Evaluation 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 2012, 276,788 patent applications were approved in the US alone – a number that has doubled in the past 15 years. 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 to help inventors to assess the patentability of their ideas prior to writing a full application. In this paper, we carry out an intensive study and evaluation of both patent specific and standard query reformulation methods for patent prior art search with partial patent applications.

Keywords: Query Reformulation, Patent Search, Experimentation.

1 Introduction

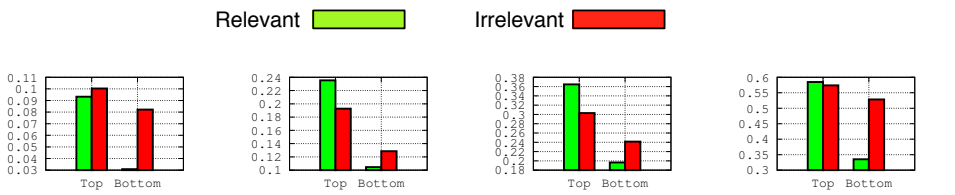
Patents are used by entities to legally protect their inventions and represent a multi-billion dollar industry of licensing and litigation. In 2012, 276,788 patent applications were approved in the US alone a number that has doubled in the past 15 years. Hence, helping both inventors and patent examiners assess the patentability of a given patent application through a patent prior art search is a critical 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 [7]: (i) queries are (partial) patent applications, which consist of documents with hundreds of words organized into several sections, while queries in text and web search constitute only a few words; (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 relevant documents. Another important characteristic about patent 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, and try to make sure that finding any relevant prior work by the patent examiner is a hard job.

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. Prior art search with queries that represent unfinished patent applications is certainly 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[1] between different queries (representing the title, abstract, claims, or descriptions intended to represent a partial patent application) 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 that perform the worst) of CLEP-IP 2010 evaluated according to Mean Average Precision (MAP). Note that, while the title section is usually composed by an average of six terms, the other sections are longer, ranging from ten to thousands of terms.



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

Fig. 1: Average Jaccard similarity of (ir)relevant documents with the result sets for different queries.

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) the best overlap for any relevant document set for any set of queries is less than one in four terms. 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

[1] The Jaccard similarity is used to measure the term overlap between two sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. Before applying the Jaccard similarity, patent-specific stopwords were removed, as suggested by [10].

standard query reformulation methods, but also the effectiveness of query reformulation methods that exploit patent-specific characteristics. In this paper, we try to mainly answer the following questions: *What are these patent query reformulation methods and how do they work? What is the best section in a patent application to use as a query? What is the best patent query reformulation method? To which extent are they efficient compared to standard query reformulation approaches?*

The main contributions of this work can be summarized as follows:

1. We propose a deep study of the state of the art patent query reformulation methods.
2. A thorough comparative analysis of these methods against standard query reformulation methods on standardized datasets of CLEF-IP.

The rest of the paper is organized as follows: in Section 2 we present a set of patent specific query reformulation methods; in Section 3 we present our evaluation framework and results analysis; and in Section 4 we conclude with possible directions for future work.

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 a reduction or an expansion of the query. *Query Reduction* (QR) [5] reduces the query such that superfluous information is removed, while *Query Expansion* (QE) [3] enhance the query with additional terms likely to occur in relevant documents.

2.1 Patent Query Expansion Methods

In this paper, we consider four patent specific query expansion methods, that we believe are the most representative.

Maximal Marginal Relevance for Patent Query Expansion In this case, we want to define a method of "diverse" term selection — such as the *Maximal Marginal Relevance* (MMR) [2] algorithm for result set diversification. The idea is to use it for diverse term selection. In the case of query expansion, we call this method MMRQE.

