Published in Lecture Notes in Computer Science 2785, 66-90, 2003
which should be used for any reference to this work

Report on CLEF 2002 Experiments: Combining Multiple Sources of Evidence

Jacques Savoy

Institut interfacultaire d'informatique
Université de Neuchâtel, Pierre-à-Mazel 7, 2000 Neuchâtel, Switzerland
jacques.savoy@unine.ch
http://www.unine.ch/info/clef/

Abstract. In our second participation in the CLEF retrieval tasks, our first objective was to propose better and more general stopword lists for various European languages (namely, French, Italian, German, Spanish and Finnish) along with improved, simpler and efficient stemming procedures. Our second goal was to propose a combined query-translation approach that could cross language barriers and also an effective merging strategy based on logistic regression for accessing the multilingual collection. Finally, within the Amaryllis experiment, we wanted to analyze how a specialized thesaurus might improve retrieval effectiveness.

1 Introduction

Taking our experiments of last year as a starting point [1], in CLEF 2002 we participated in the French, Italian, Spanish, German, Dutch and Finnish monolingual tasks, where our information retrieval approaches can work without having to rely on a dictionary. In Section 2, we describe how we improved our stopword lists and simple stemmers for the French, Italian, Spanish and German languages. For German, we also suggest a new decompounding algorithm. For Dutch, we used the available stoplist and stemmer, and for Finnish we designed a new stemmer and stopword list. In order to obtain a better overview of our results, we have evaluated our procedures using ten different retrieval schemes. In Section 3, we discuss how we chose to express the requests in English for the various bilingual tracks, and automatically translated them using five different machine translation (MT) systems and one bilingual dictionary. We study these various translations and, on the basis of the relative merit of each translation device, we investigate various combinations of them. In Section 4, we describe our multilingual information retrieval results, while investigating various merging strategies based on results obtained during our bilingual tasks. Finally, in the last section, we present various experiments carried out using the Amaryllis corpus, which included a specialized thesaurus that could be used to improve the retrieval effectiveness of the information retrieval system.

2 Monolingual Indexing and Search

Most European languages in the Indo-European language family (including French, Italian, Spanish, German and Dutch) can be viewed as flectional languages within which polymorph suffixes are added at the end of a flexed root. However, the Finnish language, a member of the Uralic language family (along with Turkish), is based on a concatenative morphology in which suffixes, more or less invariable, are added to roots that are generally invariable.

Any adaptation for the other languages in the CLEF experiments of indexing or search strategies prepared for English requires the development of stopword lists and fast stemming procedures. Stopword lists are composed of non-significant words that are removed from a document or a request before the indexing process begins. Stemming procedures try to remove inflectional and derivational suffixes in order to conflate word variants into the same stem or root.

This first section will deal with these issues and is organized as follows: Section 2.1 contains an overview of our eight test-collections while Section 2.2 describes our general approach to building stopword lists and stemmers for use with languages other than English. In order to decompound German words, we try a simple decompounding algorithm as described in Section 2.3. Section 2.4 describes the Okapi probabilistic model together with various vector-space models and we evaluate them using eight test-collections written in seven different languages (monolingual track).

2.1 Overview of the Test-Collections

The corpora used in our experiments included newspapers such as the *Los Angeles Times* (1994, English), *Le Monde* (1994, French), *La Stampa* (1994, Italian), *Der Spiegel* (1994/95, German), *Frankfurter Rundschau* (1994, German), *NRC Handelsbald* (1994/95, Dutch), *Algemeen Dagblad* (1995/95, Dutch), and *Tidningarnas Telegrambyrå* (1994/95, Finnish). As a second source of information, we also used various articles edited by news agencies including *EFE* (1994, Spanish) and the Swiss news agency (1994, available in French, German and Italian but without parallel translation). As shown in Tables 1 and 2, these corpora are of various sizes, with the English, German, Spanish and Dutch collections being twice the volume of the French, Italian and Finnish sources. On the other hand, the mean number of distinct indexing terms per document is relatively similar across the corpora (around 120), and this number is a little bit higher for the English collection (167.33). The Amaryllis collection contains abstracts of scientific papers written mainly in French and this corpus contains fewer distinct indexing terms per article (70.418).

An examination of the number of relevant documents per request as shown in Tables 1 and 2 reveals that the mean number is always greater than the median (e.g., the English collection contains an average of 19.548 relevant articles per query and the corresponding median is 11.5). These findings indicate that each collection contains numerous queries with a rather small number of relevant

Table 1. Test-collection statistics

	English	French	Italian	German	Spanish
Size (in MB)	425 MB	243 MB	278 MB	527 MB	509 MB
# of documents	113,005	87,191	108,578	225,371	215,738
# of distinct terms	330,753	320,526	503,550	1,507,806	528,753
Number of distinct indexing terms / document					
Mean	167.33	130.213	129.908	119.072	111.803
Median	138	95	92	89	99
Maximum	1,812	1,622	1,394	2,420	642
Minimum	2	3	1	1	5
Number of queries	42	50	49	50	50
Number of rel. items	821	1,383	1,072	1,938	2,854
Mean rel. items / request	19.548	27.66	21.878	38.76	57.08
Median	11.5	13.5	16	28	27
Maximum	96	177	86	119	321
Minimum	1	1	3	1	3

