Automatic prior art searching and patent encoding at CLEF-IP '10

[1]Douglas Teodoro, [2]Julien Gobeill, [1]Emilie Pasche, [1]Dina Vishnyakova, [2]Patrick Ruch and [1]Christian Lovis,

[1]BiTeM group, Medical Informatics Service, University of Geneva
4 rue Gabrielle-Perret-Gentil, 1211 Geneva, Switzerland
{douglas.teodoro, emilie.pasche, dina.vishnyakova, christian.lovis}@hcuge.ch
[2]BiTeM group, Library and Information Sciences Department, University of Applied Sciences, 7 route de Drize, 1227 Carouge, Switzerland
{julien.gobeill, patrick.ruch}@hesge.ch

Abstract. In the intellectual property field two tasks are of high relevance: prior art searching and patent classification. Prior art search is fundamental for many strategic issues such as patent granting, freedom to operate and opposition. Accurate classification of patent documents according to the IPC code system is vital for the interoperability between different patent offices and for the prior art search task involved in a patent application procedure. In this paper, we report our experiments with prior art searching and patent classification in the context of CLEF-IP '10 evaluation track. In the *Prior Art Candidates* search task, we strongly improved our last year's model based on our experiments on training data (MAP 0.22), but official results, alas, were far from the expected ones (MAP 0.14). Regarding multilingual issues, our simple Google translator strategy achieved a 10% improvement. Nevertheless we think that the multilingual aspects in CLEF-IP'10 were less clear than for CLEF-IP'09. Finally, exploiting applicant's citations led to a 30% improvement, but their visibility depends on who (the applicant or the examiner) performs the prior art search in the simulated task. This issue needs clarification by the organizers for the forthcoming campaigns. In the *Classification* task, we apply the k-NN algorithm in the categorisation process and explore different retrieval models, ranking combinations and languages features in order to enhance our results. Using multi-collection in the classification process improved the results by 2%. Both the prior art search and classification systems are in the top three rank among the participants.

Keywords: Information retrieval, Prior art search, IPC encoding, Patent classification, k-NN

1 Introduction

According to EPO, it is estimated that 80% of the knowledge is found in patent documents. Due to its importance as source of knowledge and to the delay in patent analysis caused the growth of applications, new areas of knowledge and size of patent databases, new tools to automate patent searching and classification processes have become a hot topic in the last decades. As example, we can cite the challenges CLEF 2009, TREC-CHEM 2009-2010 and the workshops SIGIR 2000, ACL 2003 and NTCIR 3-8 which all have tasks dedicated to patent retrieval. In that context, the CLEF-IP 2010 evaluation track proposes two tasks for automation of prior art searching and of patent classification.

Prior art candidates search (PAC) is a fundamental task in patent processing, since many of the strategic issues in intellectual property rely upon retrieving patents that deal with a given invention. The most usual example is prior art search that applicants and examiners have to provide in order to grant an application. PAC may also be performed for invalidating another patent, for freedom to operate or for patent landscape. PAC primarily is an information retrieval task, in which recall is the most important measure, as one single document can invalidate a patent.

Automating the attribution of IPC codes to patent applications is important for several reasons: it assists patent officers in the patent classification task, aids inventors with the prior art search and helps referees to validate or refute a given application. When a patent application is considered or submitted, the search for previous inventions in the field relies crucially on accurate patent classification. The use of the assigned IPC code is also key information for searching patents across nations because of its language independence.

In this paper, we report the experience of the BiTeM group[1] in the CLEF-IP 2010 evaluation track. The challenge is divided into two tasks: *Prior Art Candidates search* (PAC) and *Classification* (CLS). In the *PAC* task, participants have to re-build the citations section of the 2000 applications belonging to the test set, mainly written in English. In the *CLS* task, patent applications written in English, French and German are automatically encoded using the IPC subclass descriptors.

We use an EPO patent collection composed by 2.7M documents and a set of 300 patent applications written in English, French and German to train the system. The assessments of our approaches are performed using 2000 documents in the *PAC* and *CLS* tasks. In order to improve classification we develop several re-ranking techniques that are further described.

