Chemnitz at CLEF IP 2012: Advancing Xtrieval or a baseline hard to crack

Thomas Wilhelm, Jens Kürsten, and Maximilian Eibl

Chemnitz University of Technology,
Straße der Nationen 62, 09111 Chemnitz, Germany
{thomas.wilhelm,jens.kuersten,maximilian.eibl}@cs.tu-chemnitz.de
http://www.tu-chemnitz.de/cs/mi

Abstract. For the 2012 CLEF-IP Claims to passage task we reused and improved our Xtrieval framework. Our two-step approach comprises creating two Lucene indexes: one containing the whole patent application documents and one containing the same documents split into passages. We prepared three setups and conducted each with a translated and an untranslated topic set, which was just applied to the claims. The submitted setups differ in the way of retrieving the results and merging them. No further techniques were used. Therefore our experiments had very simple setups, which nevertheless achieved good results. There are still plenty of possible improvements, which can easily be tested with our framework, because it offers a comprehensive set of methods for conducting and evaluating retrieval experiments.

1 Introduction

This year we participated in CLEF-IP: Information Retrieval in the Intellectual Property Domain specifically in the Claims to passage task. In this task the topics consist of claims from patent application documents. The goal was to match passages within patent documents from the data collection to these claims.

The 2012 CLEF-IP data collection was the same as 2011[1], which was based on the MAREC collection[1] provided by the IRF[2]. It contains approximately 3.5 million well-formatted XML documents representing about 1.5 million patents.

As in our participation in the CLEF IP 2011 Prior Art task[2], we used our Xtrieval framework[3], which besides providing a common interface for different search engines[3] also offers a comprehensive set of methods for conducting and evaluating retrieval experiments.

2 Setup

In the course of our participation we re-engineered the Xtrieval framework. We further consolidated it and enhanced it for more speed. The speed improvements were necessary because of the size of the data collection (26 gigabytes

[1] MAREC IRF, http://www.ir-facility.org/prototypes/marec
[2] Information Retrieval Facility, http://www.ir-facility.org/
[3] Apache Lucene (http://lucene.apache.org/) and Terrier (http://terrier.org/)

compressed, 107 gigabytes uncompressed) and the necessity of executing several iteration of the experiments.

Our experiments last year showed that longer queries outperform shorter ones[2]. Another point in our thinking was that the passages alone are not sufficient and their context is probably important too. Taking this into account we opted for a two-step approach. For our participation in the ASR transcript task of MediaEval[4] we used a similar approach to first identify relevant transcripts and in a second step locate the exact time frame, which contained the relevant information.

We adapted this approach to the claims to passage scenario by creating two indexes: one containing the whole patent application documents and one containing the same documents split into passages.

For the passage index the following XPaths were use:

```
/patent-document/description/p
/patent-document/description/heading
/patent-document/claims/claim
/patent-document/abstract/p
```

For the document index the XPaths used were:

```
/patent-document/abstract/p
/patent-document/description/*
/patent-document/claims/claim
/patent-document/bibliographic-data/technical-data/invention-title
```

The Xtrieval framework has a very flexible and fast implementation for reading XML data collections based on the Jaxen library[4]. It exclusively relies on XPath for selecting the content and determining the destination fields in the index.

For the retrieval phase we prepared three different setups and conducted each with a translated and an untranslated topic set. We did not translate the whole patent but just the claims referenced in the topics. For the translation we used Google's Translator Toolkit[5] and translated all claims to the three most used languages of the data collection: English, German, and French. While other languages occur in the corpus, these three languages provide the vast majority of content according to an intermediate index we created.

All three setups share the following pre-processing steps:

1. instead of relying on the given language attributes we used a language detection[6] to eliminate content tagged wrong
2. bag of words

[4] jaxen: universal Java XPath engine, `http://jaxen.codehaus.org`
[5] Google Translator Toolkit, `http://translate.google.com/toolkit/`
[6] `http://code.google.com/p/language-detection/`

We used the standard tokenizer of Lucene, which splits a text stream into tokens and recognizes some entities like URLs and e-mail addresses. Then we applied our own filters (marked with *) and some from the Lucene package. If the filter depends on the token language this was considered for the following languages: English, German, French, Russian, Italian, and Spanish.

1. *LowerCaseFilter* - converts the token to lower case
2. *RemoveShortWordsFilter** - removes words shorter than 3 characters
3. *StopFilter* - removes stop words depending on the language
4. *RemoveNumbersFilter** - removes different kinds of numbers
5. *SnowballFilter*[7] - stems the token according to its language

The following setups differ just in the way of retrieving the results and merging them. No further techniques were used. That is to say: no fields (bag of words) or field weights, no relevance feedback, no language model, and no further query expansion.

2.1 Passages only (p)

This setup should be considered as our baseline, because we only used the claims specified in the topics and searched them in the passage index.

