The University of Amsterdam at CLEF 2003

Jaap Kamps Christof Monz Maarten de Rijke Börkur Sigurbjörnsson

Language & Inference Technology Group, University of Amsterdam
Nieuwe Achtergracht 166, 1018 WV Amsterdam, The Netherlands
E-mail: {kamps, christof, mdr, borkur}@science.uva.nl

Abstract

This paper describes our official runs for CLEF 2003. We took part in the monolingual task (for Dutch, Finnish, French, German, Italian, Russian, Spanish, and Swedish), and in the bilingual task (English to Russian, French to Dutch, German to Italian, Italian to Spanish). We also conducted our first experiments for the multilingual task (both multi-4 and multi-8), and took part in the GIRT task.

1 Introduction

In this year's CLEF evaluation exercise we participated in four tasks. We took part in the monolingual tasks for each of the eight non-English languages for which CLEF provides document collections (Dutch, Finnish, French, German, Italian, Russian, Spanish, and Swedish). For the second year running, we took part in the bilingual task, and for the first time, we took part in the multilingual task. We also conducted experiments for the GIRT task.

Our participation in the *monolingual* task was motivated by a number of aims. Our first aim was to experiment with a number of linguistically motivated techniques, in particular stemming algorithms for all European languages [15]. Our second aim was to continue earlier experiments on compound splitting [10, 8], this time for all the compound rich languages, Dutch, German, Finnish, and Swedish. A third aim was to continue our experiments with knowledge-poor techniques, by using character n-grams. Our final aim was to experiment with combinations of runs, such as the combination of linguistically motivated and knowledge-poor techniques, and the combination of different weighting schemes. In the *bilingual* task our aim was to evaluate the robustness of our monolingual retrieval results, and to experiment with a variety of translation resources [16, 12, 1]. The *multilingual* task was new to us. Our aims for this task were to experiment with unweighted and weighted combination methods, and with the effect of multiple languages on retrieval effectiveness. We continued our participation in the *GIRT* task. This year, our aim was to experiment with an improved version of a document reranking strategy, tailored to the presence of classification information in the collection [8].

The paper is organized as follows. In Section 2 we describe the FlexIR system as well as the approaches used for each of the tasks in which we participated. Section 3 describes our official retrieval runs for CLEF 2003. In Section 4 we discuss the results we have obtained. Finally, in Section 5, we offer some conclusions regarding our document retrieval efforts.

2 System Description

2.1 Retrieval Approach

All retrieval runs used FlexIR, an information retrieval system developed at the University of Amsterdam. The main goal underlying FlexIR's design is to facilitate flexible experimentation with a wide variety of retrieval components and techniques. FlexIR is implemented in Perl and supports many types of preprocessing, scoring, indexing, and retrieval tools, which proved to be a major asset for the wide variety of tasks in which we took part this year. Building on last year's experience we continued our work on combining different runs. Last year's focus was on combining runs that were generated using different morphological normalization processes. This year we added a further dimension: in addition to the Lnu.ltc-based vector space model that was used at CLEF 2001 and CLEF 2002, we wanted to experiment with other retrieval models, especially with the Okapi weighting scheme and with language models.

Retrieval Models. FlexIR supports several retrieval models, including the standard vector space model, language models, and probabilistic models, all of which were used to obtain combined runs. Combined runs using the vector space model all use the Lnu.ltc weighting scheme [2] to compute the similarity between a query and a document. For the experiments on which we report in this note, we fixed *slope* at 0.2; the pivot was set to the average number of unique words per document. We also experimented with a number of alternative weighting schemes. For runs with the Okapi weighting scheme [13], we used the following tuning parameters: $k_1 = 1.5$ and $b = 0.55$ for Dutch; $k_1 = 1.5$ and $b = 0.55$ for German; $k_1 = 1.2$ and $b = 0.50$ for Spanish; and $k_1 = 0.8$ and $b = 0.35$ for Swedish. For runs with a language model [6], we used a uniform query term importance weight of 0.15.

