Ad-hoc Mono- and Bilingual Retrieval Experiments at the University of Hildesheim

René Hackl, Thomas Mandl, Christa Womser-Hacker

University of Hildesheim, Information Science
Marienburger Platz 22
D-31141 Hildesheim, Germany
mandl@uni-hildesheim.de

Abstract

This paper reports on our participation in CLEF 2005's ad-hoc multi-lingual retrieval track. The ad-hoc task introduced Bulgarian and Hungarian as new languages. Our experiments focus on the two new languages. Naturally, no relevance assessments are available for these collections yet. Optimization was mainly based on French data from last year. Based on experience from last year, one of our main objectives was to improve and refine the n-gram-based indexing and retrieval algorithms within our system.

Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software

General Terms

Measurement, Performance, Experimentation

Keywords

Multilingual Retrieval, Fusion

1 Introduction

In the CLEF 2004 campaign, we tested an adaptive fusion system based on the MIMOR model (Womser-Hacker 1997) in the multi-lingual ad-hoc track (Hackl et al. 2005). In 2005, we applied our system based on Lucene[1] to the new multi-lingual collection: We focused on Bulgarian, French and Hungarian.

2 CLEF Retrieval Experiments with MIMOR

The optimization of the retrieval system parameters was based on the French corpus of last year. The tools employed this year include Lucene and Java™-based snowball[2] analyzers as well as the Egothor[3] stemmer. In previous CLEF results it has been pointed out, that a tri-gram index does not produce good results for French (McNamee & Mayfield 2004). A 4-gram or 5-gram indexing approach seems more promising. Consequently, we conducted some test runs experimenting with the following parameters:

- Document fields: only officially permitted document fields were indexed. These were indexed as they were as well as in an extra Field FULLTEXT enclosing all the contents from the other fields.
- Origin of query terms: query terms could come from either *title* or *description* fields or both.
- Phrase queries of ngram terms: length of phrases, Boolean operators for concatenating terms

[1] Lucene: http://lucene.apache.org
[2] Snowball: http://jakarta.apache.org/lucene/docs/lucene-sandbox/snowball/
[3] Egothor: http://www.egothor.org/

- Rigidity of phrase queries: non-exact phrase queries
- Blind relevance feedback (BRF): relevant documents/expansion term configuration (with Robertson Selection Value as term weighting scheme) and origin of expansion terms
- Weighting for all parameters mentioned above

The search field FULLTEXT provided best performance overall. Searching on the other fields by themselves or in combination and with weighting did not yield as good results as the simple full-text approach. The document field employed for BRF mattered much more. Here, best results were obtained with the TEXT field. For all runs, we used the stopword lists made available by the University of Neuchatel and made a few minor changes.

2.1 Query Construction for N-Gram Indexing

For phrase queries, the approach that worked best was one that constructed queries as follows: Given a query with 3 grams NG1, NG2, NG3 build the query so that q = "NG1" OR "NG2" OR "NG3" OR "NG1 NG2" OR "NG2 NG3" OR "NG1 NG2 NG3". Of course, such a query is likely to retrieve a lot of documents. Effectively, in almost all cases the system retrieved between 80% and 95% of all documents in the database. As Table 1 shows, these results can greatly be improved by applying a more sophisticated configuration on top of the depicted query construction. One means is to allow phrases to be non-exact match phrases, i.e. allow WITHIN or NEAR-like operations, denoted by *slop* in the table. Here, the best setting was five, values started getting visibly worse from 10 up.

Table 1. Effect of NEAR n term operation for boosting singles

NEAR n terms (slop)	4-gram		5-gram	
	Recall, max=915	Avg. prec.	Recall, max=915	Avg. prec.
1	688	0.277	698	0.272
2	687	0.278	698	0.272
3	684	0.277	698	0.276
4	687	0.277	701	0.277
5	691	0.272	703	0.276
6	691	0.271	702	0.274
7	689	0.271	699	0.273
8	689	0.274	697	0.272
9	688	0.274	696	0.270
10	686	0.278	694	0.270

Table 2. Result overview boosting singles, slop 5

BRF		4-gram		5-gram	
Documents	Terms	Recall, max=915	Avg. prec.	Recall, max=915	Avg. prec.
5	10	685	0.264	706	0.275
5	20	689	0.269	707	0.280
5	30	694	0.270	711	0.282
5	40	691	0.264	710	0.283
10	10	645	0.218	670	0.222
10	20	649	0.221	675	0.230
10	30	646	0.227	677	0.234
10	40	641	0.233	676	0.232

Table 3 gives optimized boost values for the n-gram retrieval experiments. The ratio of these figures has been determined experimentally. It can be seen that *title* terms are more important than *description* terms. Moreover, longer phrases are better than short ones, limited by the fact that starting with phrases of length 4, performance began to drop.

