University of Santiago de Compostela at CLEF-IP09

José Carlos Toucedo, David E. Losada
Grupo de Sistemas Inteligentes
Dept. Electrónica y Computación
Universidad de Santiago de Compostela, Spain
{josecarlos.toucedo,david.losada}@usc.es

Abstract

In this paper we describe our participation in CLEF-IP 2009 (prior-art search task). This was the first year of the task and we focused on how to build effectively a prior art query from a patent. Basically, we implemented simple strategies to extract terms from some textual fields of the patent documents and gave preference to title terms. We ran experiments with standard BM25 configurations and we paid little attention to language-dependent issues.

Categories and Subject Descriptors

H.3.3 [**Information storage and retrieval**]: Information Search and Retrieval—*query formulation*; H.3.7 [**Information storage and retrieval**]: Digital Libraries

General Terms

Performance, Experimentation, Measurement

Keywords

Patent Retrieval, Prior Art Search, Cross Language, Query Formulation

1 Introduction

The main task of the CLEF-IP09 track is to investigate Information Retrieval (IR) techniques for patent retrieval, specifically for *prior art search*. Prior art consists of any kind of information (publication, product, process, etc.) within the patent and non-patent literature that has been made available to the public, maybe orally, before the filing date of a patent application. Therefore, a prior art search tries to retrieve any prior record with identical or similar contents to a given patent application.

This track provides the participants with a huge collection of more than one million patents from the European Patent Office (EPO). This is composed of all the patents published between 1986 and 2000. Every patent, identified by an unique number, consists of several XML documents generated at different stages of the patent's life-cycle. Therefore, each patent document is identified by the patent number plus the stage (represented as a kind code plus a version number). For instance, the patent document with the id 0981201-A3 denotes the third version of the application of the patent 0981201. The information within a patent document is structured. Some fields appear in all the stages, such as the title, the bibliographic data, the description and the claims. However, there are some fields like the abstract that only occur in a particular stage. At the EPO,

the patents can be written in English, French or German. The title and the claims of every patent are translated into these three languages. The rest of the patent is written in a single language.

In an information retrieval setting the patent to be evaluated can be regarded as the information need and all the granted patents to date as the document collection. Since a patent is made up of several documents all these documents have to be taken into account in order to produce a query patent. In the prior art search task, the query patent provided is built from the non-common fields of the patent documents and from the common fields of the document with the highest stage. A query patent constructed in this way in the EPO is about 3500 terms long on average. The query patent is therefore a long and verbose document and many of its terms are redundant or unrepresentative.

Throghout these notes we will explain the approach we have taken to address the prior art search task. This year our main objective has been to formulate a concise query that effectively represents the underlying information need.

Since this is our first participation in CLEF, we have just focused on query formulation. We recognize that there are many issues such as link analysis, entity extraction, cross-language retrieval, field boosting, etc. that might play a key role in prior art search but these will be considered for next editions of this track.

The rest of the paper is organized as follows. Section 2 describes the approach we have taken, specifically how the query is built and what experiments we designed; the runs we submitted are explained in section 3 and the conclusions we extracted are exposed in section 4.

2 Approach taken

Although every patent is composed of several documents, this track requires that retrieval is performed at patent level. This problem can be addressed following two different approaches: a) building an index of patents or b) building an index of patent documents. The former requires to define an effective strategy to combine several patent documents into a single patent representation. The second approach is simpler because it only requires to post-process the retrieved documents in order to obtain one result per patent. Our choice was to assign a patent the score of its highest ranked document. This follows the intuition that the patent document that is the most similar to the patent query reflects well the connection between the query and the underlying patent.

Although the documents contain terms from three different languages, no language-oriented distinction was made during the index construction [1]. The index contains all terms in any language for each patent document. Note that some fields are translated into the three languages in the patent (e.g. the title) and these translations appear in the index associated to the same document. Furthermore, stemming was not applied and an English stopword list (with 733 stopwords) was used in order to remove common words. This makes sense because almost 70% of the data was written in English.

2.1 Query formulation

The query patent is too long to be processed in a reasonable time and contains noisy terms that might harm performance. We think that a good query preprocessing is a key factor in order to achieve good effectiveness.

Our experiments focused on extracting the most significant terms from the query patent, i.e. those terms that are discriminative. To this aim, we used *inverse document frequency* (idf). In our evaluation with the training set, we concentrated on deciding the number of terms that should be included into the query. We ran this process in both a language-independent and language-dependent way (i.e. a single ranking of terms vs. three rankings of terms, one for each language).