Morphological Normalization. After CLEF 2002 we carried out extensive experiments with different forms of morphological normalizations for monolingual retrieval in all of the CLEF 2002 languages [7]. The options considered included word-based runs (where the tokens as they occur in the documents are indexed without processing), stemming (where we used stemmers from the Snowball family of stemmers), lemmatizing (where we used the lemmatizer built into the TreeTagger part-of-speech tagger), and compound splitting (for compound forming languages such as Dutch, Finnish, German, and Swedish). We also experimented with character n-grams (of length 4 and 5). The main lessons learned were two-fold: there is no language for which the best performing run significantly improves over the "split, and stem" run (treating splitting as a no-op for non-compound forming languages); and the hypothesis that 4-gramming is the best strategy is refuted for Spanish only. Notice that the comparisons did not involve combinations of runs, but only, what we call, *base runs*.

Stemming — To produce our base runs for CLEF 2003, we followed our own advice [7]. For all languages we created split-and-stemmed runs as well as n-gram runs. We used the family of Snowball stemming algorithms, available for all the nine languages of the CLEF collections. Snowball is a small string processing language designed for creating stemming algorithms for use in information retrieval [15].

Decompounding — For the compound rich languages, Dutch, German, Finnish, and Swedish, we also apply a decompounding algorithm. We treat all the words occurring in the CLEF corpus as potential base words for decompounding, and also use their associated collection frequencies. We ignore words of length less than four as potential compound parts, thus a compound must have at least length eight. As a safeguard against oversplitting, we only regard compound parts that have a higher collection frequency than the compound itself. We consider linking elements -s-, -e-, and -en- for Dutch; -s-, -n-, -e-, and -en- for German; -s-, -e-, -u-, and -o- for Swedish; and none for Finnish. We prefer a split with no linking element over a split with a linking element, and a split with a single character linker over a two character linker.

Each document in the collection is analyzed and if a compound is identified, the compound is kept and all of its parts are added to the document. Compounds occurring in a query are analyzed in a similar way: the parts are simply added to the query. Since we expand both the documents and the queries with compound parts, there is no need for compound formation [11].

n-Gramming — Zero-knowledge language independent runs were generated using character n-grams, with $n = 5$ for Finnish and $n = 4$ for all other languages; n-grams were not allowed to cross word boundaries.

Character Encodings. Until CLEF 2003, the languages of the CLEF collections all used the Latin alphabet. The addition of the new CLEF language, Russian, is challenging for the use of a non-Latin alphabet. The Cyrillic characters used in Russian can appear in variety of font encodings. The collection and topics are encoded using the UTF-8 or Unicode character encoding. We converted the UTF-8 encoding into a 1-byte per character encoding KOI8 or KOI8-R (for *Kod Obmena Informatsii* or Code of Information Exchange).[1] We did all our processing, such as lower-casing, stopping, stemming, and n-gramming, on documents and queries in this KOI8 encoding. Finally, to ensure the proper indexing of the documents using our standard architecture, we converted the resulting documents into the Latin alphabet using the Volapuk transliteration. We processed the Russian queries in the same way as the documents.

Stopwords. Both topics and documents were stopped using the stopword lists from the Snowball stemming algorithms [15], for Finnish we used the Neuchâtel-stoplist [4]. Additionally, we removed topic specific phrases such as 'Find documents that discuss ...' from the queries. We did not use a stop stem or stop n-gram list, but we first used a stop *word* list, and then stemmed/n-grammed the topics and documents.

[1] We used the excellent Perl package `Convert::Cyrillic` for conversion between character encodings and for lower-casing Cyrillic characters.

Blind Feedback. Blind feedback was applied to expand the original query with related terms. We experimented with different schemes and settings, depending on the various indexing methods and retrieval models used. For our Lnu.ltc and Okapi runs term weights were recomputed by using the standard Rocchio method [14], where we considered the top 10 documents to be relevant and the bottom 500 documents to be non-relevant. We allowed at most 20 terms to be added to the original query.

Combined Runs. For each of the CLEF 2003 languages we created base runs using a variety of indexing methods (see below). In addition, we used different retrieval models to create further runs (again, see below for details). We then combined our base runs using one of two methods, either a weighted interpolation or a three-way combination, as we will now explain.