MMRQE takes as input a pseudo-relevant feedback set of n documents (PRF), which is obtained after a retrieval for the initial query. From the PRF set, we build a document-term matrix of n documents and m terms as shown in Figure 2, which uses a TF-IDF weighting for each document vector (row d_i for $1 \leq i \leq n$). However, as we will see shortly, the view that will be important for us in this work is instead the term vector (column t_j for $1 \leq j \leq m$). To represent the query Q column vector in Figure 2 having a numerical entry for every document d_i, we found that computing the BM25 or TF-IDF score between each

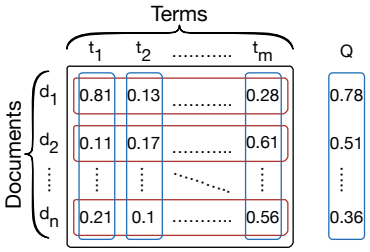


Fig. 2: Notation used in MMR QE/QR.

document d_i and the query provided the best performance (in our experiments, the score used is given by the indicated relevance model).

Given a query representation Q, we aim 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.

Synonyms Sets for Patent Query Expansion Magdy et al. [9] proposed a patent query expansion method, which automatically generates candidate synonyms sets (SynSet) for terms, and use it 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). The idea is to use 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 use 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**). This strategy is similar to adding synonyms from WordNet where no probability is assigned.

Patent Lexicon for Query Reformulation 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 build by removing general and patent stop-words from the text of IPC definition pages. 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. Then, the lexicon build 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 approach proposed tries to combine these two complementary vocabularies. In this paper we refer to this patent query expansion method as **IPC Codes**.

2.2 Patent Query Reduction Methods

Here, consider the following three patent query reduction methods.

Maximal Marginal Relevance for Patent Query Reduction Following the same motivations than those of explore diversification for term selection, we can imagine greedily rebuild the query from scratch, while choosing diversified terms (i.e. terms of the query). Here, we call this approach MMR Query Reduction (MMRQR). Formally, given a query representation Q, we aim 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 the we used all the sections of the patent documents of the PRF set to built the document-term matrix of n documents and m terms shown in Figure 2.

Language Model for Query Reduction In [4], the authors proposed a query reduction technique, which decomposes a query (a patent section) into constituent text segments and computes a Language Modeling (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 by **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 both their frequency in the class code definition, and their frequency in the query. 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 Codes**.

3 Experimental Evaluation

In this section, we discuss the results of the evaluation performed on the query reformulation methods described above.

3.1 Experimental Setup

For our experiments we used used the Lucene IR System[2] to index the English subset of CLEF-IP 2010 dataset[3] [13,15] with the default stemming and stop-word removal. We removed patent specific stop-words as described in [7]. CLEF-IP 2010 contains 2.6 million patent documents, and the English test sets of CLEF-IP 2010 correspond to 1303 topics. We also made the same experiments on the CLEF-IP 2011 dataset, but for lack of space we omit the results from the paper. However the obtained results was almost similar and presented the same trends.

In our implementation, each section of a patent (title, abstract, claims, and description) is indexed in a separate field. Hence, when a query is processed, all fields in the index are targeted, since it is sensible to use all available content. We also used the patent classification (IPC) for filtering the results by constraining them to have common classifications with the patent topic as suggested in previous works [6,15]. Finally, we report MAP, and PRES [8], which combines Recall with the quality of ranking and weights relevant documents lower in the ranking more highly than MAP. We report the evaluation metrics on the top 1000 results.

3.2 Baselines

We also want to compare the performance of the above patent specific query reformulation methods described in Section 2 to general query reformulation methods. The selected baselines are described below.

Rocchio for Query Expansion The Rocchio algorithm [16] is a classic algorithm of relevance feedback used mainly for query expansion. Basically, it provides a way of incorporating relevance feedback information into the vector space model representing a query [12]. We refer to this method as **Rocchio**[4].

[2] `http://lucene.apache.org/`

[3] `http://www.ifs.tuwien.ac.at/~clef-ip/`

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

Rocchio for Query Reduction As a general QR method, we proposed to adapt the Rocchio method for query pruning. Basically, the idea is once we have computed the Rocchio modified query vector, we take only terms of the initial query that appear in this vector and rank them using the Rocchio score. Then, we remove n terms with the lower score. We refer to this approach as **RocchioQR**.

