A Study of Query Reformulation Methods for Patent Prior Art Search with Partial Patent Applications

No Author Given

No Institute Given

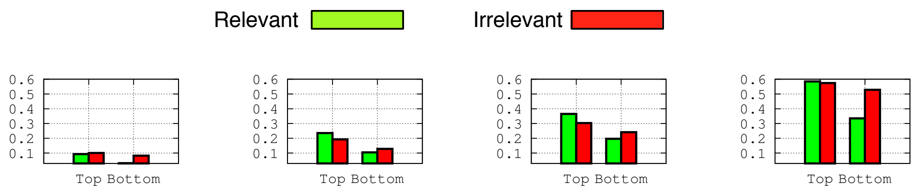
Abstract. Patents are used by entities to legally protect their inventions and represent a multi-billion dollar industry of licensing and litigation. In 2013, 302,948 patent applications were approved in the US alone – a number that has doubled in the past 15 years and which makes prior art search a daunting, but necessary task in the patent application process. In this work, we seek to investigate the efficacy of prior art search strategies from the perspective of the inventor who wishes to assess the patentability of their ideas prior to writing a full application. While much of the literature inspired by the evaluation framework of the CLEF-IP competition has aimed to assist patent examiners in assessing prior art for complete patent applications, less of this work has focused on patent search with queries representing partial applications. In the (partial) patent search setting, a query is often much longer than in other standard IR tasks, e.g., a claims section may contain hundreds or even thousands of words. While the length of such queries may suggest query reduction strategies to remove irrelevant terms, intentional obfuscation and general language used in patents suggests that it may help to expand queries with additionally relevant terms. To assess the trade-offs among all of these pre-application prior art search strategies, we comparatively evaluate a variety of partial application search and query reformulation methods. Among numerous findings, querying with a full description, perhaps in conjunction with generic (non-patent specific) query reduction methods, is recommended for best performance. However, we also find that querying with an abstract represents the best trade-off in terms of writing effort vs. retrieval efficacy (i.e., querying with the claims or description sections only lead to marginal improvements) and that for such relatively short queries, generic query expansion methods help.

Keywords: Query Reformulation, Patent Search, Experimentation.

1 Introduction

Patents are used by entities to legally protect their inventions and represent a multi-billion dollar industry of licensing and litigation. In 2013, 302,948 patent applications were approved in the US alone[1], a number that has doubled in

[1] `http://www.uspto.gov/web/offices/ac/ido/oeip/taf/us_stat.htm`



(a) Title query. (b) Abstract query. (c) Claims query. (d) Description query.

Fig. 1: Average Jaccard similarity between fields of topics and the corresponding (ir)relevant documents for different sets of top and bottom performing queries.

the past 15 years. Given that a single existing patent may invalidate a new patent application, helping inventors assess the patentability of an idea through a patent prior art search before writing a complete patent application is an important task.

Patent prior art search involves finding previously granted patents that may be relevant to a new patent application. The objective and challenges of standard formulations of patent prior art search are different from those of standard text and web search since [8]: (i) queries are (partial) patent applications, which consist of documents with hundreds or thousands of words organized into several sections, while typical queries in text and web search constitute only a few words; and (ii) patent prior art search is a recall-oriented task, where the primary focus is to retrieve all relevant documents at early ranks, in contrast to text and web search that are precision-oriented, where the primary goal is to retrieve a subset of documents that satisfy the query intent. Another important characteristic of patent prior art search is that, in contrast to scientific and technical writers, patent writers tend to generalize and maximize the scope of what is protected by a patent, which further complicates the task of formulating effective queries.

While much of the literature inspired by the evaluation framework of the CLEF-IP competition has aimed to assist patent examiners in assessing prior art for complete patent applications, less work has focused on assessing the patentability of inventions before writing a full patent application. Furthermore, prior art search with queries that represent unfinished patent applications is generally desirable, since writing a full application is time-consuming and costly, especially if lawyers are hired to assist.