Table 2. Test-collection statistics

	Dutch	Finnish	Amaryllis
Size (in MB)	540 MB	137 MB	195 MB
# of documents	190,604	55,344	148,688
# of distinct terms	883,953	1,483,354	413,262
Number of distinct indexing terms / document			
Mean	110.013	114.01	70.418
Median	77	87	64
Maximum	2,297	1,946	263
Minimum	1	1	5
Number of queries	50	30	25
Number of rel. items	1,862	502	2,018
Mean rel. items / request	37.24	16.733	80.72
Median	21	8.5	67
Maximum	301	62	180
Minimum	4	1	18

items. For each collection, we encounter 50 queries except for the Italian corpus (for which Query #120 does not have any relevant items) and the English collection (for which Query #93, #96, #101, #110, #117, #118, #127 and #132 do not have any relevant items). The Finnish corpus contains only 30 available requests while only 25 queries are included in the Amaryllis collection.

For our automatic runs we retained only the following logical sections from the original documents during the indexing process: <TITLE>, <HEADLINE>, <TEXT>, <LEAD>, <LEAD1>, <TX>, <LD>, <TI>, and <ST>. On the other

hand, we did conduct two experiments (indicated as manual runs): one with the French collection and one with the German corpus, within which we retained the following tags: for the French collection: <DE>, <KW>, <TB>, <SUBJECTS>, <CHA1>, <NAMES>, <NOM1>, <NOTE>, <GENRE>, <ORT1>, <SU11>, <SU21>, <GO11>, <GO12>, <GO13>, <GO14>, <GO24>, <TI01>, <TI02>, <TI03>, <TI04>, <TI05>, <TI06>, <TI07>, <TI08>, <PEOPLE>, <TI09>, <SOT1>, <SYE1>, and <SYF1>; while for the German corpus and for one experiment, we also used the following tags: <KW>, and <TB>.

From the topic descriptions we automatically removed certain phrases such as "Relevant documents report ...", "Find documents that give ...", "Trouver des documents qui parlent ...", "Sono valide le discussioni e le decisioni ...", "Relevante Dokumente berichten ..." or "Los documentos relevantes proporcionan información ...".

To evaluate our approaches, we used the SMART system as a test bed for implementing the Okapi probabilistic model [2] as well as other vector-space models. This year our experiments were conducted on an Intel Pentium III/600 (memory: 1 GB, swap: 2 GB, disk: 6 x 35 GB).

2.2 Stopword Lists and Stemming Procedures

In order to define general stopword lists, we began with lists already available for the English and French languages [3], [4], while for the other languages we established a general stopword list by following the guidelines described in [3]. These lists generally contain the top 200 words most frequently used in the various collections, plus articles, pronouns, prepositions, conjunctions, and very frequently occurring verb forms (e.g., "to be", "is", "has", etc.). Stopword lists used during our previous participation [1] were often extended. For the English language for example we used that provided by the SMART system (571 words). For the other languages we used 431 words for Italian (no change from last year), 462 for French (previously 217), 603 for German (previously 294), 351 for Spanish (previously 272), 1,315 for Dutch (available at CLEF Web site) and 1,134 for Finnish (these stopword lists are available at `www.unine.ch/info/clef/`).

After removing high frequency words, an indexing procedure uses a stemming algorithm that attempts to conflate word variants into the same stem or root. In developing this procedure for the French, Italian, German and Spanish languages, it is important to remember that these languages have more complex morphologies than does the English language [5]. As a first approach, our intention was to remove only inflectional suffixes such that singular and plural word forms or feminine as well as masculine forms conflate to the same root. More sophisticated schemes have already been proposed for the removal of derivational suffixes (e.g., "-ize", "-ably", "-ship" in the English language), such as in the stemmer developed by Lovins [6], based on a list of over 260 suffixes, while that of Porter [7] looks for about 60 suffixes. Figuerola *et al.* [8] for example described two different stemmers for the Spanish language, and the results show that removing only inflectional suffixes (88 different inflectional suffixes were defined)

seemed to provide better retrieval levels than did removing both inflectional and derivational suffixes (this extended stemmer included 230 suffixes).

Our stemming procedures can also be found at `www.unine.ch/info/clef/`. This year we improved our stemming algorithms for French, and also removed some derivational suffixes were also removed. For the Dutch language, we use Kraaij & Pohlmann's stemmer(`ruulst.let.ruu.nl:2000/uplift/ulift.html`) [9]. For the Finnish language, our stemmer tries to conflate various word declinations into the same stem. Finnish makes a distinction between partial object(s) and whole object(s) (e.g., "syön leilää" for "I'm eating bread", "syön leivän" for "I'm eating a (whole) bread", or "syön leipiä" for "I'm eating breads", and "syön leivät" for "I'm eating the breads"). This aspect is not currently being taken into consideration.