The rest of this paper is organised as follows. In Section 2, the corpus and training data are depicted. Moreover, we describe the methods used to retrieve documents for the PAC task and the classification system. In Section 3, the results obtained are presented and remarks are discussed. In Section 4, the paper is concluded.

[1] http://eagl.unige.ch/bitem

2 Methods and data

2.1 Training and test data

In our experiments with the *PAC* and *CLS* tasks, we use a patent collection provided by EPO containing 2.7M patent documents, including A and B files. In total, the collection contains 1.3M patents. The distribution of patent documents according to their sections for the three different languages – *English*, *French* and *German* – is described in Table 1. The organisers also provide two sets of training (300 applications) and testing (2000 applications) documents.

In the *CLS* task, the fields *title*, *abstract*, *claim*, *description*, *applicant* and *citation* are used for indexing the collection. The average number of subclass codes per patent document (A and B) in the corpus is 8491 while the median is 2927. The majority of the codes (95%) are found in 100 or more documents. Six classes, A61K, A61P, C07D, H01L, G06F and G01N, are presented 100K in or more documents.

In the *PAC* task, organizers decided this year that the gold file would contain patent documents instead of patent families. Yet, we decided for time reasons to continue to work at the level of patent family. Hence, we continue to concatenate all documents relative to a given patent family in a unique virtual file. Once the run is computed, we simply split each virtual document in all its parts.

Table 1. Section distribution of patent documents (A and B) for the 3 three languages: German (*DE*), English (*EN*) and French (*FR*).

Section	DE	EN	FR
Title	93.2%	99.6%	93.2%
Abstract	11.1%	28.1%	3.2%
Claim	14.0%	36.3%	4.7%
Description	14.0%	36.3%	4.7%
Applicant	96.3%	96.3%	96.3%
Citation	19.1%	19.1%	19.1%

We use Terrier[2] as our information retrieval (IR) engine. Terrier implements several methods to calculate the similarity between documents: *BM25*, *BB2* (Bose-Einstein model for randomness), *InL2* (inverse document frequency model for randomness), among others and it is optimised to work with large collections. It is based on JAVA and freely available online.

[2] http://ir.dcs.gla.ac.uk/terrier

2.2 Classification system

In the classification experiments, we choose a classifier based on the k-NN algorithm. Some authors [1] have shown that k-NN, together with SVM, outperforms other approaches such as neural networks, Rocchio and Naïve Bayes. Compared to SVM, k-NN scales much better to larger systems that contain many features and classes, which is the case of the proposed task.

The classification system architecture is presented in Fig. 1. A query is provided to the IR engine, which ranks the first *k* documents d_j according to ranking model. The documents are mapped to their respective codes c_i and the codes are further re-ranked using the methods described in the next subsection. A ranked list of *n* codes is then created. Depending on the multi-lingual strategy, the topics are first translated using Google Language Tools[3] before being used as input to the IR engine.

We have tuned the number of neighbours *k* so that it maximises the precision at the top rank code. It happens to be 31 according to our experiments. The slope of the ranking models was suggested by the Terrier experiments with the .GOV collection and set to 0.2381 for the *BM25* and *DFR_BM25* models to 26.04 for the *PL2*.

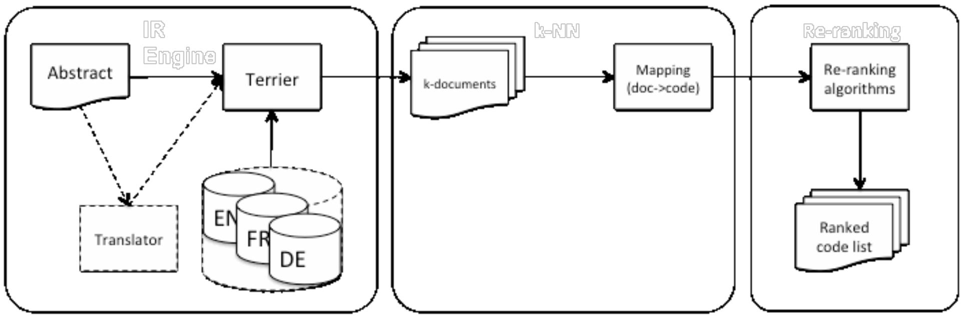

Fig. 1. Classification system: information retrieval engine (*left box*), k-NN algorithm (*middle box*) and re-ranking methods (*right box*).

