Using BM25F and KLD for Patent Retrieval

Joaquín Pérez-Iglesias , Álvaro Rodrigo, Víctor Fresno

NLP & IR Group, UNED, Madrid
{joaquin.perez,alvarory,vfresno}@lsi.uned.es

Abstract. We describe in this paper our system for the Prior-art task of CLEF-IP 2010 (a task focused on the retrieval of relevant patents to a given one) and its results. We have developed a system where patents are indexed by fields in order to allow a selection of the most discriminative terms of each field, applying Kullback-Leibler divergence as feature selection method, and using different boost factors for each field applying BM25F as ranking function. Although CLEF-IP has been proposed in a multilingual scenario, we have approached it from a monolingual perspective. The results are on the average of last year's results, what encourages us to continue the development of this system by including some kind of multi-lingual processing.

1 Introduction

The automatic retrieval of patents supposes an important application in the process of publishing new patents. Before publishing a patent, it must be granted the novelty of that invention. However, the manual search of similar patents has associated a high cost. Furthermore, the patents can have been published in any country and in any language; therefore, this is a multilingual task, where the patents to be retrieved may be written in a language different to the one of the candidate patent. These are some of the reasons why there is a growing interest in developing automatic systems able to provide patents relevant to a given one.

The CLEF-IP task at the Cross Language Evaluation Forum aims at evaluating patent retrieval systems in a multilingual scenario. CLEF-IP proposes two different tasks: the Prior art Task and the Classification Task. We have participated at the Prior art Task, where a set of patents (we call them topic patents) are given to participants, and each participant has to return, for each patent, a ranking of the patents that are considered relevant.

This search is performed over a collection with 2.6 million of patent documents and the set of topic patents contains 2000 patents (it is possible to send runs over a subset that consists of 500 patents). The documents contained in the collection represents different versions of a patent, where there can be additional information or some changes among the different versions of the same patent.

In short, our approach to the Prior art Task is based on an Information Retrieval (IR) system based on BM25F as ranking function, and the application of Kullback-Leibler divergence for selecting the most discriminative terms of each field for a given topic patent. Then, we build a query with these most discriminative terms and we assign different boost factors to the terms of different fields for improving the ranking of candidate relevant patents.

The structure of this paper is as follows: the main components of our system are described in Section 2. The description of the submitted runs is given in Section 3, while the results of these runs is shown in Section 4. Finally, some conclusions and future work are given in Section 5.

2 System Overview

This section describes the different modules of our system. The architecture of the system is given in Figures 1 and 2.

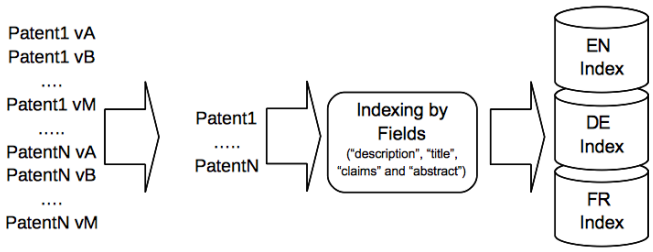

Fig. 1. Architecture of indexing phase.

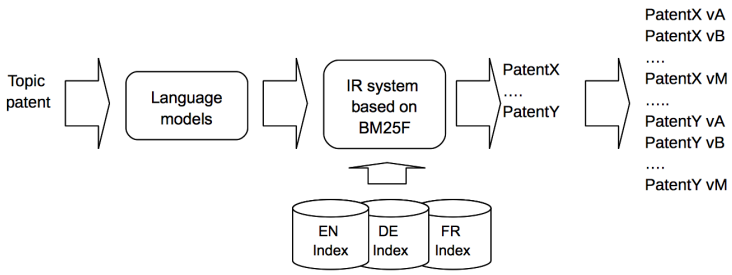

Fig. 2. Architecture of retrieval phase.

The system has three major parts: the first one for the preprocessing of patent documents, where an index per language is built; a second one for building queries; and a third one for retrieving relevant patents given a query patent. Besides, we added the possibility of using predictors at the output of the IR engine. The components of our system are described in detail in the following subsections.

2.1 Indexing

Indexing the whole Patent The different versions of a patent represent the different stages of it before its final version. Each document has a kind code for indicating the

stage of the document. The different versions of a patent does not always contain the whole data of a patent. This is why it is opened to participants the option of joining the different versions of a patent into a single one for processing purposes.