The weighted interpolation was produced as follows. First, we normalized the retrieval status values (RSVs), since different runs may have radically different RSVs. For each run we reranked these values in $[0,1]$ using:

$$RSV_i' = \frac{RSV_i - min_i}{max_i - min_i};$$

this is the Min_Max_Norm considered in [9]. Next, we assigned new weights to the documents using a linear interpolation factor λ representing the relative weight of a run:

$$RSV_{new} = \lambda \cdot RSV_1 + (1-\lambda) \cdot RSV_2.$$

For $\lambda = 0.5$ this is similar to the simple (but effective) combSUM function used by Fox and Shaw [5]. The interpolation factors λ were obtained from experiments on the CLEF 2000, 2001, and 2002 data sets (whenever available). When we combined more than two runs, we gave all runs the same relative weight, resulting effectively in the familiar combSUM.

For the GIRT task, we created alternative base runs based on the usage of the keywords in the collection, and combined these with the text-based runs.

3 Runs

We submitted a total of 34 retrieval runs: 15 for the monolingual task, 8 for the bilingual task, 3 for the multi-4 task, 5 for the multi-8 task, and 3 for the GIRT task. Below we discuss these runs in some detail.

3.1 Monolingual Runs

All our monolingual runs used the title and description fields of the topics. Table 1 provides an overview of the runs that we submitted for the monolingual task. The third column in Table 1 indicates the type of run:

- *(Split+)Stem* — topic and document words are stemmed and compounds are split (for Dutch, German, Finnish, Swedish), using the morphological tools described in Section 2. For all eight languages, we use a stemming algorithm from the Snowball family [15].
- *n-Gram* — both topic and document words are n-grammed, using the settings discussed in Section 2. For Finnish we use 5-grams, and for all other languages we use 4-grams.
- *Combined* — two base runs are combined, an n-gram run and a morphological run, using the interpolation factor λ given in the fourth column.

Additionally, for two languages where we expected the stemming algorithm to be particularly effective, Dutch and Spanish, we submitted the combination of three weighting schemes on the stemmed index (where we use decompounding for Dutch). We combine the run with Lnu.ltc with runs made with Okapi and a language model.

Furthermore, we experimented with the Okapi weighting scheme on the stemmed, and decompounded indexes for German and Swedish, and submitted the combination with the 4-gram-run using the Lnu.ltc scheme.

Finally, we also submitted three base runs for Russian, a word-based run, a stemmed run, and a 4-gram run, all using the the settings discussed in Section 2.

Run	*Language*	*Type*	*Factor*
UAmsC03GeGe4GiSb	DE	4-Gram/Split+stem	0.36
UAmsC03GeGe4GSbO	DE	4-Gram (Lnu)/Split+stem (Okapi)	0.18
UAmsC03SpSp4GiSb	ES	4-Gram/Stem	0.35
UAmsC03SpSpSS3w	ES	Stem (Lnu/Okapi/LM)	-
UAmsC03FiFi5GiSb	FI	5-Gram/Split+stem	0.51
UAmsC03FrFr4GiSb	FR	4-Gram/Stem	0.66
UAmsC03ItIt4GiSb	IT	4-Gram/Stem	0.405
UAmsC03DuDu4GiSb	NL	4-Gram/Split+stem	0.25
UAmsC03DuDuSS3w	NL	Split+stem (Lnu/Okapi/LM)	-
UAmsC03RuRuWrd	RU	Word	-
UAmsC03RuRuSbl	RU	Stem	-
UAmsC03RuRu4Gr	RU	4-Gram	-
UAmsC03RuRu4GiSb	RU	4-Gram/Stem	0.60
UAmsC03SwSw4GiSb	SV	4-Gram/Split+stem	0.585
UAmsC03SwSw4GSbO	SV	4-Gram (Lnu)/Split+stem (Okapi)	0.315

Table 1: Overview of the monolingual runs submitted. For combined runs column 3 gives the base runs that were combined, and column 4 gives the interpolation factor λ.

3.2 Bilingual Runs

We submitted a total of 7 bilingual runs, for English to Russian, French to Dutch, German to Italian, and Italian to Spanish. All our bilingual runs used the title and description fields of the topics. For the bilingual runs, we experimented with the WorldLingo machine translation [16] for translations into Dutch, Italian, and Spanish. For translation into Russian we used the PROMT-Reverso machine translation [12].