Finally, diacritic characters are usually not present in English collections (with some exceptions, such as "à la carte" or "résumé"); such characters are replaced by their corresponding non-accentuated letter in the Italian, Dutch, Finnish, German and Spanish collections.

2.3 Decompounding German Words

Most European languages manifest other morphological characteristics for which our approach has made allowances, with compound word constructions being just one example (e.g., handgun, worldwide). In German, compound words are widely used and this causes more difficulties than in English. For example, a life insurance company employee would be "Lebensversicherungsgesellschaftsangestellter" (Leben + s + versicherung + s + gesellschaft + s + angestellter for life + insurance + company + employee). Also the morphological marker ("s") is not always present (e.g., "Bankangestelltenlohn" built as Bank + angestellten + lohn (salary)). In Finnish, we also encounter similar constructions such as "rakkauskirje" (rakkaus + kirje for love + letter) or "työviikko" (työ + viikko for work + week).

According to Monz & de Rijke [10] or Chen [11], including both compounds and their composite parts in queries and documents can result in better performance while according to Molina-Salgado et al. [12], the decomposition of German words seems to reduce average precision.

In our approach we break up any words having an initial length greater than or equal to eight characters. Moreover, decomposition cannot take place before an initial sequence `[V]C`, meaning that a word might begin with a series of vowels that must be followed by at least one consonant. The algorithm then seeks occurrences of one of the models described in Table 3. For example, the last model "gss g s" indicates that when we encounter the character string "gss" the computer is allowed to cut the compound term, ending the first word with "g" and beginning the second with "s". All the models shown in Table 3 can include letter sequences that are impossible to find in a simple German word such as "dtt", "fff", or "ldm". Once it has detected this pattern, the computer makes sure that the corresponding part consists of at least four characters, potentially beginning with a series of vowels (criterion noted as `[V]`), followed by a `CV`

Table 3. Decompounding patterns for German

String sequence			End of previous word			Beginning of next word					
schaften	schaft	.	tion	tion	.	ern	er	.	schg	sch	g
weisen	weise	.	ling	ling	.	tät	tät	.	schl	sch	l
lischen	lisch	.	igkeit	igkeit	.	net	net	.	schh	sch	h
lingen	ling	.	lichkeit	lichkeit	.	ens	en	.	scht	sch	t
igkeiten	igkeit	.	keit	keit	.	ers	er	.	dtt	dt	t
lichkeit	lichkeit	.	erheit	erheit	.	ems	em	.	dtp	dt	p
keiten	keit	.	enheit	enheit	.	ts	t	.	dtm	dt	m
erheiten	erheit	.	heit	heit	.	ions	ion	.	dtb	dt	b
enheiten	enheit	.	lein	lein	.	isch	isch	.	dtw	dt	w
heiten	heit	.	chen	chen	.	rm	rm	.	ldan	ld	an
haften	haft	.	haft	haft	.	rw	rw	.	ldg	ld	g
halben	halb	.	halb	halb	.	nbr	n	br	ldm	ld	m
langen	lang	.	lang	lang	.	nb	n	b	ldq	ld	q
erlichen	erlich	.	erlich	erlich	.	nfl	n	fl	ldp	ld	p
enlichen	enlich	.	enlich	enlich	.	nfr	n	fr	ldv	ld	v
lichen	lich	.	lich	lich	.	nf	n	f	ldw	ld	w
baren	bar	.	bar	bar	.	nh	n	h	tst	t	t
igenden	igend	.	igend	igend	.	nk	n	k	rg	r	g
igungen	igung	.	igung	igung	.	ntr	n	tr	rk	r	k
igen	ig	.	ig	ig	.	fff	ff	f	rm	r	m
enden	end	.	end	end	.	ffs	ff	.	rr	r	r
isten	ist	.	ist	ist	.	fk	f	k	rs	r	s
anten	ant	.	ant	ant	.	fm	f	m	rt	r	t
ungen	ung	.	tum	tum	.	fp	f	p	rw	r	w
schaft	schaft	.	age	age	.	fv	f	v	rz	r	z
weise	weise	.	ung	ung	.	fw	f	w	fp	f	p
lisch	lisch	.	enden	end	.	schb	sch	b	fsf	f	f
ismus	ismus	.	eren	er	.	schf	sch	f	gss	g	s

sequence. If decomposition does prove to be possible, the algorithm then begins working on the right part of the decomposed word.

As an example, the compound word "Betreuungsstelle" (meaning "care center" is made up of "Betreuung" (care) and "Stelle" (center, place)). This word is definitely more than seven characters long. Once this has been verified, the computer begins searching for substitution models starting with the second character. The computer will find a match with the last model described in Table 3, and thus form the words "Betreuung" and "Stelle." This break is validated because the second word has a length greater than four characters. This term also meets criterion `[V]CV` and finally, given that the term "Stelle" has less than eight letters, the computer will not attempt to continue decomposing this term. Our approach for decompounding German words is based on the linguistic rules used to build German compounds. As an alternative, we could decompound German words using a list of German words which may then be used to generate all pos-

sible ways to break up a compound and then select the decomposition containing the minimum number of component words as suggested by Chen [13].