Since the objective of the task is to retrieve relevant patents (no matter the version), we considered more appropriated to work with patents given as results of combining all its different versions. The main reason of this decision was the fact that the different stages of a patent represent different subsets of the whole patent and, then, the combination of them would allow to dispose of the terminology of the whole patent for selecting the most discriminative terms of it.

Indexing by Fields Once we have a patent, only the terms of the fields considered relevant for us are extracted. We selected as relevant fields of a patent the "description", "title", "claims" and "abstract" based on checking the observations already made in CLEF-IP 2009 [1]. Afterwards, the terms of each field are indexed in a different field of the index, resulting in having four fields per index.

Our decision of considering four different fields in the index arose from the fact that each patent has different sections and then, the terms of a certain section can be more representative for that section than for other sections. Otherwise, high discriminative terms of a section could not be so relevant when they are used for searching over whole patents. Furthermore, by using fields, we can assign different boost factors to the terms of each fields. Thus, if we think that a field is more relevant than another one, it is easier for us to give more importance to that field.

Indexing by Language The collection of patents used in CLEF-IP contains documents in English, German and French. We have two options at the indexing period regarding how to deal with languages: to create a unique index which contains all the patents no matter their language; or to create an index per language.

It must be taken into account that we used in this edition a monolingual approach. Then, if we used a unique index, the terms of a language might affect the frequency of terms in another language. For example, a high relevance term in a language could be a stopword in a different language, and then, it would not appear in any document of the second language. Therefore, the statistics of term frequency that are used for retrieving documents could be not appropriate if we merge documents in different languages. This is why we decided to create three different indexes, one per language. Each index contains only patents in this language. In the case of fields in a language different to the one of the document (for example, the title can be in different languages), the field is ignored.

This fact has as a main drawback that in case of a patent should be retrieved with a topic expressed in a different language, this patent document would not be retrieved. Finally, the preprocessing step for feeding the indexes included stemming and stop words removal specifically by language.

2.2 Building the Query

Once we have all the patents in indexes, we have to select which query is going to be used for retrieving the most relevant patents to a given one. The first possibility would be

to use the whole topic patent as a query. However, this is different to classical IR, where the size of the query is quite smaller. A big size of the query has associated a higher delay for obtaining the output from the IR engine. This observation lead us to think that the best approach is to select for each topic patent, the most discriminative terms for building the query. Given that we considered important to separate the different fields of a patent in the index, we obtained the most relevant terms by field.

We made use of language models for selecting the terms of a query. In fact, we built two different language models given a topic document: one language model of the whole indexed collection, and another one for the given topic document. Each of these language models is built taking into account each one of the considered fields.

We computed the Kullback-Leibler divergence (KLD) [2] between the language model of the topic and the one of the collection (at the level of each field). A term is considered highly discriminant if it appears frequently on the topic query q, and, at the same time, its frequency in the collection is not significant. KLD measures the divergence between two probability distributions: the probability of a term within this query and within the whole collection C. It can be expressed as follows:

$$KLD_{p_Q,p_C} = p_Q(t) \cdot log(\frac{p_Q(t)}{p_C(t)}) \quad (1)$$

where p_Q is the probability of each term t within the patent topic q, that is the frequency of the term t within the document q, divided by the length (number of terms) of q. Finally, p_C, is the probability of the same term t within the whole collection, and it is equivalent to the frequency of term t in the collection divided by the total number of terms in C.

Applying the previous equation we will be able to rank all the terms from the patent topic according to their importance within the query. After ranking the terms by their divergence, a threshold is established and only those terms with divergence above the threshold are selected.

Thus, by using KLD we are able to build queries that contain the most discriminative terms of each field, what we hope that will allow us to retrieve the most relevant patents to a given one.

2.3 Patent Retrieval

With the query already build, we have to fed the IR engine with that query in order to obtain a ranking of patents considered relevant to the input query. The decision of the IR model must take into account that we are working with an index with different fields. The combination of fields can be achieved by means of using a ranking function which is able to exploit this type of document structures. This is why we chose BM25F [4].

In order to use BM25F, we first obtain the accumulated weight of a term over all fields as next:

$$\textit{weight(t,d)} = \sum_{c \textit{ in } d} \frac{occurs^d_{t,c} \cdot boost_c}{((1 - b_c) + b_c \cdot \frac{l_c}{avl_c})}$$

where l_c is the field length; avl_c is the average length for the field c; b_c is a constant related to the field length, similar to b in BM25 and $boost_c$ is the boost factor applied to field c.