To assess the difficulty of querying with partial patent applications, we refer to Figure 1. Here we show an analysis of the average Jaccard similarity[2] between different queries (representing the title, abstract, claims, or descriptions of partial patent applications) and the labeled relevant (all) and irrelevant documents (top 10 irrelevant documents ranked by BM25 [14]). We show results for the top 100 and bottom 100 queries (100 queries that perform the best, and 100 queries

[2] The Jaccard similarity is used to measure the term overlap between two sets. Before applying the Jaccard similarity, patent-specific stopwords were removed, as suggested by [10].

that perform the worst) of CLEP-IP 2010 evaluated according to Mean Average Precision (MAP). Note that while the title section is usually composed of an average of six terms, the other sections are longer, ranging from tens to thousands of terms. There are three notable trends here: (i) term overlap increases from title to description since the query size grows accordingly; (ii) the bottom 100 performing queries tend to have much smaller term overlap with the relevant documents than the top 100 queries; and (iii) even in the best case of querying with very long description sections, the average term overlap indicates many terms of relevant documents are not found in the query.

Similar observations in the general patent prior art search literature [9] have led to a research focus on query reformulation. Therefore, we suggest an investigation of *query reformulation* [1] methods as a means for improving the term overlap between queries that represent partial patent applications and relevant documents, with the objective of assessing not only the performance of standard query reformulation methods, but also the effectiveness of query reformulation methods that exploit patent-specific characteristics.

In summary, to aid the patent inventor in developing an effective pre-application prior art search strategy, we seek to answer the following questions in this work:

- What parts of a patent application should a patent inventor write first to achieve effective prior art search? What are the trade-offs in section writing effort vs. the retrieval performance of querying with that section?
- In query expansion, do any sections of patents serve as better sources of expansion terms? What expansion methods work best, and in what settings?
- For query reformulation (both query expansion and reduction), which methods work best, and in what settings? Do patent-specific reformulation methods offer advantages over more generic IR reformulation methods?

To answer these questions, we perform a thorough comparative analysis of partial patent application query strategies and reformulation methods on the CLEF-IP patent prior art search datasets.

The rest of the paper is organized as follows: in Section 2, we present a variety of generic and patent-specific query reformulation methods; in Section 3, we present the evaluation results and analysis to answer the above questions; and in Section 4, we conclude with key observations from this evaluation that lead to concrete recommendations for patent prior art search with partial applications.

2 Query Reformulation for Patents

Query Reformulation is the process of transforming an initial query Q to another query Q'. This transformation may be either an expansion or a reduction of the query. *Query Expansion* (QE) [4] enhances the query with additional terms likely to occur in relevant documents. Hence, given a query representation Q, QE aims to select an optimal subset T_k of k terms, which are relevant to Q, then build Q' such as $Q' = Q \cup T_k$. As for *Query Reduction* (QR) [6], it is the process that reduces the query such that superfluous information is removed. Hence, given a

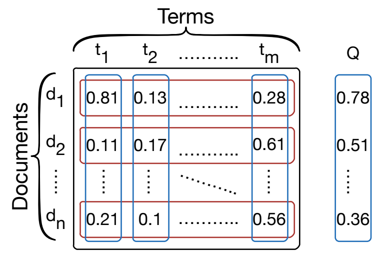


Fig. 2: Notation used in MMR QE/QR.

query representation Q, QR aims to select an optimal subset $T_k \subset Q$ of k terms, which are relevant to Q, then build Q' such as $Q' = T_k$.

In the following sections, we describe the standard and patent-specific query reformulation methods that we evaluate in Section 3.

2.1 Generic Query Reformulation Methods

The Rocchio Algorithm for Relevance Feedback: The Rocchio algorithm [16] is a classic algorithm of relevance feedback used mainly for query expansion. In brief, it provides a method of incorporating relevance feedback information into the vector space model representing a query [12]. The underlying theory behind Rocchio is to find a query vector $\overrightarrow{Q'}$, that maximizes similarity with relevant documents while minimizing similarity with irrelevant documents. Typically, a pseudo-relevance feedback (PRF) set of k top ranked documents obtained after an initial run of the query is considered as the set of relevant documents to build $\overrightarrow{Q'}$. We refer to this method as RocchioQE.[3]

Similarly, Rocchio can be used as a QR method. Basically, the idea is that once the Rocchio-modified query vector has been computed, it is possible to select only the terms that appear in the initial query Q and rank them using the Rocchio score and finally, select the top k terms with the highest score to build Q'. We refer to this approach as RocchioQR.

Maximal Marginal Relevance for Query Reformulation: As a general method for query reformulation, we also consider a method of "diverse" term selection — an adaptation of the *Maximal Marginal Relevance* (MMR) [3] algorithm for result set diversification. But, rather than use MMR for diverse document selection (as typically used), it is used here for diverse term selection — the hypothesis being that diverse term selection may improve coverage of relevant terms in the PRF set.