2.4 Indexing and Searching Strategy

In order to obtain a broader view of the relative merits of various retrieval models, we first adopted a binary indexing scheme within which each document (or request) was represented by a set of keywords, without any weight. To measure the similarity between documents and requests, we counted the number of common terms, computed according to the inner product (retrieval model denoted "doc=bnn, query=bnn" or "bnn-bnn"). For document and query indexing, binary logical restrictions however are often too limiting. In order to weight the presence of each indexing term in a document surrogate (or in a query), we can take term occurrence frequency (denoted tf) into account, resulting in better term distinction and increasing our indexing flexibility (retrieval model notation: "doc=nnn, query=nnn" or "nnn-nnn").

Those terms that do however occur very frequently in the collection are not considered very helpful in distinguishing between relevant and non-relevant items. Thus we might count their frequency in the collection (denoted df), or more precisely the inverse document frequency (denoted by idf=ln(n/df)), resulting in more weight for sparse words and less weight for more frequent ones. Moreover, a cosine normalization can prove beneficial and each indexing weight

Table 4. Weighting schemes

bnn	$w_{ij} = 1$	npn	$w_{ij} = tf_{ij} \cdot ln\left[\frac{n - df_j}{df_j}\right]$
nnn	$w_{ij} = tf_{ij}$		
ntc	$w_{ij} = \frac{tf_{ij} \cdot idf_j}{\sqrt{\sum_{k=1}^{t}(tf_{ik} \cdot idf_k)^2}}$	atn	$w_{ij} = idf_j \cdot \left[\frac{0.5 + 0.5 \cdot tf_{ij}}{max\ tf_{i\cdot}}\right]$
lnc	$w_{ij} = \frac{ln(tf_{ij})+1}{\sqrt{\sum_{k=1}^{t}(ln(tf_{ik})+1)^2}}$	ltn	$w_{ij} = (ln(tf_{ij}) + 1) \cdot idf_j$
Okapi	$w_{ij} = \frac{(k_1+1) \cdot tf_{ij}}{K + tf_{ij}}$ with $K = k_1 \cdot \left[(1-b) + b \cdot \frac{l_i}{avdl}\right]$		
ltc	$w_{ij} = \frac{(ln(tf_{ij})+1) \cdot idf_j}{\sqrt{\sum_{k=1}^{t}[(ln(tf_{ik})+1) \cdot idf_k]^2}}$		
dtu	$w_{ij} = \frac{(ln(ln(tf_{ij})+1)+1) \cdot idf_j}{(1-slope) \cdot pivot + (slope \cdot nt_i)}$		
dtc	$w_{ij} = \frac{(ln(ln(tf_{ij})+1)+1) \cdot idf_j}{\sqrt{\sum_{k=1}^{t}[(ln(ln(tf_{ik})+1)+1) \cdot idf_k]^2}}$		
Lnu	$w_{ij} = \frac{\frac{ln(tf_{ij})+1}{ln\left(\frac{l_i}{nt_i}\right)+1}}{(1-slope) \cdot pivot + (slope \cdot nt_i)}$		

could vary within the range of 0 to 1 (retrieval model denoted "ntc-ntc"). Table 4 shows the exact weighting formulation w_{ij} for each indexing term T_j in a document D_i in which n indicates the number of documents D_i in the collection, nt_i the number of unique indexing terms in D_i, and l_i the sum of tf_{ij} for a given document D_i.

Other variants of this formula can also be created, especially if we determine the occurrence of a given term in a document to be a rare event. Thus, it may be a good practice to give more importance to the first occurrence of this word as compared to any successive or repeating occurrences. Therefore, the tf component may be computed as 0.5 + 0.5·[tf / max tf in a document] (retrieval model denoted "doc=atn").

Finally, we consider that a term's presence in a shorter document provides stronger evidence than it does in a longer document. To account for this, we integrated document length within the weighting formula, leading to more complex IR models; for example, the IR model denoted by "doc=Lnu" [14], "doc=dtu" [15]. Finally for CLEF 2002, we also conducted various experiments using the Okapi probabilistic model [2] within which K = k_1·[(1-b) + b·(l_i/avdl)], representing the ratio between the length of D_i measured by l_i (sum of tf_{ij}) and the collection mean noted by avdl.

In our experiments, the constants b, k_1, avdl, pivot and slope are fixed according to the values listed in Table 5. To evaluate the retrieval performance of these various IR models, we adopted the non-interpolated average precision technique (computed on the basis of 1,000 retrieved items per request by the TREC-EVAL program [16]), providing both precision and recall with the use of a single number. Brand & Brünner [17] have evaluated in more detail the retrieval effectiveness achieved when modifying the values of these parameters.