2.3.1 Ranking strategies

In our attempt to improve the precision of the top *n* ranked codes, we have experimented several re-ranking algorithms as described in *Methods 1* to *7*. First, in *Method 1, 2* and *3* we compare the use of a single index containing all the three language documents against the use of a monolingual indexes and of query translation. Further, in *Method 4* we experiment the combination of different ranking models (*BM25*, *BM25_DFR* and *PL2*) and the combination of patent collections (derived from the different language in the

[3] http://translate.google.com

documents). As previously demonstrated [2], the combination of patent collections can enhance the classification results. Finally, analogously to the work of Xiao et al. [3], in *Method 5* we apply some simple re-ranking algorithms to the lists obtained in the *Methods 1* and *2*. These methods are implemented as follow:

Method 1. The collection containing *English* documents is indexed. Queries in *French* and *German* are translated to *English* before being submitted against this index. The model *BM25*, *DFR_BM25* and *PL2* are used to retrieve the documents. The codes are mapped and ranked using their frequency in the top *k* retrieved documents, as showed in Eq. (1) (see [3]):

$$\mathrm{codefreq}_{\mathrm{c_i}} = \sum_{j=1}^{k} f(c_i,d_j) \ , \qquad (\mathbf{1})$$

where *f* is defined by:

$$f(c_i,d_j) = \begin{cases} 1 \ , \ \mathrm{c_i} \in d_j \\ 0 \end{cases} . \qquad (\mathbf{2})$$

Method 2. An index is created using the whole collection. Queries in the three original languages are submitted against this index. The model *DFR_BM25* is used to calculate the document/query similarity. The codes are mapped and ranked using Eq. (1).

Method 3. Three different indexes are created from the *English, French* and *German* patent documents. Each index contains only sections from one language plus application and citation sections. Queries are translated to all the three languages and submitted against their respective index (DE->DE, EN->EN and FR->FR). The model *DFR_BM25* is used to fetch the documents. The codes are mapped and ranked using their frequency [Eq. (1)] in the top *k* retrieved documents.

Method 4. In this method, the results of *Method 3* are combined linearly in order to see how the combination of different collections can improve the results. Since the language indexes have different performances, they receive different weights in the combination: 1.00 for *English*, 0.25 for *German* and 0.15 for *French*. In the same line of thought, the results of *Method 1* are combined. As in the language combination, the models receive different weights with 0.05 for *BM25*, 1.00 for *DFR_BM25* and 0.01 for *PL2*. The weights were obtained from the training phase.

Method 5. In this method, the results obtained in *Methods 1* and *3* are re-ranked using the *rank list combination (rank combination)* method described in Xiao et al. [3]:

$$list_{\mathrm{c_i}} = \sum_{j=1}^{12} \frac{1}{\alpha r_{c_{ij}}} \; , \tag{3}$$

where r_{cij} is the code's rank in the ranked list j and varies between 1 and n. The lists used are *original* (Eq. (1)), *sum*, *listweak*, which are all described in Xiao et al. [3], and *citation*, which is derived as follow:

citation – It has been shown in previous experiments that citation is an important source of information for patent retrieval [5,6,7,8]. Eq. (4) shows how the codes c_i are ordered according to this method:

$$citation_{c_i} = \sum_{j=1}^{k} f(c_i, g(d_c, d_j)) \; , \tag{4}$$

where f is defined by:

$$f(c_i, g(d_c, d_j)) = \begin{cases} sim(d_j) \; , \; d_c \in d_{j,} \mathrm{c_i} \in d_c \\ 0 \end{cases} , \tag{5}$$

and d_c is a document cited by d_j. α is the weight of each ranking method, *original*, *sum*, *listweak* and *citation*, respectively set to 1.15, 1.00, 0.75 and 0.30.

2.3 Prior art search system

The prior art search used for CLEF-IP'10 system largely relies upon our last year's system [6]. Several additional strategies were evaluated throughout the pre-processing, the retrieval, and the post-processing steps. For this purpose, we worked with training data and simply computed a baseline run, and then tried to optimize the Mean Average Precision.