3.3 Query Expansion Results

In this section, we discuss the results of the evaluation performed on the QE methods described above. But before, we first discuss the effect of the size of the PRF set on the performance. Table 1 shows the impact of the PRF size on the performance for the two QE algorithms Rocchio and MMRQE. These results are shown on the CLEF-IP 2010 training dataset, which consists of 196 topics. We observe that the best QE performance results are obtained when using few documents in the PRF set as it was also reported in [9] (in our case, the top five gave the best results). This is certainly due to the fact that a large PRF set will include too much irrelevant documents, whose the terms may negatively affect the quality of the expanded query.

Table 1: Effect of PRF with various numbers of feedback documents on the CLEF-IP 2010 dataset. 20 terms are used for query expansion.

Query/Source	**Metric**	**Method**	**5**	**10**	**20**
Query: Abstract	MAP	Rocchio	0.074	0.072	0.070
	BL=0.073	MMRQE	0.074	0.071	0.071
Source: Claims	PRES	Rocchio	0.409	0.409	0.409
	BL=0.403	MMRQE	0.411	0.411	0.410
Query: Claims	MAP	Rocchio	0.083	0.080	0.079
	BL=0.081	MMRQE	0.082	0.080	0.080
Source: Claims	PRES	Rocchio	0.443	0.445	0.446
	BL=0.433	MMRQE	0.445	0.444	0.442

Next, we carry out comprehensive experiments with the following specific options:

- **Query type:** {Title, Abstract, Claims, Description}
- **Query expansion source:** {Abstract, Claims, Description} (considering that there is no interest in using the title as source for the expansion)
- **Relevance model:** {BM25, Vector Space Model (VSM)}
- **Term selection method:** {Rocchio, MMRQE, IPC Codes, WSynSet, USynSet}

Figure 3 shows the results obtained in terms of MAP and PRES for CLEF-IP 2010 for different numbers of expanded terms k on the x-axis (with $k = 0$ using no QE, just the baseline retrieval model). For lack of space we show only the

results of queries extracted from the claims and the abstract used as source of query expansion. From these results, we make the following observations: (i) for the two retrieval models (VSM and BM25), MMRQE provides the best performance for both MAP and PRES (except for MAP, where Rocchio BM25 provides better performance than MMRQE BM25), (ii) for both MMRQE and Rocchio, the best performance is obtained while adding no more than 50 terms to the original queries (adding more terms may have no effect, or decrease the performance), and (iii) exploiting external sources for query expansion provides poor performance (IPC code definition and SynSets).

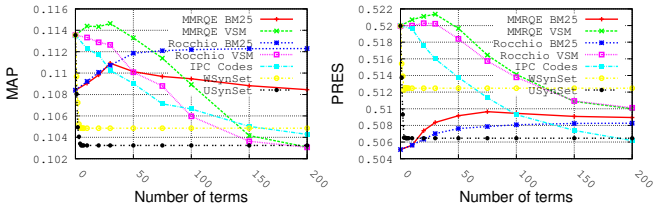


Fig. 3: Results obtained while using the claims for querying and the abstract as source of query expansion on the CLEF-IP 2010 dataset.

To summarize all the results obtained over all the above configurations, Figures 5, and 6, show the performance obtained for all the QE methods, while selecting the optimal number of terms used for the expansion (number of terms that maximizes the performance for each method). From these results, we first observe that the best section to use for querying is the description section (see Figures 5b, and 6d). We attribute this to the fact that the description section has more content along with relevant terms that define the invention since a detailed summary of the invention is described therein.

According to our experiments, the best source for query expansion is the claims section. We attribute this to the fact that, the claims contain not only relevant, but also, specific terminology, since the scope of the invention is described therein. However, when querying using the claims, other sources of query expansion provided better performance. This may be because claims are very similar between them and contained specific terms; consequently, the queries lack of diversity and general terms or synonyms that are used to describe similar inventions. It is interesting to notice that the description is not either a good source for expansion, since its content is too broad, therefore, it contains many irrelevant terms that hurt the performance.