Given that French, Italian and Spanish morphology is comparable to that of English, we decided to index French, Italian and Spanish documents based on word stems. For the German, Dutch and Finnish languages and their more complex compounding morphology, we decided to use a 5-gram approach [18], [19]. However, contrary to [19], our generation of 5-gram indexing terms does not span word boundaries. This value of 5 was chosen because it performed better

Table 5. Parameter setting for the various test-collections

Language	b	k_1	avdl	pivot	slope
English	0.8	2	900	100	0.1
French	0.7	2	750	100	0.1
Italian	0.6	1.5	800	100	0.1
Spanish	0.5	1.2	300	100	0.1
German	0.55	1.5	600	125	0.1
Dutch	0.9	3.0	600	125	0.1
Finnish	0.75	1.2	900	125	0.1
Amaryllis	0.7	2	160	30	0.2

with the CLEF 2000 corpora [20]. Using this indexing scheme, the compound "das Hausdach" (the roof of the house) will generate the following indexing terms: "das", "hausd", "ausda", "usdac" and "sdach".

Our evaluation results as reported in Tables 6 and 7 show that the Okapi probabilistic model performs best for the five different languages. In the second position, we usually find the vector-space model "doc=Lnu, query=ltc" and in the third "doc=dtu, query=dtc". Finally, the traditional tf-idf weighting scheme ("doc=ntc, query=ntc") does not exhibit very satisfactory results, and the simple term-frequency weighting scheme ("doc=nnn, query=nnn") or the simple coordinate match ("doc=bnn, query=bnn") results in poor retrieval performance. However, Amati et al. [21] indicate that the PROSIT probabilistic model may result in better performance than the Okapi approach, at least for the Italian collection.

For the German language, we assumed that the 5-gram indexing, decompounded indexing and word-based document representations are distinct and independent sources of evidence about document content. We therefore decided to combine these three indexing schemes and to do so we normalized similarity values obtained from each of these three separate retrieval models, as shown in Equation 1 (see Section 4). The resulting average precision for these four approaches is shown in Table 7, thus demonstrating how the combined model usually results in better retrieval performance.

It has been observed that pseudo-relevance feedback (blind-query expansion) seems to be a useful technique for enhancing retrieval effectiveness. In this study, we adopted Rocchio's approach [14] with $\alpha = 0.75$, $\beta = 0.75$ whereby the system was allowed to add m terms extracted from the k best-ranked documents from the original query. To evaluate this proposition, we used the Okapi probabilistic model and enlarged the query by 10 to 20 terms (or until 300 within the 5-gram

Table 6. Average precision of various indexing and searching strategies (monolingual)

	Average precision			
Query TD Model	English 42 queries	French 50 queries	Italian 49 queries	Spanish 50 queries
doc=Okapi, query=npn	**50.08**	**48.41**	**41.05**	**51.71**
doc=Lnu, query=ltc	48.91	46.97	39.93	49.27
doc=dtu, query=dtc	43.03	45.38	39.53	47.29
doc=atn, query=ntc	42.50	42.42	39.08	46.01
doc=ltn, query=ntc	39.69	44.19	37.03	46.90
doc=ntc, query=ntc	27.47	31.41	29.32	33.05
doc=ltc, query=ltc	28.43	32.94	31.78	36.61
doc=lnc, query=ltc	29.89	33.49	32.79	38.78
doc=bnn, query=bnn	19.61	18.59	18.53	25.12
doc=nnn, query=nnn	9.59	14.97	15.63	22.22

Table 7. Average precision of various indexing and searching strategies (German)

	Average precision			
Query TD Model	German words 50 queries	German decompounded 50 queries	German 5-gram 50 queries	German combined 50 queries
doc=Okapi, query=npn	**37.39**	**37.75**	**39.83**	**41.25**
doc=Lnu, query=ltc	36.41	36.77	36.91	39.79
doc=dtu, query=dtc	35.55	35.08	36.03	38.21
doc=atn, query=ntc	34.48	33.46	37.90	37.93
doc=ltn, query=ntc	34.68	33.67	34.79	36.37
doc=ntc, query=ntc	29.57	31.16	32.52	32.88
doc=ltc, query=ltc	28.69	29.26	30.05	31.08
doc=lnc, query=ltc	29.33	29.14	29.95	31.24
doc=bnn, query=bnn	17.65	16.88	16.91	21.30
doc=nnn, query=nnn	14.87	12.52	8.94	13.49

model, as shown in Table 9) found in the 5 or 10 best-retrieved articles. The results shown in Tables 8 and 9 indicate that the optimal parameter setting seems to be collection dependent. Moreover, performance improvement also seems to be collection dependent (or language dependent). While no improvement was shown for the English corpus, there was an increase of 8.55% for the Spanish corpus (from an average precision of 51.71 to 56.13), 9.85% for the French corpus (from 48.41 to 53.18), 12.91% for the Italian language (41.05 to 46.35) and 13.26% for the German collection (from 41.25 to 46.72, combined model, Table 9).