Table 2 provides an overview of the runs that we submitted for the bilingual task. The third column in Table 2 indicates the type of run. For all the four bilingual pairs, we submitted a combination of the stemmed (and

Run	*Topics*	*Documents*	*Type*	*Factor*
UAmsC03GeIt4GiSb	DE	IT	4-Gram/Stem	0.7
UAmsC03EnRu4Gr	EN	RU	4-Gram	-
UAmsC03EnRuSbl	EN	RU	Stem	-
UAmsC03EnRu4GiSb	EN	RU	4-Gram/Stem	0.6
UAmsC03FrDu4Gr	FR	NL	4-Gram	-
UAmsC03FrDuSblSS	FR	NL	Split+stem	-
UAmsC03FrDu4GiSb	FR	NL	4-Gram/Split+stem	0.3
UAmsC03ItSp4GiSb	IT	ES	4-Gram/Stem	0.4

Table 2: Overview of the bilingual runs submitted. For combined runs column 4 gives the base runs that were combined, and column 5 gives the interpolation factor λ.

decompounded for Dutch) run with a 4-gram run. Since we put particular interest in the translations to Dutch, we also submitted the two underlying base runs. Finally, we also submitted the stemmed and n-grammed base runs for the translation into Russian.

3.3 Multilingual Runs

We submitted a total of 8 multilingual runs, three for the small multilingual task and five for the large multilingual task, all using the title and description of the English topic set. For the multilingual runs, we experimented with the WorldLingo machine translation [16] for translations into Dutch, French, German, Italian, and Spanish. For translation into Swedish we used the first translation mentioned in the Babylon online dictionary [1].

Table 3 provides an overview of the runs that we submitted for the multilingual task. The fourth column in Table 3 indicates the document sets used. In effect, we conducted three sets of experiments: (i) on the four language small multilingual set (English, French, German, and Spanish), (ii) on the six languages for which we have an acceptable machine translation (also including Dutch and Italian), and (iii) on the seven languages (also including Swedish, but no Finnish documents) for which we have, at least, an acceptible bilingual dictionary.

Run	*Task*	*Topics*	*Documents*	*Type*
UAmsC03EnM44Gr	multi-4	EN	DE, EN, ES, FR	4 × 4-Gram
UAmsC03EnM44GiSb	multi-4	EN	DE, EN, ES, FR	4 × 4-Gram/(Split+)stem
UAmsC03EnM4SS4G	multi-4	EN	DE, EN, ES, FR	4 × 4-Gram, 4 × (Split+)stem
UAmsC03EnM84Gr6	multi-8	EN	DE, EN, ES, FR, IT, NL	6 × 4-Gram
UAmsC03EnM8SS4G6	multi-8	EN	DE, EN, ES, FR, IT, NL	6 × 4-Gram, 6 × (Split+)stem
UAmsC03EnM84Gr	multi-8	EN	DE, EN, ES, FR, IT, NL, SV	7 × 4-Gram
UAmsC03EnM84GiSb	multi-8	EN	DE, EN, ES, FR, IT, NL, SV	7 × 4-Gram/(Split+)stem
UAmsC03EnM8SS4G	multi-8	EN	DE, EN, ES, FR, IT, NL, SV	7 × 4-Gram, 7 × (Split+)stem

Table 3: Overview of the multilingual runs submitted. Column 5 indicates the base runs used to generate the multilingual run.

For each of these experiments, we submitted a number of combined runs, where we used the (unweighted) combSUM rule introduced by [5]. First, we combined a single, uniform run per language, in all cases a 4-gram run. Second, per language we formed a weighted combination of the 4-gram and stemmed run (with decompounding for Dutch, German, and Swedish). We used the following relative weights of the 4-gram run: 0.6 (Dutch), 0.4 (English), 0.7 (French), 0.5 (German), 0.6 (Italian), 0.5 (Spanish), and 0.8 (Swedish). These runs of the different languages were combined using the combSUM rule. Third, we simply formed a pool of all 4-gram and stemmed runs (where we decompounded for Dutch, German, and Swedish) for all languages, and derived a combined run using the combSUM rule.