As expected, we observed that query expansion is not useful for very long queries (i.e. description), indicating that in advanced stages of the patent application process, QE is not relevant. We also notice that when dealing with

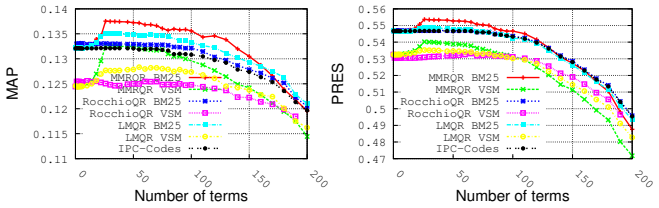


Fig. 4: Results obtained for QR while using the description for querying on the CLEF-IP 2010 dataset.

more mid-long queries such as abstract or claims, MMRQE is more effective than Rocchio, which suggest that diverse term selection is not crucial for short queries. We also note that using the IPC code definitions (as suggested by [11]) and SynSet (method of [9]) as a source of expansion, gave poor performance (see IPC Codes and SynSet bars along the Figures). Finally, regarding the best term selection method, we conclude that in general MMRQE provides better performance than Rocchio.

3.4 Query Reduction Results

In this section, we discuss the results of the evaluation performed on the QR methods described above. As recommended in [4] and confirmed in our own experimentation (not shown due to lack of space), best QR performance results are also obtained when using few documents in the PRF set (in our case, the top five gave the best results).

In the following, we carry out experiments with the following specific options:

- **Query type:** {Title, Abstract, Claims, Description} (considering that there is no interest in reducing a title query)
- **Relevance model:** {BM25, Vector Space Model (VSM)}
- **Term selection method:** {RocchioQR, MMRQR, LMQR, IPC Codes}

Figure 4 shows the results obtained in terms of MAP and PRES for CLEF-IP 2010 for different numbers of removed terms k on the x-axis (with $k = 0$ using no QR, just the baseline retrieval model). For lack of space we show only the results of queries extracted from the description. These results tell us mainly two things: (i) for the two retrieval models, MMRQR provides the best performance for both MAP and PRES, and (ii) for almost all methods, the best performance is obtained when removing about 30 terms from the original queries (in the case where the description is used for querying). Removing more terms will decrease significantly the performance.

To summarize all the results obtained over all the above configurations, Figures 7, and 8 show the performance obtained for all the QR methods, when

selecting the optimal number of terms removed from the original queries (number of terms removed that maximizes the performance for each method).

From these results, we make the following observations: (i) query reduction is very often not useful for short queries (i.e. title), since no QR method outperforms significantly the baseline (i.e. No QR), (ii) when dealing with very long query (i.e. description), BM25 based QR methods perform better than VSM based QR methods, and (iii) in general, MMRQR provides better performance than the other methods.

4 Conclusion

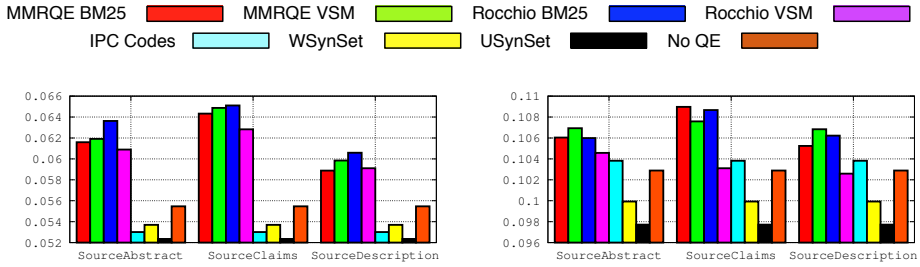
In this paper we analyzed general and specific query reformulation methods for patent prior art search for partial (incomplete) patent applications on two patent retrieval corpora, namely CLEF-IP 2010 and CLEF-IP 2011. We demonstrated that QE methods are critical for short queries, i.e. title, abstract, and claims, but useless for very long queries, i.e. the description section. We also showed that claims is the best section that works with QE both to query with and to use as a source of query expansion terms, suggesting that claims should be written at early stages of the patent application drafting so that they can be use to performed patent prior art search. In the same vein, we also found that the patent specific fields are more suited as a source for expansion than external sources such as synonym dictionaries. Here, future work concerns how can we exploit patent specific meta-data such as inventor and citation networks for query expansion