In the case of QE, we call this diversified expansion method MMR Query Expansion (MMRQE). MMRQE takes as input a PRF set, which is used to build

[3] We used the LucQE module, which provides an implementation of the Rocchio method for Lucene. `http://lucene-qe.sourceforge.net/`

a document-term matrix of n documents and m terms as shown in Figure 2 (the TF-IDF is used to populate the matrix for each document vector). To represent the query Q in the documents' dimension as in Figure 2, we use the BM25 or TF-IDF score between each document d_i and the query. Hence, given a query representation Q, MMRQE aims to select an optimal subset of k terms $T_k^* \subset D$ (where $|T_k^*| = k$ and $k \ll |m|$) relevant to Q but inherently different from each other (i.e., diverse). This can be achieved by building T_k^* in a greedy manner by choosing the next optimal term t_k^* given the previous set of optimal term selections $T_{k-1}^* = \{t_1^*, \ldots, t_{k-1}^*\}$ (assuming $T_0^* = \emptyset$) using the MMR diverse selection criterion:

$$t_k^* = \underset{t_k \notin T_{k-1}^*}{\arg\max}[\lambda \cos(Q, t_k) - (1 - \lambda) \max_{t_j \in T_{k-1}^*} \cos(t_j, t_k)] \tag{1}$$

Here, the first cosine similarity term measures relevance between the query Q and possible expansion term t_k while the second term penalizes the possible expansion term according to its cosine similarity with any currently selected term in T_{k-1}^*. The parameter $\lambda \in [0, 1]$ trades off relevance and diversity. For MMRQE, we found that $\lambda = 0.5$ generally provide the best results, according to our experiments on the CLEF-IP training dataset collection.

For QR, we can greedily rebuild the query from scratch, while choosing diversified terms from the query itself. Here, we call this approach MMR Query Reduction (MMRQR). Formally, given a query representation Q, MMRQR aims to select an optimal subset of k terms $T_k^* \subset Q$ (where $|T_k^*| = k$ and $k < |Q|$) relevant to Q but inherently different from each other (i.e., diverse). This can be achieved by building T_k^* in a greedy manner by choosing the next optimal term t_k^* given the previous set of optimal term selections $T_{k-1}^* = \{t_1^*, \ldots, t_{k-1}^*\}$ (assuming $T_0^* = \emptyset$) using an adaptation of the MMR diverse selection criterion. Note that we use all the sections of the patent documents in the PRF set to built the document-term matrix of n documents and m terms shown in Figure 2. For MMRQR, we found that $\lambda = 0.8$ generally provide the best results in our experiments on the CLEF-IP dataset collection.

The key insight we want to highlight is that MMRQE does not select expansion terms independently as in practical usage of Rocchio, but rather it selects terms that have uncorrelated usage patterns across documents, thus hopefully encouraging diverse term selection that covers more documents for a fixed expansion budget k and ideally, higher recall.

2.2 Patent-specific Query Reformulation Methods

Synonym Sets for Patent Query Expansion: Magdy et al. [9] proposed a patent query expansion method, which automatically generates candidate synonym sets (SynSet) for terms to use as a source of expansion terms. The idea for generating the SynSet comes from the characteristics of the CLEF-IP patent collection, where some of the sections in some patents are translated into three languages (English, French, and German). They used these parallel manual translations to create possible synonyms sets. Hence, for a word w in one language which has possible translations to a set of words in another language $w_1, w_2, \ldots, w_n$, this set of words can be considered as synonyms or at least related to each other. The generated SynSet is used for query expansion in two ways: (i) The first one used the probability associated with the SynSet entries as a weight for each expanded term in the query (denoted WSynSet). Therefore, each term was replaced with its SynSet entries with the probability of each item in the SynSet acting as a weight to the term within the query. (ii) The second one neglected this associated probability and used uniform weighting for all synonyms of a given term (denoted USynSet).

Patent Lexicon for Query Expansion: Mahdabi et al. [11] proposed to build a query-specific patent lexicon based on definitions of the International Patent Classification (IPC). The lexicon is simply built by removing general and patent-specific stop-words from the text of IPC definition pages. Each entry in the 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. Then, the lexicon is used to extract expansion concepts related to the context of the information need of a given query patent. To this end, the IPC class of the query patent is searched in the lexicon and the terms matching this class are considered as candidate expansion terms. The proposed approach tries to combine these two complementary vocabularies. In this paper we refer to this patent query expansion method as IPC Codes.