This year, we also participated in the Dutch and Finnish monolingual tasks, the results of which are given in Table 10, while the average precision obtained using the Okapi model for blind-query expansion is given in Table 11. For these two languages, we also applied our combined indexing model based on the 5-

Table 8. Average precision using blind-query expansion

	Average precision			
Query TD Model	English 42 queries	French 50 queries	Italian 49 queries	Spanish 50 queries
doc=Okapi, query=npn	**50.08**	48.41	41.05	51.71
5 docs / 10 best terms	49.54	53.10	45.14	55.16
5 docs / 15 best terms	48.68	**53.18**	46.07	54.95
5 docs / 20 best terms	48.62	53.13	**46.35**	54.41
10 docs / 10 best terms	47.77	52.03	45.37	55.94
10 docs / 15 best terms	46.92	52.75	46.18	56.00
10 docs / 20 best terms	47.42	52.78	45.87	**56.13**

Table 9. Average precision using blind-query expansion (German corpus)

	Average precision			
Query TD Model	German words 50 queries	German decompounded 50 queries	German 5-gram 50 queries	German combined 50 queries
doc=Okapi, query=npn	37.39	37.75	39.83	41.25
# docs / # terms	5/40 42.90	5/40 42.19	10/200 45.45	**46.72**
# docs / # terms	5/40 **42.90**	5/40 **42.19**	5/300 **45.82**	46.27

gram and word-based document representations. While for the Dutch language, our combined model seems to enhance the retrieval effectiveness, for the Finnish language it does not. This however was a first trial for our proposed Finnish stemmer and this solution seemed to improve average precision over a baseline run without a stemming procedure (Okapi model, unstemmed 23.04, with stemming 30.45, an improvement of +32.16%).

In the monolingual track, we submitted ten runs along with their corresponding descriptions, as listed in Table 12. Seven of them were fully automatic using the request's Title and Description logical sections, while the last three were based on the topic's Title, Description and Narrative sections. In these last three runs, two were labeled "manual" because we used logical sections containing manually assigned index terms. For all runs, however, we did not use any "real" manual intervention during the indexing and retrieval procedures.

Table 10. Average precision for the Dutch and Finnish corpora

	Average precision					
Query TD Model	Dutch words 50 queries	Dutch 5-gram 50 queries	Dutch combined 50 queries	Finnish words 30 queries	Finnish 5-gram 30 queries	Finnish combined 30 queries
Okapi–npn	**42.37**	**41.75**	**44.56**	**30.45**	**38.25**	**37.51**
Lnu–ltc	42.57	40.73	44.50	27.58	36.07	36.83
dtu–dtc	41.26	40.59	43.00	30.70	36.79	36.47
atn–ntc	40.29	40.34	41.89	29.22	37.26	36.51
ltn–ntc	38.33	38.72	40.24	29.14	35.28	35.31
ntc–ntc	33.35	34.94	36.41	25.21	30.68	31.93
ltc–ltc	32.81	31.24	34.46	26.53	30.85	33.47
lnc–ltc	31.91	29.67	34.18	24.86	30.43	31.39
bnn–bnn	18.91	20.87	23.52	12.46	14.55	18.64
nnn–nnn	13.75	10.48	12.86	11.43	14.69	15.56

Table 11. Average precision using blind-query expansion

	Average precision		
Query TD Model	Dutch words 50 queries	Dutch 5-gram 50 queries	Dutch combined 50 queries
doc=Okapi, query=npn	42.37	41.75	44.56
# docs/# terms	5/60 47.86	5/75 45.09	48.78
# docs/ # terms	5/100 **48.84**	10/150 **46.29**	**49.28**
	Finnish words 30 queries	Finnish 5-gram 30 queries	Finnish combined 30 queries
doc=Okapi, query=npn	30.45	38.25	37.51
# docs/# terms	5/60 31.89	5/75 40.90	39.33
# docs/ # terms	5/15 **32.36**	5/175 **41.67**	**40.11**

Table 12. Official monolingual run descriptions

Run name	Lang.	Query	Form	Model	Query expansion	Precision
UniNEfr	FR	TD	auto	Okapi	no expansion	48.41
UniNEit	IT	TD	auto	Okapi	10 best docs / 15 terms	46.18
UniNEes	SP	TD	auto	Okapi	5 best docs / 20 terms	54.41
UniNEde	DE	TD	auto	comb	5/40 word, 10/200 5-gram	46.72
UniNEnl	NL	TD	auto	comb	5/60 word, 5/75 5-gram	48.78
UniNEfi1	FI	TD	auto	Okapi	5 best docs / 75 terms	40.90
UniNEfi2	FI	TD	auto	comb	5/60 word, 5/75 5-gram	39.33
UniNEfrtdn	FR	TDN	man	Okapi	5 best docs / 10 terms	59.19
UniNEestdn	SP	TDN	auto	Okapi	5 best docs / 40 terms	60.51
UniNEdetdn	DE	TDN	man	comb	5/50 word, 10/300 5-gram	49.11

3 Bilingual Information Retrieval

In order to overcome language barriers, we based our approach on free and readily available translation resources that automatically translate queries into the desired target language. More precisely, the original queries were written in English and we used no parallel or aligned corpora to derive statistically [22] or semantically related words in the target language. Section 3.1 describes our combined strategy for cross-lingual retrieval while Section 3.2 provides some examples of translation errors.