The number of query terms is difficult to set because few query terms make that the query processing is fast but the information need might be misrepresented; on the other hand, if many

[1] We deeply thank the support of Erik Graf and Leif Azzopardi, from University of Glasgow, who granted us access to their indexes.

terms are taken the query will contain many noisy terms and, furthermore, the query processing time might be prohibitive. We have studied two methods to choose a suitable number of terms: (i) establishing a fixed number of terms for all queries and (ii) establishing a fixed percentage of the query patent length (i.e. the query size varies from one query patent to another).

Once the number of query terms has been selected, we must determine how they are extracted. We explored two strategies: language-independent and language-dependent. Suppose that we select n terms from the original query patent regardless of the language. This means that all query patent terms (english, french and german terms) are ranked together and we simply select the n terms with the highest *idf* from this list. Because of the nature of the languages, it is likely that the three languages present different *idf* patterns. Besides, there are fewer German/French documents than English documents and, therefore, this introduces a bias in terms of *idf*. We therefore felt that we needed to test other alternatives for selecting terms. We tried out an extraction of terms where each language contributes with the same number of terms. In this second strategy we first grouped the terms of a query patent depending on their language (no classification was needed since every field in the XML is tagged with language information). Next, the highest $n' = \lfloor n/3 \rfloor$ terms from each group are extracted. The query is finally obtained by compiling the terms from the three groups.

So far, we simply explained which options we have considered for query formulation. In section 2.3 we will explain how some combinations of these strategies perform.

2.2 Retrieval model

We used the well-known BM25 retrieval model [2] with the usual parameters ($b = 0.75$, $k_1 = 1.2$, $k_3 = 1000$), but we also tried several variations for b and k_1 in the submitted runs.

The platform under which we executed our experiments was the Lemur Toolkit [1].

All experiments were executed in the LDC, a system provided by the Information Retrieval Facility (IRF) [2].

2.3 Training

With the training data provided by the track, we studied two dimensions: query length and language. Query length refers to the way in which query size is set. As argued above, this can be done in a query-dependent (i.e. a given percentage of the patent query terms are selected) or query-independent way (i.e. a fixed number of terms are selected for all queries). The language dimension reflects the way in which terms are ranked (language-independent, i.e. a single rank for all terms; language-dependent, one rank of terms for every language). Hence, our training consisted of studying how the four combinations of these dimensions perform in terms of two well-known measures, MAP and Bpref.

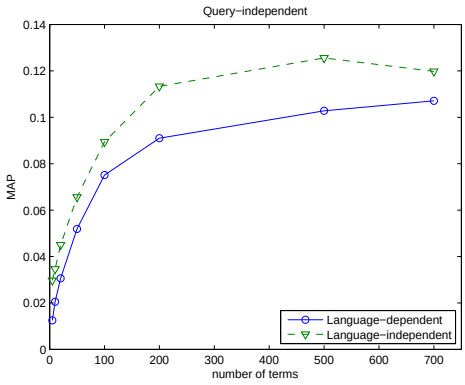
The following results were obtained with the large training set (500 queries) of the main task, which contains queries in the three languages. In this case, we only tried the usual parameters for the BM25 retrieval model.

Figures 1(a) and 2(a) consider the case where the number of terms is fixed for all queries. We clearly get better performance when the language is not taken into account during the training. However, figures 1(b) and 2(b), where terms are selected using a percentage of the query length, show a different trend. Figure 1(b) shows that, for values less than 50% [3], no significant difference can be stablished in terms of MAP. In contrast, figure 2(b) shows that the language-dependent choice is slightly more consistent than the language-independent one in terms of Bpref.

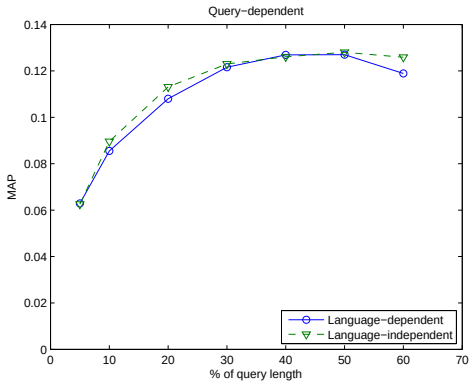
We have to choose between two opposite models, the model that consists of combining the query-dependent and language-dependent strategies and, on the other hand, the model that considers the query-independent and language-independent strategies together. If we observe carefully the plots we will note that these two models do not differ in MAP values but, in terms of Bpref, the model that is language and query dependent presents the best performance.

[2]We are grateful to the IRF for the support given to us.

[3]We are not taking into account greater values because the resulting queries are too long.