Language Model for Query Reduction: In [5], the authors proposed a query reduction technique, which decomposes a query (a patent section) into constituent text segments and computes Language Model (LM) similarities by calculating the probability of generating each segment from the top ranked documents (PRF set). Then, the query is reduced by removing the least similar segments from the query. We refer to this method as LMQR.

IPC Codes for Query Reduction: Based on the intuition that, terms in the IPC code definition may represent "stop-words", especially if they are rare (infrequent in the patent application), one can think to reduce a patent query as follows: (i) For each patent application, take the definitions of the IPC codes which are associated to it. Then, (ii) rank the terms of the query according to the difference in their frequency in the query and their frequency in the class code definition. Finally, (iii) remove bottom terms of this ranking from the query (i.e. good terms are terms that occur a lot in the query, and few in the class code definition, whereas bad terms are those that occur few in the query, and a lot in the class code definition). In the evaluation section we denote this approach IPC-StopWords.

Further Afield: While some more complex patent-specific methods have also been explored for general patent prior art search [2,18,11], space limitations preclude an exhaustive comparison to all available methods. Notwithstanding this, we believe the above outlined patent-specific query reformulation methods circumscribe a range of patent-specific approaches spanning synonym lexicons, specially derived language models, and IPC code resources; hence our evaluation supports the objective of identifying general query reformulation methods from the novel perspective of partial patent application prior art search that may be deserving of further investigation in future work.

3 Experimental Evaluation

In this section we first explain the experimental setup for evaluating the effectiveness of patent prior art search with partial applications. Then, we discuss the results of QE and QR methods in Sections 3.2 and 3.3 respectively.

3.1 Experimental Setup

For our experiments, we used used the Lucene IR System[4] to index the English subset of CLEF-IP 2010 and CLEF-IP 2011 datasets[5] [13,15] with the default settings for stemming and stop-word removal. We also removed patent-specific stop-words as described in [8]. CLEF-IP 2010 contains 2.6 million patent documents, and the English test sets of CLEF-IP 2010 correspond to 1303 topic sources of partial patent application queries. We also experimented with the CLEF-IP 2011 dataset, but observed the same overall trends as for CLEF-IP 2010 and hence omit these redundant results due to space limitations.

In our implementation, each section of a patent (title, abstract, claims, and description) is indexed in a separate field so that different sections can be used, for example, as source of expansion terms. However, when a query is processed, all indexed fields are targeted, since this generally offers best retrieval performance. As suggested in previous work [7,15] and found to yield best results in our experiments, we used the patent classification (IPC) codes assigned to the query topics to filter search results to match at least one of the query IPC codes.

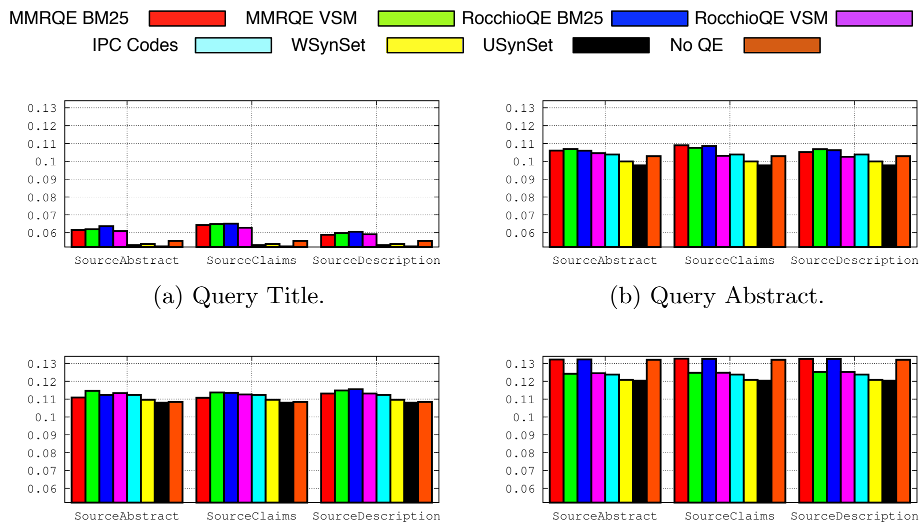
We report both MAP and PRES (Patent Retrieval Evaluation Score). The PRES metric places more emphasis on high-recall retrieval by weighting relevant documents lower in the ranking more highly than MAP. We report these ranking evaluation metrics on the top 1000 results.