Next, a non-linear saturation $\frac{weight}{k_1+weight}$, in order to reduce the effect of term frequency to the final score is applied.

$$R(q,d) = \sum_{t\ in\ q} \frac{weight(t,d)}{k_1 + weight(t,d)} \cdot idf(t) \tag{2}$$

$idf(t)$ is computed as in the BM25 case

$$idf(t) = \log \frac{N - df(t) + 0.5}{df(t) + 0.5} \tag{3}$$

where N is the number of documents in the collection and df is the number of documents where appears the term t. We decided to include different boost factors because they allow us to change the importance given to the terms of a certain field.

2.4 Predictors

A novelty included in our system is the use of predictors for selecting the most promising ranking to a given patent topic when several queries to this topic are used. That is, the predictor allows us to create and launch different queries for a given topic, and to select the ranking that is considered to perform better.

We have applied predictors based on the dispersion where we suppose that if a ranking has a high value of standard deviation in their document scores, it could indicate that the ranking function has been able to discriminate between relevant and non-relevant documents [3]. On the other hand, if a low level of dispersion appears, because the ranking function has assigned similar weights, it can be interpreted as if it was not able to distinguish between relevant documents from non-relevant ones.

Therefore, two different prediction methods are tested. Given ranking list scores (RL) and their means $\mu(RL)$, we define two different predictors:

- Max SD
$$\sigma_{max} = max[\sigma(RL_{[1,d]}) : d \in RL]$$
- SD at Best cut point, where we use a ranking list size equivalent to 60 as test as it was inferred from the training collection

$$\sigma(RL) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(score(d_i) - \mu(RL))^2}$$

3 Submitted Runs

The different runs submitted to CLEF-IP were obtained by applying different decisions in some of the modules of the system and according to the results obtained at the development period.

The first decision for creating a run was the amount of terms to be used in a query. We took a look at the average length of each field in the collection and we performed several experiments that helped us to decide the amount of terms to be used. These terms are selected taking the most discriminative ones of each field according to KLD.

The first intuition is to think that the more terms in the query, the better the results. However, it is no clear that more terms in a query allow to improve results. Besides, an increase in the number of queries has associated an increase in the time for running a query. This is why we performed experiments considering less than 1000 terms per query. We experimented using terms from all the fields, as well as terms from only a field, discovering that the better results were obtained with terms from only the description field.

Secondly, we have also given different boost values to each field in order to increase the importance of the terms of a certain field in a query. In this case, we used the last year's observations about the importance of the title in this task. The other field which received a higher boost factor was description, which showed to be important in our experiments and improved the performance of the system when its boost factor was increased.

The next step is to decide what index is going to be used. Given a topic, the language of the topic (information that is available in the given topic) is used for deciding which index (and only this index) is going to be used. Thus, English topics are searched only in the index with English patents, etc. This decision leads to a decrease in performance when there are relevant patents in a language different to the one of the topic. In this case, we would need to include some kind of cross-lingual retrieval, what is not available in the current version of our system.

Finally, it must be considered that our system returns a ranking of patents, but participant runs at CLEF-IP must return specific versions of a patent. Since all the versions of a patent constituted the whole patent, given a patent in the ranking return by the IR engine, we changed the *id* of this patent with the *ids* of all the versions of this patent.

Taking into consideration all the aspects mentioned above, we submitted the following eight runs:

1. **Run 1 (Predictor maxsd):** given the rankings returned by the IR engine in the other runs (excluding the other predictor), this run selected for each topic the most promising ranking according to the Max SD predictor (described in Section 2.4).
2. **Run 2:** the queries build in this run contained the whole title, 30 terms from the abstract, 100 terms from the description and 100 terms from the claims section of the topic patent. The terms are selected taking the ones with are more discriminative according to KLD. The boost factors for all the fields are the same.
3. **Run 3:** the queries used in this run have the same terms that the ones in run 2. The only difference was to give the double boost factor to the terms of the description field over the other fields. With this run, we wanted to test our intuition of giving more importance to the description field.
4. **Run 4:** the queries used in this run have the same terms that the ones in run 2. The only difference was to give five times more boost factor to the title over the other fields. We created this run based on the observations made by last year's participants about the importance of the title.