This year, we used five machine translation systems, namely

1. SYSTRAN^{TM} [23] (`babel.altavista.com/translate.dyn`),
2. GOOGLE.COM (`www.google.com/language_tools`),
3. FREETRANSLATION.COM (`www.freetranslation.com`),
4. INTERTRAN (`www.tranexp.com:2000/InterTran`),
5. REVERSO ONLINE (`translation2.paralink.com`).

As a bilingual dictionary we used the BABYLON^{TM} system (`www.babylon.com`).

3.1 Automatic Query Translation

In order to develop a fully automatic approach, we chose to translate the queries using five different machine translation (MT) systems. We also translated query terms word-by-word using the BABYLON bilingual dictionary, which provides not only one but several terms as the translation of each word submitted. In our experiments, we decided to pick the first translation given (labeled "baby1"), the first two terms (labeled "baby2") or the first three available translations (labeled "baby3").

The first part of Table 13 lists the average precision for each translation device used along with the performance achieved by manually translated queries. For German, we also reported the retrieval effectiveness achieved by the three different approaches, namely using words as indexing units, decompounding the German words according to our approach and the 5-gram model. While the REVERSO system seems to be the best choice for German and Spanish, FREETRANSLATION is the best choice for Italian and BABYLON 1 the best for French.

In order to improve search performance, we tried combining different machine translation systems with the bilingual dictionary approach. In this case, we formed the translated query by concatenating the different translations provided by the various approaches. Thus in the line entitled "Comb 1" we combined one machine translation system with the bilingual dictionary ("baby1"). Similarly, in lines "Comb 2" and "Comb 2b", we listed the results of two machine translation approaches, and in lines "Comb 3", "Comb 3b" and "Comb 3b2" the three machine translation systems. With the exception of the performance under "Comb 3b2", we also included terms provided by the "baby1" dictionary look-up in the translated queries. In columns "MT 2" and "MT 3", we evaluated the combination of two or three, respectively, machine translation systems. Finally, we also combined all translation sources (under the heading "All") and all machine translation approaches under the heading "MT all".

Since the performance of each translation device depends on the target language, in the lower part of Table 13 we included the exact specification for each of the combined runs. For German, for each of the three indexing models, we used the same combination of translation resources. From an examination of the retrieval effectiveness of our various combined approaches listed in the middle part of Table 13, a clear recommendation cannot be made. Overall, it seems better to combine two or three machine translation systems with the bilingual dictionary approach ("baby1"). However, combining the five machine translation systems (heading "MT all") or all translation tools (heading "All") did not result in very satisfactory performance.

Table 14 lists the exact specifics of our various bilingual runs. However, when submitting our official results, we used the wrong numbers for Query #130 and #131 (we switched these two query numbers). Thus, both queries have an average precision 0.00 in our official results and we report the corrected performance in Tables 14 and 17 (multilingual runs).

3.2 Examples of Translation Failures

In order to obtain a preliminary picture of the difficulties underlying the automatic translation approach, we analyzed some queries by comparing translations produced by our six machine-based tools with query formulations written by humans (examples are given in Table 15). As a first example, the title of Query #113 is "European Cup". In this case, the term "cup" was analyzed as teacup by all automatic translation tools, resulting in the French translations

Table 13. Average precision of various query translation strategies (Okapi model)

	Average precision					
Query TD Device	French	Italian	Spanish	German word	German decomp.	German 5-gram
Original	48.41	41.05	51.71	37.39	37.75	39.83
Systran	42.70	32.30	38.49	28.75	28.66	27.74
Google	42.70	32.30	38.35	28.07	26.05	27.19
FreeTrans	40.58	**32.71**	40.55	28.85	**31.42**	27.47
InterTran	33.89	30.28	37.36	21.32	21.61	19.21
Reverso	39.02	N/A	**43.28**	**30.71**	30.33	**28.71**
Babylon 1	**43.24**	27.65	39.62	26.17	27.66	28.10
Babylon 2	37.58	23.92	34.82	26.78	27.74	25.41
Babylon 3	35.69	21.65	32.89	25.34	26.03	23.66
Comb 1	46.77	33.31	44.57	34.32	34.66	32.75
Comb 2	48.02	34.70	**45.63**	35.26	34.92	32.95
Comb 2b	48.02		45.53	35.09	34.51	32.76
Comb 3	**48.56**	34.98	45.34	34.43	34.37	**33.34**
Comb 3b	48.49	35.02	45.34	34.58	34.43	32.76
Comb 3b2				**35.41**	**35.13**	33.25
MT 2		**35.82**				
MT 3	44.54	35.57	44.32	33.53	33.05	31.96
All	47.94	35.29	44.25	34.52	34.31	32.79
MT all	46.83	35.68	44.25	33.80	33.51	31.66
Comb 1	Rever-baby1	Free-baby1	Rever-baby1	Reverso-baby1		
Comb 2	Reverso systran-baby1	Free-google baby1	Rever-systran baby1	Reverso-systran-baby1		
Comb 2b	Reverso google-baby1		Rever-google baby1	Reverso-google-baby1		
Comb 3	Reverso-free google-baby1	Free-google inter-baby1	Free-google rever-baby1	Reverso-systran-inter-baby1		
Comb 3b	Reverso-inter google-baby1	Free-google systr-baby1	Free-google rever-baby2	Reverso-google-inter-baby1		
Comb 3b2				Reverso-systran-inter-baby2		
MT 2		Free-google				
MT 3	Reverso systr-google	Free-google inter	Free-google reverso	Reverso-inter-systran		