3.2 Query Expansion Results

In this section, we discuss the results of partial patent queries with the QE methods described in Section 2. In doing this experimentation, there are many configuration options and associated questions to consider:

– **Partial patent query type:** We consider a query of a partial patent application to consist of either the title, the abstract, the claims, or the description section. Critical questions are: what part of a partial application an inventor

[4] http://lucene.apache.org/
[5] http://www.ifs.tuwien.ac.at/~clef-ip/



(a) Query Title.

(b) Query Abstract.

(c) Query Claims.

(d) Query Description.

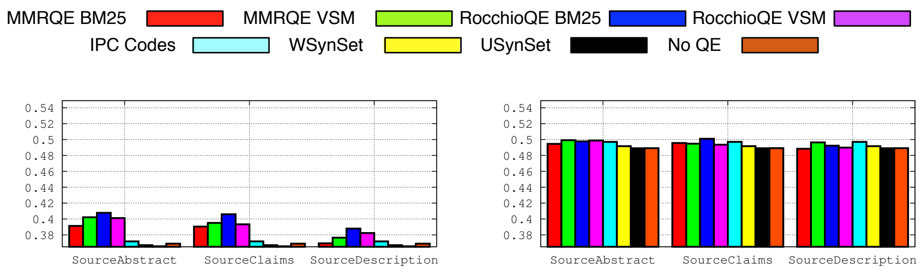
Fig. 3: MAP for QE methods on CLEF-IP 2010.

should write to obtain the best search results? And what QE methods work best for each type of query?

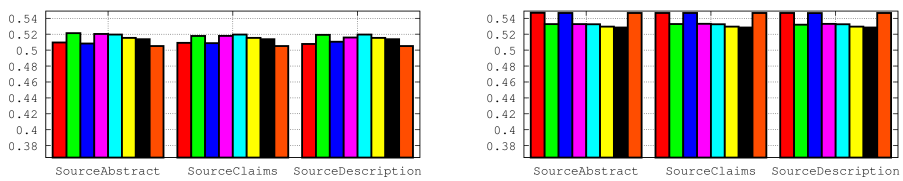
- **Query expansion source:** We consider the abstract, claims, and description sections as different term sources to determine which section offers the best source of expansion terms, e.g., are words in the claims of particularly high value as expansion terms? We omit the use of the title as a source of expansion terms noting that this configuration performed poorly due to the relative sparsity of useful expansion terms in the titles of the PRF set.
- **Relevance model:** For initial retrieval of documents in the *pseudo-relevant* feedback set (PRF) and subsequent re-retrieval, there are various options for the relevance ranking model. In this work, we explore a probabilistic approach represented by the popular BM25 [14] algorithm, as well as a vector space model (VSM) approach, TF-IDF [17]. A natural question is which relevance model works best for query expansion for patent prior art search?
- **Term selection method:** We consider the different query expansion methods described above, i.e. RocchioQE, MMRQE, IPC Codes, WSynSet, USynSet and ask what is the best QE method for patent search?

To summarize all the results obtained over all the above configurations, Figure 3 and Figure 4 show the MAP and PRES obtained for all the QE methods, while selecting the optimal number of terms used for the expansion (the number of terms that maximizes the performance for each method). From these results, we make the following observations:

1. The best partial application section to use for querying is the description section. We attribute this to the fact that the description section has more



(a) Query Title. (b) Query Abstract.

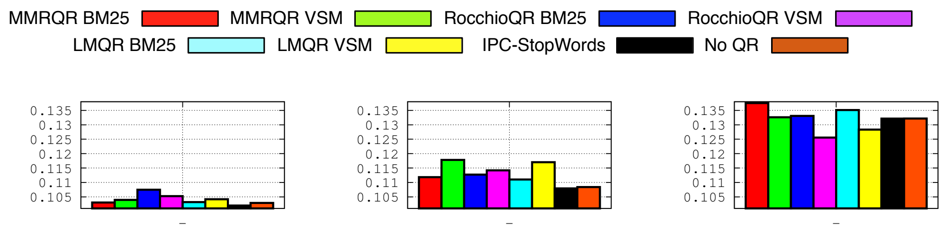


(c) Query Claims. (d) Query Description.