Table 3. Boost values for n-gram-based retrieval experiments

	Boosts according to origin	
# of terms in phrase	Origin: title	Origin: description
1	3, if short: 10	1, if short: 8
2	4	2
3	5	2
4	5	2

The single most important issue though are short terms. Phrase queries with only one term are of course just plain term queries. If, however, such a term query contains a term that has a smaller word length than the gram size, and taking into account that stopwords are eliminated, there is strong evidence that that term is highly important. In fact, most of these terms were acronyms or foreign words, e.g. in 2004 topics “g7“, “sida“ (French acronym for AIDS), “mir“ (Russian space station), “lady“ (Diana).

Blind relevance feedback had little impact on n-gram retrieval performance. For some queries, good short terms like those mentioned above were added to the query. However, terms selected by the algorithm received no special weight, i.e. they received a weight of one. Higher weights worsened the retrieval results. Furthermore, considering more than the top five documents for blind relevance feedback did not improve performance. Table 4 summarizes the results the best configurations achieved.

Table 4. Recall and average precision figures for ngram-based retrieval experiments. The table lists the best performing runs for all instances and combinations.

Indexing-Method	Optimization	Blind relevance feedback	Recall, max=915	Avg. prec.
4-gram	base run	none	507	0.126
4-gram	with single term phrases	none	551	0.178
4-gram	boosting single term phrases	none	684	0.26
4-gram	boosting singles, slop 5	none	691	0.272
4-gram	boosting, slop 5	5 docs, 30 terms	694	0.27
5-gram	boosting, slop 5	5 docs, 30 terms	711	0.282
5-gram	boosting	5 docs, 30 terms	707	0.275

2.2 Boosting Document Fields for Stemming Approaches

Subsequently, stemming replaced the n-gram indexing procedure in another test series. Three different stemmers were used: Egothor, Lucene, and Snowball. Table 5 shows the results of the base runs.

Table 5. Base runs with stemming algorithms

	Recall, max=915	Avg. prec.
Lucene Stemmer, base run	817	0.356
Snowball Stemmer, base run	821	0.344
Egothor Stemmer, base run	817	0.346

Queries that contained terms from both *title* and *description* fields from the topic files performed better than those that were based on only one source. The weighting of these terms, however, was a major impact factor. Several experiments with different boost values and blind relevance feedback parameters were carried out for each stemmer. The following tables 6, 7 and 8 show the results for the three stemmers.

Table 6. Results with Lucene stemmer

Boost Values			BRF		Results	
Title	**Description**	**BRF**	**Docs.**	**Terms**	**Recall, max=915**	**Avg. prec.**
9	3	1	5	10	856	0.379
9	3	1	5	20	857	0.388
9	3	1	5	30	863	0.405
9	3	1	5	40	857	0.402
9	3	2	5	10	855	0.379
9	3	2	5	20	854	0.390
9	3	2	5	30	857	0.403
9	3	2	5	40	855	0.392
9	3	3	5	10	855	0.379
9	3	3	5	20	857	0.385
9	3	3	5	30	861	0.394
9	3	3	5	40	858	0.388
base run					817	0.356

Table 7. Results with Snowball stemmer

Boost Values			BRF		Results	
Title	**Description**	**BRF**	**Docs.**	**Terms**	**Recall, max=915**	**Avg. prec.**
9	3	1	5	10	850	0.362
9	3	1	5	20	855	0.387
9	3	1	5	30	856	0.400
9	3	1	5	40	854	0.396
9	3	2	5	10	851	0.359
9	3	2	5	20	853	0.376
9	3	2	5	30	855	0.391
9	3	2	5	40	854	0.385
9	3	3	5	10	851	0.362
9	3	3	5	20	852	0.377
9	3	3	5	30	856	0.385
9	3	3	5	40	853	0.382
base run					821	0.344

Table 8. Results with Egothor stemmer

Boost Values			BRF		Results	
Title	**Description**	**BRF**	**Docs.**	**Terms**	**Recall, max=915**	**Avg. prec.**
9	3	1	5	10	849	0.359
9	3	1	5	20	850	0.376
9	3	1	5	30	852	0.389
9	3	1	5	40	848	0.388
9	3	2	5	10	852	0.354
9	3	2	5	20	850	0.385
9	3	2	5	30	851	0.390
9	3	2	5	40	837	0.389
9	3	3	5	10	855	0.351
9	3	3	5	20	849	0.382
9	3	3	5	30	843	0.389
9	3	3	5	40	831	0.386
base run					817	0.346

Yet again, searching on structured document parts instead of the full text was worse. More importantly, even the baseline run with an Egothor-based stemmer was better than any n-gram run. Table 9 summarizes the settings for the best runs. Boost values were applied to title, description and terms from blind relevance feedback in this order.

