadh Ounis. Incorporating term dependency in the DFR framework. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 843–844, 2007.

[82] Florina Piroi, Mihai Lupu, Allan Hanbury, and Veronika Zenz:. Clef-ip 2011: Retrieval in the intellectual property domain. In *CLEF (Notebook Papers/Labs/Workshop)*, 2011.

[83] Florina Piroi and John Tait. CLEF-IP 2010: Retrieval experiments in the intellectual property domain. In *CLEF-2010 (Notebook Papers/LABs/Workshops)*, 2010.

[84] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 275–281, 1998.

[85] Christopher D. Manning Richard Socher, John Bauer and Andrew Y. Ng. Parsing with compositional vector grammars. In *Proceedings of Association for Computational Linguistics (ACL)*, 2013.

[86] S. E. Robertson. The probability ranking principle in IR. *Readings in Information Retrieval*, pages 281–286, 1997.

[87] Stephen E. Robertson and Karen Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976.

[88] Stephen E. Robertson, C. J. van Rijsbergen, and Martin F. Porter. Probabilistic models of indexing and searching. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 35–56, 1980.

[89] J.J. Rocchio. Performance indices for document retrieval systems. In *Report ISR-8, The Computation Laboratory of Harvard University*, 1964.

[90] Giovanna Roda, John Tait, Florina Piroi, and Veronika Zenz. CLEF-IP 2009: Retrieval experiments in the intellectual property domain. *Multilingual Information Access Evaluation I. Text Retrieval Experiments, 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009*, pages 385–409, 2009.

[91] Michail Salampasis, Georgios Paltoglou, and Anastasia Giahanou. Report on the CLEF-IP 2012 experiments: Search of topically organized patents. In *Proceedings of CLEF (Notebook Papers/LABs/Workshops)*, 2012.

[92] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM (CACM)*, 18:613–620, 1975.

[93] J. A. Shaw and E. A. Fox. Combination of multiple searches. *In TREC 1994*, 1994.

[94] Mark D. Smucker, James Allan, and Ben Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 623–632, 2007.

[95] Toru Takaki, Atsushi Fujii, and Tetsuya Ishikawa. Associative document retrieval by query subtopic analysis and its application to invalidity patent search. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 399–405, 2004.

[96] Jie Tang, Bo Wang, Yang Yang, Po Hu, Yanting Zhao, Xinyu Yan, Bo Gao, Minlie Huang, Peng Xu, Weichang Li, and Adam K. Usadi. PatentMiner: topic-driven patent analysis and mining. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1366–1374, 2012.

[97] Wenbin Tang, Jie Tang, Tao Lei, Chenhao Tan, Bo Gao, and Tian Li. On optimization of expertise matching with various constraints. *Neurocomputing*, 76(1):71–83, 2012.

[98] Douglas Teodoro, Julien Gobeill, Emilie Pasche, Dina Vishnyakova, Patrick Ruch, and Christian Lovis. Automatic Prior Art Searching and Patent Encoding at CLEF-IP '10. *Workshop of the Cross-Language Evaluation Forum, LABs and Workshops, Notebook Papers*, 2010.

[99] Douglas Teodoro, Emilie Pasche, Dina Vishnyakova, Christian Lovis, Julien Gobeill, and Patrick Ruch. Automatic ipc encoding and novelty tracking for effective patent mining. In *The 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access.*, pages 309–317, 2010.

[100] Kristina Toutanova, Dan Klein, Christopher Manning, , and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT-NAACL)*, pages 252–259, 2003.

[101] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

[102] Manisha Verma and Vasudeva Varma. Exploring keyphrase extraction and ipc classification vectors for prior art search. In *Proceedings of CLEF (Notebook Papers/LABs/Workshops)*, 2011.

[103] Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 61–69, 1994.

[104] Mattan Winaver, Oren Kurland, and Carmel Domshlak. Towards robust query expansion: Model selection in the language modeling framework. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 729–730, 2007.

[105] Sen Wu, Jimeng Sun, and Jie Tang. Patent partner recommendation in enterprise social networks. In *Proceedings of the International ACM Conference on Web Search and Web Data Mining (WSDM)*, pages 43–52, 2013.

[106] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 4–11, 1996.

[107] Xiaobing Xue and W. Bruce Croft. Automatic query generation for patent search. *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 2009.

[108] Xiaobing Xue and W. Bruce Croft. Transforming patents into prior-art queries. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 808–809, 2009.

[109] Yang Yang, Jie Tang, Jacklyne Keomany, Yanting Zhao, Juanzi Li, Ying Ding, Tian Li, and Liangwei Wang. Mining competitive relationships by learning across heterogeneous networks. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1432–1441, 2012.

[110] Yin Yang, Nilesh Bansal, Wisam Dakka, Panagiotis Ipeirotis, Nick Koudas, and Dimitris Papadias. Query by document. In *Proceedings of the International ACM Conference on Web Search and Web Data Mining (WSDM)*, pages 34–43, 2009.

[111] Elad Yom-Tov, Shai Fine, David Carmel, and Adam Darlow. Learning to estimate query difficulty. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 512–519, 2005.

[112] ChengXiang Zhai and John D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of ACM SIGIR conference on Research and Developement in Information Retrieval*, pages 334–342, 2001.