5. **Run 5:** the queries used in this run have the same terms that the ones in run 2. The only difference was to give five times more boost factor to the title over the other fields, except the description field that receives the double boost factor that claims and abstract. This run was created for testing the effect of giving more importance to both the title and the description. The reason of giving much more importance to the title was that this field has less terms than the other fields, and it needs a greater boost factor.
6. **Run 6:** the queries build in this run contained only the 120 most discriminative terms from the description field of the topic patent. The terms were selected taking the ones which are more discriminative according to KLD. With this run, we wanted to test the importance of the description field by using only terms from it. The number of terms was taken based on the results in the development period, where good results were obtained considering only this field.
7. **Run 7:** the queries used in this run used only terms from the description field, as in run 6. However, we included 240 terms instead of 120 in order to check whether more terms from the description were helpful.
8. **Run 8 (Predictor sd 60):** given the rankings returned by the IR engine in the other runs (excluding the other predictor), this run selected for each topic the most promised ranking according to the predictor that uses standard deviation at cut 60 (described in Section 2.4).

4 Analysis of Results

The results obtained by the runs described in Section 3 are shown in Table 1. Besides, the results of the best participant system at CLEF-IP 2010 are also given for comparison purposes. The results are given according to different evaluation measures, where the main one is the Mean Average Precision (MAP).

Table 1. Results of the submitted runs.

run	map	set_P	P_5	P_10	P_50	P_100	set_R	R_5	R_10	R_50	R_100
best	0.2645	0.0273	0.4209	0.3482	0.1603	0.1027	0.6945	0.1279	0.197	0.3917	0.4809
run 1	0.0927	0.0095	0.179	0.1446	0.0752	0.0505	0.4375	0.0489	0.078	0.1874	0.2443
run 2	0.1051	0.0099	0.2027	0.1631	0.0812	0.0543	0.4565	0.0575	0.0907	0.2057	0.2663
run 3	0.0963	0.0095	0.1883	0.1535	0.0756	0.0507	0.4367	0.0526	0.0843	0.1903	0.2475
run 4	0.1054	0.0099	0.2026	0.1633	0.0813	0.0543	0.4569	0.0575	0.0911	0.2054	0.2664
run 5	0.0966	0.0095	0.1891	0.1531	0.0755	0.0507	0.4374	0.0529	0.084	0.1897	0.2476
run 6	0.0901	0.0095	0.1751	0.1408	0.0743	0.0512	0.4353	0.047	0.0743	0.1838	0.2477
run 7	0.0848	0.0089	0.1718	0.1336	0.0687	0.0465	0.4048	0.0476	0.0724	0.1677	0.221
run 8	0.0935	0.0094	0.181	0.1446	0.0741	0.0503	0.4341	0.0498	0.0792	0.186	0.2447

A first look at our results shows that our performance is far from the best system. However, our system performed in average a 0.1 of MAP, which was the average result

in last year's edition. Therefore, we think that it is an adequate result in this our first participation at CLEF-IP. Our best results were obtained according to recall, showing that our system was able to retrieve a 40% of relevant patents. However, our ranking needs to be improved if we want to obtain better results. We want to do a deeper study of our results in order to detect errors. Nevertheless, we think that the inclusion of some multilingual processing could lead us to outperform results.

Regarding the comparison among our runs, the best results where obtain where all the fields (title, abstract, claims and description) were taken into account. In these runs, the increase of the boost factor in the title allowed to outperform results, showing that different boost factors can be important in this task. However, boost factors must be selected carefully, since a higher boost factor in the description gave us worse results.

5 Conclusions and Future Work

We have described in this paper an approach for retrieving relevant patents to a input one. This has been our first participation in this task, where we have obtained results similar to the last year's average, what encourages us to continue our work. Our approach has been based on the consideration of different fields of a patent for creating the index and using the retrieval function, in combination with the use of the Kullback-Leibler divergence for detecting the most discriminative terms that will take part of a query.

Future work is focused on performing a deeper analysis of results in order to detect errors in our approach with the purpose of improving our system. Moreover, since CLEF-IP is multilingual task, we are thinking about how to include a cross-lingual search in our system.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation within the project QEAVis-Catiex (TIN2007-67581-C02-01), the Regional Government of Madrid under the Research Network MA2VICMR (S-2009/TIC-1542), the Education Council of the Regional Government of Madrid and the European Social Fund. We are very grateful to Juan Martínez-Romo for providing us its implementation of Language Models.

References

1. G. Roda F. Piroi and V. Zenz. CLEF-IP 2009 Evaluation Summary. In *Proceedings of CLEF 2009. LNCS*, 2009.
2. S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79—-86, 1951.
3. Joaquín Pérez-Iglesias and Lourdes Araujo. Ranking list dispersion as a query performance predictor. pages 371–374. 2009.
4. Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple bm25 extension to multiple weighted fields. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49, New York, NY, USA, 2004. ACM.