2.3.1 Pre-Processing strategies

Document Representation. In the framework of CLEF-IP 09 evaluation [6], we established that the best Document Representation for our system included *Title*, *Abstract*, *Claims*, and *IPC codes* (in both subclass and subgroup forms), but not *Description*. This year, we evaluated the contribution of other unexploited fields that are *Applicants* and *Inventors*. From the *Applicants* field contained in a patent document, we try to split the information and to extract three different fields that are the Applicants' names, the Applicants' countries, and the Applicant's address. The same strategy is used with the *Inventors* field.

Query Representation. Last year, we established that the best Query Representation for our system was the same we used for the collection plus *Description*. No further experiments were conducted regarding the Query Representation, unless including applicants and/or inventors information as for the collection.

Multilingual issues. This year, the collection includes documents in which English, French and/or German versions of each field can be present. Our strategy was to exclusively work in English and was simple: for each patent document, when the English version of a given field amongst *Title*, *Abstract* and *Claims* is available, we use this English version. Otherwise, if a French or a German version is available, we simply apply Google Translator on it. The same strategy is used for both documents and queries.

2.3.2 Retrieval strategies

The Information Retrieval step is performed with Terrier. Last year, we conducted a set of experiments in order to determine the best tuning, that was using Terrier BM25 with b=1.15 for weighting scheme, and Terrier Bose-Einstein for Query Expansion model. The same parameters are kept for CLEF-IP 2010.

2.3.3 Post-Processing strategies

Applicants' Countries. We investigated the hypothesis that the country of origin of the applicants, or the inventors, brings information, since citations are more likely to come from the same area due to a geographical bias [10].

Applicants' proposed Citations. Citations are extracted from the query *Description* field with simple regular expressions.

3 Results and discussions

In this section, we present the official results in the CLEF-IP '10 challenge for the *PAC* and *CLS* tasks.

3.1 Classification results

In our experiments in the *CLS* task, we submitted seven official runs, which are listed in Table 2. Comparing the baseline run *FREQ_Run1*, obtained from *Method 1* using *BM25* model, with *FREQ_Run2*, which is also obtained from *Method 1* but based in the divergence from randomness (*DFR_DM25*) model, we see a relevant improvement of 15% in the classifier performance.

When comparing *Method 1* (*FREQ_Run2*), which uses an *English* collection for indexing and translates the topics from other languages to *English*, with *Method 2* (*FREQ_Run3*), which uses indexes and queries from the three original languages, the results are very similar. From these results, we conclude that translation of the topics is not necessary if documents of the same topic's language are presented in the index. Otherwise, translation does not affect the classification results in the case of inexistent original topic's language in the index. This corroborates with our result in NTCIR-8 [2].

Our best run (MULTI_Run1) obtains 0.7281 of performance (MAP) and it uses *Method 4*, with the combination of the different language indexes obtained in *Method 3*. It shows an improvement of 1.9% over the best model of *Method 1* and 1.2% improvement over *Method 2*. We obtained similar results in [2] combining patent collections from different offices (JPO and USPTO). In MULTI_Run2, the combination of models obtained from *Method 1* also improves the results slightly (1.0%).

Table 2. BiTeM official results in *CLS* task.

Run[4]	Language	Index	Model	Ranking method	MAP
FREQ_Run1	EN	1	BM25	codefreq	0.6194
FREQ_Run2	EN	1	DFR_BM25	codefreq	0.7145
FREQ_Run3	DE+EN+FR	1	DFR_BM25	codefreq	0.7195
MULTI_Run1	DE+EN+FR	3	DFR_BM25	codefreq	0.7281
MULTI_Run2	EN	1	BM25+DFR_BM25 +PL2	codefreq	0.7216
LIST_MULTI_Run1	DE+EN+FR	3	DFR_BM25	list	0.7259
LIST_MULTI_Run2	EN	1	BM25+DFR_BM25 +PL2	list	0.7227

[4] Official run ids are prefixed by the group name, *bitem*, and suffixed by the task acronym, *CLS*.