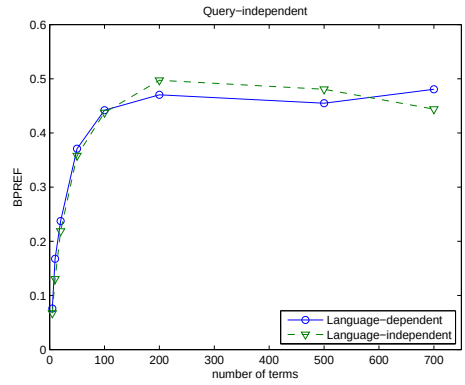


(a) Query-independent experiments

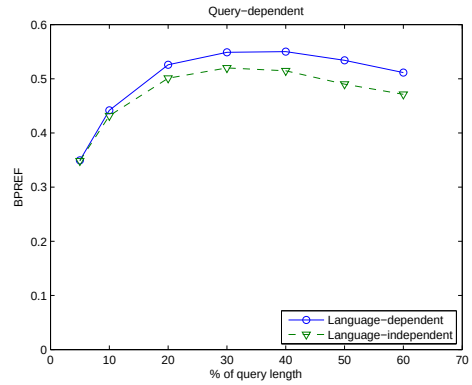


(b) Query-dependent experiments

Figure 1: MAP performance



(a) Query-independent experiments



(b) Query-dependent experiments

Figure 2: BPREF performance

Based on this training, we also concluded that a 40% of query length is a good trade-off between performance and efficiency.

3 Submitted runs

We participated in the *Main* task of this track with eight runs for the *Small* set of topics, which contains 500 queries in different languages.

First, we submitted four runs considering the scenario that best worked for our training experiments. These four runs differ on the retrieval model parameters:

- *uscom_BM25a*: $b = 0.2$, $k_1 = 0.1$, $k_3 = 1000$
- *uscom_BM25b*: $b = 0.75$, $k_1 = 1.2$, $k_3 = 1000$
- *uscom_BM25c*: $b = 0.75$, $k_1 = 1.6$, $k_3 = 1000$
- *uscom_BM25d*: $b = 1$, $k_1 = 1.2$, $k_3 = 1000$

Furthermore, we submitted four additional runs where the final queries were expanded with the title terms of the query patent. In this way, the query term frequency of these terms is augmented

and the presence of the title terms in the final queries is guaranteed. These new runs are labeled as the previous ones plus an extra "t".

4 Results and conclusions

The official evaluation results of our submitted runs are summarized in Table 1.

	P	P5	P10	P100	R	R5	R10	R100	MAP	nDCG
uscom_BM25a	.0029	.0948	.0644	.0141	.4247	.0900	.1183	.2473	.0837	.4466
uscom_BM25b	.0041	.1184	.0858	.0205	.5553	.1082	.1569	.3509	.1079	.4410
uscom_BM25c	.0042	.1180	.0858	.0206	.5563	.1104	.1564	.3504	.1071	.4341
uscom_BM25d	.0042	.1188	.0852	.0206	.5630	.1113	.1558	.3500	.1071	.4346
uscom_BM25at	.0031	.1004	.0680	.0151	.4549	.0937	.1223	.2637	.0867	.4331
uscom_BM25bt	.0042	**.1280**	**.0908**	**.0213**	.5729	**.1176**	**.1631**	**.3610**	**.1133**	**.4588**
uscom_BM25ct	.0042	.1268	.0898	.0212	.5722	.1172	.1611	.3602	.1132	.4544
uscom_BM25dt	**.0043**	.1252	.0892	**.0213**	**.5773**	.1163	.1606	.3609	.1121	.4455

Table 1: Submitted runs for CLEF-IP 09

The first conclusion we can extract from the evaluation is that our decision to force the presence of title terms worked well. Regardless of the configuration of the BM25 parameters, the run with the title terms always obtains better performance than its counterpart. Therefore, we can state that the title terms represent an important factor in prior art search.

Furthermore, among the configurations with the title terms the best run is the one labeled as *uscom_BM25bt*. This run corresponds to the usual parameters of the BM25 retrieval model, i.e. $b = 0.75$, $k_1 = 1.2$, $k_3 = 1000$.

5 Acknowledgements

We are deeply grateful to Erik Graff and Leif Azzopardi, from University of Glasgow, for their help during our experiments.

We also thank the support of the IRF.

This research was co-funded by FEDER and *Xunta de Galicia* under projects 07SIN005206PR and 2008/068.

References

[1] The Lemur Toolkit. `http://www.lemurproject.org`.

[2] S. Robertson, S. Walker, S. Jones, M.M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. pages 109–126, 1996.