Fig. 4: PRES for QE methods on CLEF-IP 2010.

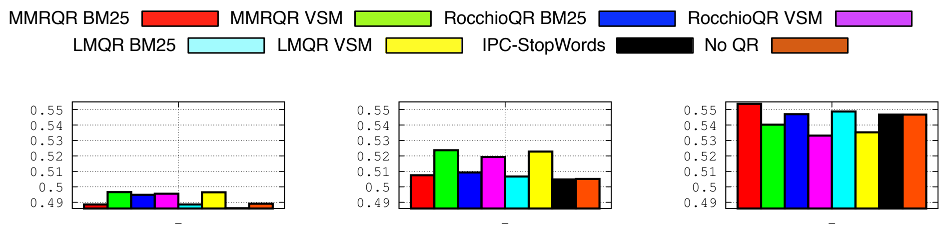
content along with relevant terms that define the invention since a detailed summary of the invention is described therein.

2. However, perhaps a better trade-off in terms of effort vs. retrieval performance is to query with the abstract. Relative to the description, it takes much less effort to write the abstract. Further, querying with the abstract provides a substantial boost in retrieval performance compared to the title (about 165% for MAP). In contrast, querying with the claims and description offer only marginal performance gains (about 10% to 30% for MAP) compared to using the abstract.
3. Query expansion is not useful for very long queries (i.e. description) since no method outperforms the baseline. This indicates that in advanced writing stages of the patent preparation process, QE is not useful.
4. When dealing with short queries such as the title or abstract, MMRQE is less effective than Rocchio, whereas it appears to provide slightly better comparative results for the longer claims query. This suggests diverse term selection may be helpful for long queries.
5. The description section does not appear to be a good source for expansion, likely since its content is too broad and it contains many irrelevant terms.
6. When dealing with short and medium-length queries (i.e., title, abstract, and claims), VSM performs better than BM25, while for very long queries (i.e., description), BM25 performs.
7. In general, generic QE methods like Rocchio tend to outperform patent-specific QE methods, although among patent-specific methods, the IPC Codes approach seemed to work best.



(a) Query Abstract. (b) Query Claims. (c) Query Description.

Fig. 5: MAP for QR methods on CLEF-IP 2010.



(a) Query Abstract. (b) Query Claims. (c) Query Description.

Fig. 6: PRES for QR methods on CLEF 2010.

3.3 Query Reduction Results

Next we discuss the results of the evaluation performed on the QR methods described in Section 2. As for QE, we carry out comprehensive experiments with the following configuration options and associated questions to consider:

- **Partial patent query type:** We apply QR methods to a query of a partial patent application, consisting of the abstract, the claims or the description sections. A critical question is what part of a partial application is best suited for QR? Note that we consider that there is no interest in reducing a title query since it already contains very few terms.
- **Relevance model:** We explore a probabilistic approach represented by the popular BM25 [14] algorithm, as well as a vector space model (VSM) approach, TF-IDF [17]. A natural question is which relevance model works best for query reduction for patent prior art search?
- **Term selection method:** We consider the different query reduction methods described above, i.e. RocchioQR, MMRQR, LMQR, IPC-StopWords and ask what is the best QR method for patent search? Further, how do these results compare to QE for the same queries?

To summarize all the results obtained over all the above configurations, Figures 5, and 6 show the respective MAP and PRES performance obtained for all QR methods, when selecting the optimal number of terms removed from the original queries. From these results, we make the following observations:

1. The best performing QR methods show benefits vs. No QR for all queries (i.e., abstract, claims, and description).
2. The term selection methods that provide the best performance are, in general, RocchioQR and MMRQR.
3. When dealing with medium-length queries (i.e., abstract and claims), VSM performs better than BM25, while for very long queries (i.e., description), BM25-based QR methods perform better than VSM-based QR methods.
4. In comparison to the MAP and PRES results for QE from Figure 3 and Figure 4, the best QE and QR methods perform comparably for abstract and claims queries, whereas for description queries, the best QR method slightly outperforms the best QE method and No QR. Hence, the best overall retrieval result in this work in terms of both MAP and PRES comes from a description query with a generic (non-patent specific) QR method.