3.4 The GIRT Task

We submitted a total of 3 runs for the GIRT task, all using both the German topics and collection. We used the title and description fields of the topics, and used the title and abstract fields of the collection. We experimented with a reranking strategy based on the keywords assigned to the documents, the resulting rerank runs also use the controlled-vocabulary fields in the collection.

Table 4 provides an overview of the runs that we submitted for the GIRT task. The fourth column in Table 4 indicates the type of run. The stemmed and 4-grammed runs mimics the settings of our monolingual runs for

Run	*Topics*	*Documents*	*Type*
UAmsC03GeGiWrd	DE	DE	Word
UAmsC03GeGi4GriR	DE	DE	Reranking of Stem
UAmsC03GeGiSbliR	DE	DE	Reranking of 4-Gram

Table 4: Overview of the runs submitted for the GIRT task.

German, although we did not use decompounding. The word-based run serves as a baseline for performance. The other two runs experiment with an improved version of our keyword-based reranking strategy introduced at CLEF 2002 [8]. We calculate vectors for the keywords based on their (co)occurrences in the collection. The main innovation is in the use of higher dimensional vectors for the keywords, for which we use the best reduction onto a 100-dimensional euclidean space. The reranking strategy is as follows. We calculate vectors for all initially retrieved documents, by simply taking the mean of the vectors of keywords assigned to the documents. We calculate a vector for a topic by taking the relevance-weighted mean of the top 10 retrieved documents. We now have a vector for each of the topics, and for each of the retrieved documents. Thus, ignoring the RSV of the retrieved documents, we can simply rerank all documents by increasing euclidean distance between the document and topic vectors. Next, we combine the original text-based similarity scores of the baserun, with the keyword-based distances using the unweighted combSUM rule of [5].

4 Results

This section summarizes the results of our CLEF 2003 submissions.

4.1 Monolingual Results

Table 5 contains our non-interpolated average precision scores for all languages. In addition to the scores for our submitted runs, the table also lists the scores for the base runs that were used to generate the combined runs.

	Dutch	*Finnish*	*French*	*German*	*Italian*	*Russian*	*Spanish*	*Swedish*
(Split+)Stem	*0.4984*	*0.4453*	*0.4511*	*0.4840*	*0.4726*	0.2536	*0.4678*	*0.3957*
n-Gram	*0.4996*	*0.4774*	*0.4616*	*0.5005*	*0.4227*	**0.3030**	*0.4733*	*0.4187*
Combined	0.5072	**0.5137**	**0.4888**	0.5091	**0.4781**	0.2988	0.4841	0.4371
(% Change)	(+1.52%)	(+7.60%)	(+5.89%)	(+1.72%)	(+1.16%)	(−1.39%)	(+2.28%)	(+4.39%)
More runs:								
Lnu/Okapi/LM	**0.5138**	–	–	–	–	–	**0.4916**	–
Lnu/Okapi	–	–	–	**0.5200**	–	–	–	**0.4556**
Words	–	–	–	–	–	0.2551	–	–

Table 5: Overview of MAP scores for all submitted monolingual runs and for the underlying base runs. Best scores are in boldface; base runs that were not submitted are in italics. The figures in brackets indicate the improvement of the combined run over the best underlying base run.

Both the stemmed runs (with decompounding for the compound-rich languages) and the n-gram runs perform well, with the n-gram runs outperforming the stemmed runs for seven out of eight languages. Only for Italian, the stemmed run performs better than the 4-gram run. This deviating behavior for Italian may be due to the different ways of encoding marked characters in the Italian sub-collections [3].

The (binary) combination of the stemmed and n-grammed base runs leads to improvements over the best underlying score for seven out of eight languages. Only for Russian, the 4-gram run is somewhat better than the combined run. This may be due to the difference in performance of both underlying base runs. The Snowball stemmer for Russian has no evident effect for the monolingual topics, the score is even a fraction lower than the score of a plain word-based run.

At this time we do not know yet whether the high scores of the combinations involving an Okapi base run are to the effect of combining or due to the training and fine-tuning that we performed for our Okapi base runs.

4.2 Bilingual Results

To begin with, Table 6 shows our MAP scores for the four bilingual sub tasks: French to Dutch, German to Italian, Italian to Spanish, and English to Russian.