Regarding QR methods, we showed that these techniques are effective to some extent for claims and description sections, which are considered the longest sections in a patent application. Future work may consist of exploiting query quality predictors to identify useless terms in a query using machine learning methods.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Publishing Company, USA, 2nd edition, 2010.
2. J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.
3. E. N. Efthimiadis. Query expansion. *Annual Review of Inf. Systems and Technology (ARIST)*, 31:121–187, 1996.
4. D. Ganguly, J. Leveling, W. Magdy, and G. J. Jones. Patent query reduction using pseudo relevance feedback. In *CIKM*, 2011.
5. 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.
6. 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.

7. W. Magdy. *Toward Higher Effectiveness for Recall-Oriented Information Retrieval: A Patent Retrieval Case Study.* PhD thesis, Dublin City University School of Computing, 2012.
8. W. Magdy and G. J. Jones. PRES: A Score Metric for Evaluating Recall-oriented Information Retrieval Applications. In *SIGIR*, pages 611–618, New York, NY, USA, 2010. ACM.
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.



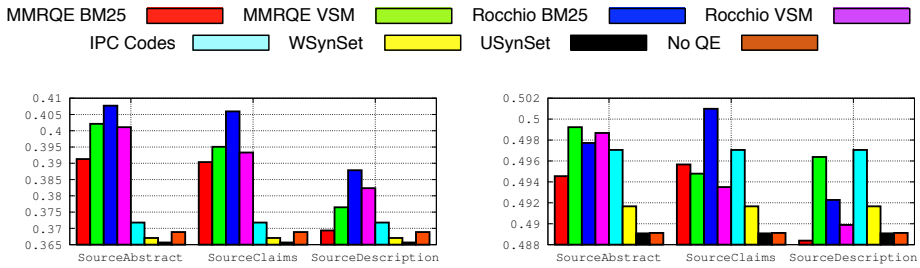
(a) Query Title.

(b) Query Abstract.

(c) Query Claims.

(d) Query Description.

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



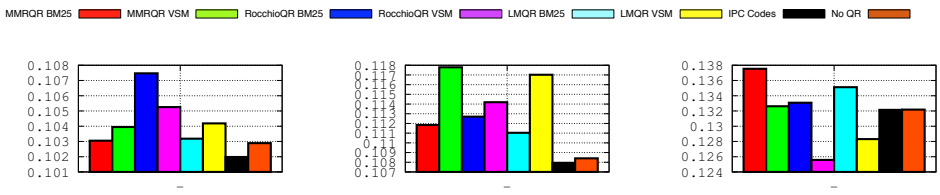
(a) Query Title.

(b) Query Abstract.

(c) Query Claims.

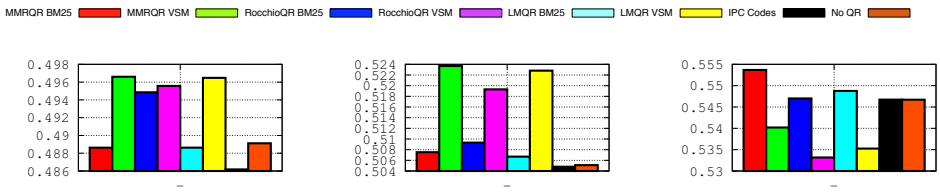
(d) Query Description.

Fig. 6: PRES for QE methods on CLEF-IP 2010.



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

Fig. 7: MAP for QR methods on CLEF-IP 2010.



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

Fig. 8: PRES for QR methods on CLEF 2010.