Finally, the results of *Method 5* do not show any improvement when compared to their counterparts (MULTI_Run1 *vs* LIST_MULTI_Run1 and MULTI_Run2 *vs* LIST_-MULTI_Run2) from *Method 4*. These results were not expected from the training results, where we saw an improvement of 1.5% in *Methods 5*. We believe that it may have been due to overfitting.

In our attempt to analyse the reasons for the classification errors we try to correlate 1) the code classes, 2) the code document frequency (CDF) and 3) the size of queries with the query average precision. For 1) and 2) we do not find any clear correlation. The 50 best and 50 worst code classes have random distributions of codes with an overlap of approximately 30% between them. For the CDF correlation, the 50 best and 50 worst have also similar CDF. However, for hypotheses 3) we notice (Fig. 2) a linear increase in the average query precision with the size of the topic. We believe that this can give us some indications of where we should improve the classifier.

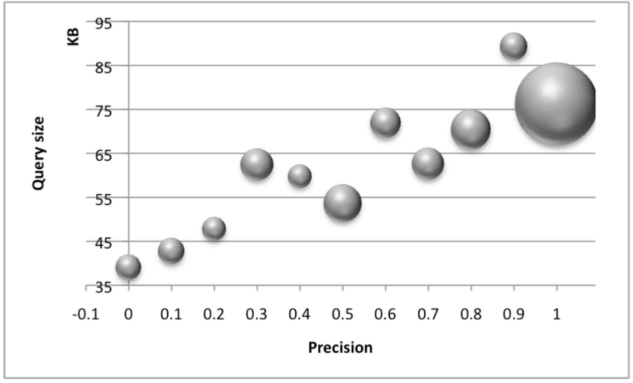

Fig. 2. *Query size vs Query precision* in the official run. Notice an almost linear relation between the average precision and the average query size for the 2000 topics.

3.2 Prior Art Candidates search results

All the reported experiments were conducted with training data. In order to evaluate a strategy, we compute a baseline run, using last year's best features, and then try to increase the Mean Average Precision.

3.2.1 Pre-Processing strategies

Multilingual issues. We start with a Baseline run, for which only original *English* is used, i.e. no translations. MAP for this baseline run is 0.106. Our simple translation strategy leads to a +8% improvement for MAP when applied for the collection, +10% when applied for both the collection and the queries (see Table 3). This improvement needs to be compared with more sophisticated strategies evaluated within this benchmark.

Table 3. Evaluating translation strategies in PAC task. EN means that only English fields are used, while EN+TR means that both English and translated fields are used.

Strategy	MAP
Baseline	0.106
Translation strategy applied to the collection	0.114
Translation strategy applied to the collection and the queries	0.117

Document Representation. We start with a Baseline run, which was computed using *Titles, Abstracts, Claims, IPC codes* for both collection and queries, and also *Description* for queries. We aim at evaluating the contribution of the different information contained in the *Applicants* and *Inventors* fields. Experiments show (see Table 4) that the information contained in both fields is relevant, and helpful for the Information Retrieval. Including applicants and inventors names respectively both leads to a +3% improvement. Including the country of origin seems to be ineffective. Addresses are noisy information in the patent. Yet, using them leads to +6% improvement. Our strategy was to split information contained in the *Applicants* or *Inventors* fields, in order to avoid what seems to be noise. The fact remains that the best results are obtained with all the fields, without any splitting.

Table 4. Evaluation of the different strategies for Document and Query Representation. *App* stands for *Applicants* and *inv* does for *Inventors*.

Strategy	MAP
Baseline	0.117
Including app names	0.120
Including app names and countries	0.120
Including app names and inv names	0.124
Including app names, plus inv names and countries	0.124
Including app names and addresses, plus inv names and addresses	0.131

3.2.2 Post-Processing strategies

Applicant's country. Closer analysis on training data reveals that, in the gold file, 50% of the cited patents share the same country of origin of the applicant than the patent used as query. Moreover, there seems to be clear patterns depending on the country. For Japanese patents, 70% of the cited patents come from Japan, while 10% come from USA. For French patents, 31% of the cited patents come from France, while 19% come from

Germany. We can hypothesize that rules inferred from these patterns can improve the model in a re-ranking way. Unfortunately, we tried several boosting or filtering strategies, but never obtained better results than the baseline.