4 Conclusions

In this paper, we analyzed various query strategies for patent prior art search with partial (incomplete) applications along with generic and patent-specific query reformulation (expansion and reduction) methods. We performed a comprehensive comparative evaluation of these methods on the CLEF-IP patent corpus for prior art search.

We showed that the description is the best partial application section to query with, followed by the claims, the abstract, and lastly the title section. However, the largest boost in performance (about 165% for MAP) comes when switching from a title query to an abstract query; smaller relative boosts are given by querying instead with the claims or description (about 10% to 30% for MAP). This is a critical insight since it is substantially easier for the patent inventor to draft an abstract rather than a full patent description and in doing so, still manage to retrieve the majority of prior art that would have been retrieved with the full description.

We observed that query expansion (QE) methods are useful for short to medium length queries (i.e., title, abstract, and claims), but useless for very long queries (i.e., the description section). We also showed that the description section does not provide the best source of expansion terms for QE, rather the claims or the abstract tend to offer better candidate terms for QE. In the same vein, we also found traditional IR methods like Rocchio or variations to work just as well for QE (and generally better) in comparison to patent-specific methods using specialized expansion sources such as synonym lexicons or IPC code definitions.

Regarding query reduction (QR) methods, we showed these techniques are generally most effective compared to QE for the description section (the longest section used as a partial application query). Albeit by a slim margin over No QR, the overall best retrieval performance results in this work are achieved with generic (non-patent specific) QR methods for description queries.

In conclusion, we return to our initial objective to aid the patent inventor in identifying an effective pre-application prior art search strategy. Our evaluation reveals the critical insight that while querying with a full description, perhaps combined with generic query reduction methods, yields strong overall retrieval performance, querying with just an abstract and using generic QE methods may yield the best trade-off in terms of writing effort vs. retrieval performance.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Publishing Company, USA, 2nd edition, 2010.
2. S. Bashir and A. Rauber. Improving retrievability of patents in prior-art search. In *ECIR*, 2010.
3. J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.
4. E. N. Efthimiadis. Query expansion. *Annual Review of Inf. Systems and Technology (ARIST)*, 31:121–187, 1996.
5. D. Ganguly, J. Leveling, W. Magdy, and G. J. Jones. Patent query reduction using pseudo relevance feedback. In *CIKM*, 2011.
6. G. Kumaran and V. R. Carvalho. Reducing long queries using query quality predictors. In *SIGIR*, SIGIR '09, pages 564–571, New York, NY, USA, 2009. ACM.
7. P. Lopez and L. Romary. Patatras: retrieval model combination and regression models for prior art search. CLEF'09, pages 430–437, Berlin, Heidelberg, 2009. Springer-Verlag.
8. W. Magdy. *Toward Higher Effectiveness for Recall-Oriented Information Retrieval: A Patent Retrieval Case Study*. PhD thesis, Dublin City University School of Computing, 2012.
9. W. Magdy and G. J. Jones. A study on query expansion methods for patent retrieval. In *PaIR*, 2011.
10. P. Mahdabi, L. Andersson, M. Keikha, and F. Crestani. Automatic refinement of patent queries using concept importance predictors. In *SIGIR*, pages 505–514, New York, NY, USA, 2012. ACM.
11. P. Mahdabi, S. Gerani, J. X. Huang, and F. Crestani. Leveraging conceptual lexicon: query disambiguation using proximity information for patent retrieval. In *SIGIR*, 2013.
12. C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
13. F. Piroi, M. Lupu, A. Hanbury, and V. Zenz. Clef-ip 2011: Retrieval in the intellectual property domain. In *CLEF (Notebook Papers/Labs/Workshop)*, 2011.
14. S. E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-2. In *TREC*, pages 21–34, 1993.
15. G. Roda, J. Tait, F. Piroi, and V. Zenz. Clef-ip 2009: Retrieval experiments in the intellectual property domain. In C. Peters, G. Nunzio, M. Kurimo, T. Mandl, D. Mostefa, A. Penas, and G. Roda, editors, *Multilingual Information Access Evaluation I. Text Retrieval Experiments*, volume 6241 of *Lecture Notes in Computer Science*, pages 385–409. Springer, 2009.
16. G. Salton. *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1971.
17. G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Comm. ACM*, 18(11):613–620, nov. 1975.
18. M. Verma and V. Varma. Patent search using ipc classification vectors. In *PaIR*, 2011.