	French to Dutch	*German to Italian*	*Italian to Spanish*	*English to Russian*
(Split+)Stem	0.3693	*0.3402*	*0.3160*	**0.2270**
n-Gram	0.3803	*0.3411*	***0.3588***	0.1983
Combined	**0.3835**	**0.3830**	0.3535	0.2195
(% Change)	(+0.84%)	(+12.28%)	(−1.48%)	(−3.30%)

Table 6: Overview of MAP scores for all bilingual runs. Best scores are in boldface. The figures in brackets indicate the improvement of the combined run over the best underlying base run.

As for the monolingual runs, both the stemmed (and decompounded for Dutch) and the 4-grammed indexes perform well, with the 4-gram runs outperforming the stemmed runs for three out of four languages. The exception, this time, is Russian where the stemmed run is now better than the 4-gram run. The combination is effective for French to Dutch and German to Italian. For Italian to Spanish, the 4-gram base run (not submitted) scores better than the combined run, and for English to Russian the stemmed run is scoring better than the combined run.

A conclusion on the effectiveness of the Russian stemmer turns out to be premature. Although the stemmer failed to improve retrieval effectiveness for the monolingual Russian task, it turns out to be effective for the bilingual Russian task.

	Dutch	*Italian*	*Spanish*	*Russian*
Best monolingual	0.5138	0.4781	0.4916	0.3030
Best bilingual	0.3835	0.3830	0.3588	0.2270
(% Change)	(−25.36%)	(−19.89%)	(−27.01%)	(−25.08%)

Table 7: Decrease in effectiveness for bilingual runs.

Table 7 shows the decrease in effectiveness compared to the best monolingual run for the respective target language. The difference ranges from a 20% to a 27% decrease in MAP score. This seems quite acceptable, considering that

we used a simple, straightforward machine translation for the bilingual tasks [16]. This result gives us some confidence in the robustness of the morphological normalization methods employed for building the indexes.

4.3 Multilingual Results

Table 8 shows our multilingual MAP scores for the small multilingual task (covering four languages) and for the large multilingual task (covering all eight non-English CLEF languages).

	Multi-4	*Multi-8*	
		(without FI/SV)	*(without FI)*
n-Gram	0.2953	0.2417	0.2467
Combined n-Gram/(Split+)Stem	**0.3341**	*0.2797*	0.2871
n-Gram and (Split+)Stem	0.3292	0.2755	**0.2884**

Table 8: Overview of MAP scores for all multilingual runs. Best scores are in boldface; runs that were not submitted are in italics.

For the small multilingual task, first making a weighted combination per language outperforms the unweighted combination of all n-gram and stemmed runs. For the large multilingual task, when using only six of the eight languages, we see the same pattern: first making a weighted combination run per language (not submitted) outperforms the unweighted combination. However, when we include our Swedish results in the large multilingual task, we see that the unweighted combination of all the 4-gram and stemmed base even slightly outperforms the weighted combinations.

Our results show that multilingual retrieval on a subpart of the collection (leaving out one or two languages) can still be an effective strategy. However, the results also indicate that the inclusion of further languages does consistently improve MAP scores.

4.4 Results for the GIRT Task

Table 9 contains our MAP scores for the GIRT monolingual task. In addition to the scores for our submitted runs, the table also lists the scores for the base runs that were used to generate the rerank runs.

	GIRT	*Rerank*	*% Change*
Words	0.2360	–	–
Stemmed	*0.2832*	0.3361	+18.68%
4-Gram	*0.3449*	**0.3993**	+15.77%

Table 9: Overview of MAP scores for all submitted GIRT runs, and for the underlying base suns. Best scores are in boldface; base runs that were not submitted are in italics.

The results for the GIRT tasks show, on the one hand, the effectiveness of stemming and n-gramming approaches over a plain word index. On the other hand, the results show a significant improvement of retrieval effectiveness due to our keyword-based reranking method. The improvement comes on top of the improvement due to blind feedback, and consistent even for high performing base runs.