Applicant's citation. Last year, a CLEF-IP'09 participant took benefit from the citations that the applicant provides in the Description field. In our report [6], we raised objections regarding this strategy, because this information may be not visible for the person who accomplishes the Prior Art, depending whether he is the applicant or the examiner. This year, since nothing forbids it, we chose to extract these applicants' citations contained in Description. Evaluated on training data, from a baseline run which achieves a MAP of 0.153, using applicant's citations leads to a +39% improvement (MAP of 0.213). Therefore, two different official runs were submitted, one called "Applicant's view" which simulates the Prior Art Search for the applicant, and another one called "Examiner's view" which simulates the Prior Art Search for the examiner and which includes the Applicant's citations.

3.2.3 Official runs

We hence submitted two runs, depending on the use of the applicant's citation. Final tuning on training data led to a MAP of 0.153 for the Applicant's View, but the official run only achieved a MAP of 0.106. For the Examiner's view (including applicants citations), we achieved a MAP of 0.213 for training data, but only 0.14 for official results (+32% compared to the Applicant's View).

4 Conclusion

In this paper we report our work in the *Prior Art Candidates search* and *Classification* in the CLEF-IP 2010 evaluation track. A corpus of 2.7M patents documents is used during the IR stage. The systems are evaluated with 2000 patent applications on both tasks.

In the *CLS* task, our system was ranked top three among the 7 participants, reaching 73% of mean average precision in the best run. The use of the multi-patent collections improved slightly the performance of the classification system. Moreover, the use of a multi-lingual collection or monolingual plus query translation showed to be equivalent concerning their classification performances. We plan to use the Catchword Index provided by WIPO to see if we can further improve our classification results. Moreover, we want to exercise the classification system using n-grams.

In the *PAC* task, our system, which largely relies on last year's system, was ranked top three among the 9 participants, while official results are disappointing regarding to the results obtained with training data. Further analysis needs to reveal the reason of such a bias. Our translation strategy was simple, but regarding to the weak amount of multilingual data, this +10% improvement is encouraging. We think that the multilingual aspects in CLEF-IP'10 were less clear than for CLEF-IP'09. Including inventors and applicants information is effective, but splitting them in different parts in order to reduce the noise is not. Finally, including applicants provided citations leads to a +35% improvement.

References

1. Sebastiani F.: Machine learning in automated text categorization. ACM Computing Surveys, 34, 1--47 (2002)
2. Teodoro D., Gobeill J., Ruch P. et al.: Automatic IPC encoding and novelty tracking for effective patent mining. In Proceedings of NTCIR-8 Workshop Meeting (2010).
3. Xiao T., Cao F., Li T., Song G., Zhou K., J Zhu., and Wang H.: Knn and re-ranking models for English patent mining at NTCIR-7. In Proceedings of NTCIR-7 Workshop Meeting, 2008.
4. Nanba H., Fujii A., Iwayama M., and Hashimoto T.: Overview of the patent mining task at the NTCIR-7 workshop. In Proceedings of NTCIR-7 Workshop Meeting (2008).
5. Gobeill J., Teodoro D., Pasche E., and Ruch. P.: Report on the TREC 2009 experiments: Chemical IR track. In the Eighteenth Text REtrieval Conference (TREC-18) (2009).
6. Gobeill J., Teodoro D., Pasche E., and Ruch. P.: Simple pre and post processing strategies for patent searching in the CLEF intellectual property track 2009. In CLEF 2009 Proceedings in Lecture Notes in Computer Sciences (in press).
7. Chakrabarti S., Dom B., Agrawal R., and Raghavan P.: Using taxonomy, discriminants, and signatures for navigating in text databases. In Proceedings of 23rd VLDB conference (1997).
8. Chakrabarti S., Dom B., Agrawal R., and Raghavan P.: Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. The VLDB Journal, 7,163--178 (1998).
9. Criscuolo P. and Verspagen B.: Does it matter where patent citations come from? Inventor vs. examiner citations in European patents. In Research Policy, Elsevier, vol. 37(10), 1892--1908, (2008)