Table 9. Best runs of stemmer-based retrieval experiments

Stemmer	Run Type	Recall, max = 915	Avg. Prec.
Egothor	brf 5 10, boost 9 3 3	855	0.351
Egothor	brf 5 60, boost 9 3 1	843	0.394
Lucene	brf 5 30, boost 9 3 1	863	0.405
Snowball	brf 5 30, boost 9 3 1	856	0.400

3 Results of Submitted Runs

The parameters settings optimized with the French collection of CLEF 2004 were applied to the multi-lingual collection in 2005. We submitted monolingual runs for Bulgarian, French, Hungarian and domain specific (GIRT), bilingual runs for French and GIRT. For Bulgarian and Hungarian we employed the setting outlined above for two runs each – 4-gram and 5-gram: searching on full text representations, boosting single terms which were shorter than the grams length, using BRF (5 docs, 30 terms), and a slop of 5.

For French, we used the Lucene-stemmer and the settings derived above. Additionally, we carried out a 5-gram based run as a comparison to Bulgarian and Hungarian. Both of these monolingual were then reshaped by adding terms tentatively derived from the multilingual European terminology database Eurodicautom[4]. We extracted additional terms from the top three hits from the database, if they were available. At least one of the query terms had to be present in the resulting term list, no special subject domain was chosen. These terms were assigned a weight of one.

In the ad-hoc task, we submitted two English-to-French runs, one of which was enhanced by additional Eurodicautom terms, and one Russian-to-French run, all translated by ImTranslator[5]. The settings were the same as for the monolingual runs.

Table 9. Results from the CLEF 2005 Workshop. EDA = Euradicautom

	RunID	Languages	Run Type	retrieved	Relevant docs.	Avg. Prec.
monolingual	UHIBG1	Bulgarian	5-gram	587	778	0.189
	UHIBG2	Bulgarian	4-gram	597	778	0.195
	UHIHU1	Hungarian	5-gram	733	939	0.310
	UHIHU2	Hungarian	4-gram	776	939	0.326
	UHIFR1	French	Lucene stemmer	2346	2537	0.385
	UHIFR2	French	Lucene stemmer, EDA	2364	2537	0.382
	UHIFR3	French	5-gram	1816	2537	0.340
	UHIFR4	French	5-gram, EDA	1851	2537	0.274
bi-ling-ual	UHIENFR1	English -> French	Lucene stemmer, ImTranslator	2269	2537	0.337
	UHIENFR2	English -> French	Lucene stemmer, EDA	2307	2537	0.347
	UHIRUFR1	Russsian -> French	Lucene stemmer, ImTranslator	1974	2537	0.269

Considering the lack of experience with the new languages, the results are satisfying. However, more work with n-gram as well as stemming approaches are necessary for these languages.

[4] http://europa.eu.int/eurodicautom/Controller
[5] http://freetranslation.paralink.com/

4 Conclusion

For the participation in CLEF 2005, we could stabilize the n-gram indexing and search. The performance remains worse than for stemming based runs. We compared three stemmers with different parameter settings.

For future participations in ad-hoc tasks, we intend to apply the RECOIN (REtrieval COmponent INtegrator)[6] framework (Scheufen 2005). RECOIN is an object oriented JAVA framework for information retrieval experiments. It allows the integration of heterogeneous components into an experimentation system where many experiments may be carried out.

Acknowledgements

We would like to thank Nina Kummer and Sarah Risse for including the Egothor stemmer into our system. We also acknowledge the work of Viola Barth and Joachim Pfister who ported the trec_eval tool to Java.

References

Braschler, Martin (2004): Combination Approaches for Multilingual Text Retrieval, Information Retrieval. In: Information Retrieval, 7 (1/2) Kluwer pp. 183-204.

Carpineto, C.; de Mori, R.; Romano, G.; Bigi, B. (2001): An Information-Theoretic Approach to Automatic Query Expansion. In: ACM Transactions on Information Systems. 19 (1). pp. 1-27.

Hackl, René; Mandl, Thomas; Womser-Hacker, Christa (2005): Mono- and Cross-lingual Retrieval Experiments at the University of Hildesheim. In: Peters, Carol; Clough, Paul; Gonzalo, Julio; Kluck, Michael; Jones, Gareth; Magnini, Bernard (eds): Multilingual Information Access for Text, Speech and Images: Results of the Fifth CLEF Evaluation Campaign. Berlin et al.: Springer [LNCS 3491] pp. 165-169.

McNamee, Paul; Mayfield, James (2004): Character N-Gram Tokenization for European Language Text Retrieval. In: Information Retrieval 7 (1/2) pp. 73-98.

Scheufen, Jan-Hendrik (2005): Das RECOIN Framework für Information Retrieval Experimente. In: Mandl, Thomas; Womser-Hacker, Christa (eds.): Effektive Information Retrieval Verfahren in der Praxis: Proceedings Vierter Hildesheimer Information Retrieval und Evaluierungsworkshop (HIER 2005) Hildesheim, 20.7.2005. Konstanz: Universitätsverlag [Schriften zur Informationswissenschaft] to appear.

Womser-Hacker, C. (1997): Das MIMOR-Modell. Mehrfachindexierung zur dynamischen Methoden-Objekt-Relationierung im Information Retrieval. Habilitationsschrift. Universität Regensburg, Informationswissenschaft.

[6] http://recoin.sourceforge.net