2.2 Documents combined with Passages (dp)

In this setup the query is constructed by merging the patent documents and the extracted claims. The content of the patent documents got a lesser weight than the claims to focus more on the claims. The queries were issued just on the passage index.

2.3 Documents before Passages (d2-p)

Our most sophisticated setup was the two-step approach, which was mentioned earlier. In the first step we retrieved a set of potentially relevant patent documents. For the second step their identifiers were used to amend the query to the passage index. The identifiers in the passage query are just optional to not exclude passages, which are still relevant but are not included in the first step. Some experiments with the provided test set showed beforehand, that limiting the second step to the results obtained in the first one will achieve a significant lower score.

[7] Snowball, `http://snowball.tartarus.org/`

3 Results

Table 1 shows the achieved results. Our best run, except for precision at passage level, is 'Documents before Passages' without any translation (tuc-d2-p). The second best run is 'Passages only', which is also the best at precision at passage level.

Both 'Documents combined with Passages' runs (tuc-dp and tuc-dpmt) gained equal results in the evaluation (see table 1). The source of this lies in the almost equal result set, which both runs produced. But as they slightly differ from each other, we can eliminate an error in our submission. We think it shows that on the one hand the passages had a small impact on the result set and on the other hand that the result set did not benefit as much as hoped from the translation.

Generally, the machine translation did not lead to better results. It even changed the results for the worse. This is likely owing to the universal translation service, which we used. A translation tailored for patents could perform better.

Table 1. Results (sorted by MAP at Document level)

	Document level			Passage level	
Run name	*PRES@100*	*Recall@100*	*MAP*	*MAP(D)*	*Precision(D)*
tuc-d2-p	0.1599	0.2094	0.0663	0.0385	0.0490
tuc-p	0.1430	0.1941	0.0501	0.0314	0.0522
tuc-dp	0.1363	0.1854	0.0424	0.0254	0.0383
tuc-dpmt	0.1363	0.1854	0.0424	0.0254	0.0383
tuc-d2-pmt	0.1218	0.1599	0.0614	0.0257	0.0297
tuc-pmt	0.0926	0.1428	0.0246	0.0167	0.0323

4 Summary and future work

Our focus this year was especially on improving and testing the framework. Therefore our experiments had very simple setups, which nevertheless achieved good results. The passages-only run without translation should be rated as our baseline and all other runs compared with it. This comparison shows that the only run outperforming our baseline, except for the precision at passage level, is our two-step approach without any translation (tuc-d2-p). All other runs achieved lower scores, which show they are not suited to improve the retrieval.

Because of the improved speed for the index and retrieval process in our framework, we could iterate more and experiment with different weights for the different combinations of documents and passages.

There are still plenty of possible improvements, which can easily be tested with our framework: pre-tokenization filters, token filters, retrieval systems (i.e. Lucene and Terrier), query expansion, and query reformulation. As the framework supports the calculation of different measures (i.e. PRES@n, MAP, and

so on) one can compare the results with previous experiments. This could more easily be done with tools like EvaluatIR[5] or Compeval[6]. Also an integration of these tools could be beneficial.

References

1. Piroi, F., Lupu, M., Hanbury, A., Zenz, V.: Clef-ip 2011: Retrieval in the intellectual property domain. [7]
2. Becks, D., Eibl, M., Jürgens, J., Kürsten, J., Wilhelm, T., Womser-Hacker, C.: Does patent ir profit from linguistics or maximum query length? [7]
3. Kürsten, J., Wilhelm, T.: Extensible retrieval and evaluation framework: Xtrieval. In Baumeister, J., Atzmüller, M., eds.: LWA. Volume 448 of Technical Report., Department of Computer Science, University of Würzburg, Germany (2008) 107–110
4. Schmidt, K., Korner, T., Heinich, S., Wilhelm, T.: A two-step approach to video retrieval based on asr transcriptions. In Larson, M., Rae, A., Demarty, C.H., Kofler, C., Metze, F., Troncy, R., Mezaris, V., Jones, G.J.F., eds.: MediaEval. Volume 807 of CEUR Workshop Proceedings., CEUR-WS.org (2011)
5. Armstrong, T.G., Moffat, A., Webber, W., Zobel, J.: Evaluatir: an online tool for evaluating and comparing ir systems. In Allan, J., Aslam, J.A., Sanderson, M., Zhai, C., Zobel, J., eds.: SIGIR, ACM (2009) 833
6. Wilhelm, T., Kürsten, J., Eibl, M.: A tool for comparative ir evaluation on component level. In Ma, W.Y., Nie, J.Y., Baeza-Yates, R.A., Chua, T.S., Croft, W.B., eds.: SIGIR, ACM (2011) 1291–1292
7. Petras, V., Forner, P., Clough, P.D., eds.: CLEF 2011 Labs and Workshop, Notebook Papers, 19-22 September 2011, Amsterdam, The Netherlands. In Petras, V., Forner, P., Clough, P.D., eds.: CLEF (Notebook Papers/Labs/Workshop). (2011)