5 Conclusions

The experiments on which we report in this note indicate a number of things. First, morphological normalization does improve retrieval effectiveness, especially for languages that have a more complex morphology than English. We also showed that n-gram-based approaches can be a viable option in the absence of linguistic resources to support deep morphological normalization. Although no panacea, the combination of runs provides a method that may help improve base runs, even high quality base runs. The interpolation factors required for the best gain in performance seem to be fairly robust across topic sets. Moreover, the effectiveness of the unweighted combination of runs is usually close to the weighted combination, and seems to gain in effectiveness the more base runs are available. Our bilingual experiments reconfirmed that a simple machine translation strategy can be effective for bilingual retrieval. The combination of bilingual runs, in turn, leads to an effective strategy for multilingual retrieval. Finally, our results for domain-specific retrieval show the effectiveness of stemming and n-gramming even for specialized collection. Moreover, manually assigned classification information in such scientific collections can be fruitfully exploited for improving retrieval effectiveness.

Acknowledgments

We want to thank Harald Hammarstrom for advice on Finnish and Swedish, Vera Hollink for technical support, and Valentin Jijkoun for help with the Russian collection. Jaap Kamps was supported by NWO under project number 400-20-036. Christof Monz was supported by the Physical Sciences Council with financial support from NWO under project 612-13-001, and by a grant from NWO under project number 220-80-001. Maarten de Rijke was supported by grants from NWO, under project numbers 612-13-001, 365-20-005, 612.069.006, 612.000.106, 220-80-001, and 612.000.207.

References

[1] Babylon. Online dictionary, 2003. http://www.babylon.com/.

[2] C. Buckley, A. Singhal, and M. Mitra. New retrieval approaches using SMART: TREC 4. In D.K. Harman, editor, *The Fourth Text REtrieval Conference (TREC-4)*, pages 25–48. National Institute for Standards and Technology. NIST Special Publication 500-236, 1996.

[3] CLEF. Cross language evaluation forum, 2003. http://www.clef-campaign.org/.

[4] CLEF-Neuchâtel. CLEF resources at the University of Neuchâtel, 2003. http://www.unine.ch/info/clef.

[5] E.A. Fox and J.A. Shaw. Combination of multiple searches. In D.K. Harman, editor, *The Second Text REtrieval Conference (TREC-2)*, pages 243–252. National Institute for Standards and Technology. NIST Special Publication 500-215, 1994.

[6] D. Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, Center for Telematics and Information Technology, University of Twente, 2001.

[7] V. Hollink, J. Kamps, C. Monz, and M. de Rijke. Monolingual document retrieval for European languages. *Information Retrieval*, 6, 2003.

[8] J. Kamps, C. Monz, and M. de Rijke. Combining evidence for cross-language information retrieval. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Evaluation of Cross-Language Information Retrieval Systems, CLEF 2002*, Lecture Notes in Computer Science. Springer, 2003.

[9] J.H. Lee. Combining multiple evidence from different properties of weighting schemes. In E.A. Fox, P. Ingwersen, and R. Fidel, editors, *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 180–188. ACM Press, New York NY, USA, 1995.

[10] C. Monz and M. de Rijke. Shallow morphological analysis in monolingual information retrieval for Dutch, German and Italian. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Evaluation of Cross-Language Information Retrieval Systems, CLEF 2001*, volume 2406 of *Lecture Notes in Computer Science*, pages 262–277. Springer, 2002.

[11] R. Pohlmann and W. Kraaij. Improving the precision of a text retrieval system with compound analysis. In J. Landsbergen, J. Odijk, K. van Deemter, and G. Veldhuijzen van Zanten, editors, *Proceedings of the 7th Computational Linguistics in the Netherlands Meeting (CLIN 1996)*, pages 115–129, 1996.

[12] PROMT-Reverso. Online translator, 2003. http://translation2.paralink.com/.

[13] S.E. Robertson, S. Walker, and M. Beaulieu. Experimentation as a way of life: Okapi at TREC. *Information Processing & Management*, 36:95–108, 2000.

[14] J.J. Rocchio, Jr. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, Prentice-Hall Series in Automatic Computation, chapter 14, pages 313–323. Prentice-Hall, Englewood Cliffs NJ, 1971.

[15] Snowball. Stemming algorithms for use in information retrieval, 2003. http://www.snowball.tartarus.org/.

[16] Worldlingo. Online translator, 2003. http://www.worldlingo.com/.