Table 14. Average precision and description of our official bilingual runs (Okapi model)

	Average precision				
Query TD	French	French	French	Italian	Italian
	UniNEfrBi	UniNEfrBi2	UniNEfrBi3	UniNEitBi	UniNEitBi2
Combined	Comb 3b	MTall+baby2	MT all	Comb 2	Comb 3
#doc/#ter	5 / 20	5 / 40	10 / 15	10 / 60	10 / 100
Corrected	**51.64**	50.79	48.49	38.50	**38.62**
Official	49.35	48.47	46.20	37.36	37.56
Query TD	Spanish	Spanish	Spanish	German	German
	UniNEesBi	UniNEesBi2	UniNEesBi3	UniNEdeBi	UniNEdeBi2
Combined	MT 3	Comb 3b	Comb 2	Comb 3b2	Comb 3
#doc/#ter	10 / 75	10 / 100	10 / 75	5 / 100	5 / 300
Corrected	50.67	**50.95**	50.93	**42.89**	42.11
Official	47.63	47.86	47.84	41.29	40.42

"tasse" or "verre" (or "tazza" in Italian, "Schale" in German ("Pokal" can be viewed as a correct translation alternative) and "taza" or "Jícara" (small teacup) in Spanish).

In Query #118 ("Finland's first EU Commissioner"), the machine translation systems failed to provide the appropriate Spanish term "comisario" for "Commissioner" but returned "comisión" (commission) or "Comisionado" (adjective relative to commission). For this same topic number, the manually translated query seemed to contain a spelling error in Italian ("commi**s**ario" instead of "commi**ss**ario"). For the same topic, the translation provided in German "Beauftragter" (delegate) does not correspond to the appropriate term "Kommissar" (and "-" is missing in the translation "EUBEAUFTRAGTER").

Other examples: for Query #94 ("Return of Solzhenitsyn") which is translated manually in German ("Rückkehr Solschenizyns"), our automatic translation systems fail to translate the proper noun (returning "Solzhenitsyn" instead of "Solschenizyns"). Query #109 ("Computer Security") is translated manually in Spanish as "Seguridad Informática" and our various translation devices return different terms for "Computer" (e.g., "Computadora", "Computador" or "ordenador") but not the more appropriate term "Informática".

4 Multilingual Information Retrieval

Using our combined approach to automatically translate a query, we were able to search target document collections with queries written in a different language. This stage however represents only the first step in the development of multilanguage information retrieval systems. We also need to investigate the situation where users write a query in English in order to retrieve pertinent documents in English, French, Italian, German and Spanish, for example. To deal with this

Table 15. Examples of unsuccessful query translations

C113 (query translations failed in French, Italian, German and Spanish)
<EN-TITLE> European Cup
<FR-TITLE manually translated> Coupe d'Europe de football
<FR-TITLE FREETRANSLATION> Tasse européenne
<FR-TITLE BABYLON 1> Européen verre
<FR-TITLE BABYLON 2> Européen résident de verre tasse
<FR-TITLE BABYLON 3> Européen résident de l'Europe verre tasse coupe
<IT-TITLE manually translated> Campionati europei
<IT-TITLE SYSTRAN> Tazza Europea
<IT-TITLE GOOGLE> Tazza Europea
<GE-TITLE manually translated> Fussballeuropameisterschaft
<GE-TITLE SYSTRAN> Europäische Schale
<GE-TITLE REVERSO> Europäischer Pokal
<SP-TITLE manually translated> Eurocopa
<SP-TITLE INTERTRAN> Europea Jícara
<SP-TITLE REVERSO> Taza europea
C118 (query translations failed in Italian, German and Spanish)
<EN-TITLE> Finland's first EU Commissioner
<IT-TITLE manually translated> Primo commisario europeo per la Finlandia
<IT-TITLE GOOGLE> Primo commissario dell'Eu della Finlandia
<IT-TITLE FREETRANSLATION> Finlandia primo Commissario di EU
<GE-TITLE manually translated> Erster EU-Kommissar aus Finnland
<GE-TITLE GOOGLE> Finnlands erster EUBEAUFTRAGTER
<GE-TITLE REVERSO> Finlands erster EG-Beauftragter
<SP-TITLE manually translated> Primer comisario finlandés de la UE
<SP-TITLE GOOGLE> Primera comisión del EU de Finlandia
<SP-TITLE REVERSO> El primer Comisionado de Unión Europea de Finlandia

multi-language barrier, we divided our document sources according to language and thus formed five different collections. After searching in these corpora and obtaining five result lists, we needed to merge them in order to provide users with a single list of retrieved articles.

Recent literature has suggested various solutions to merging separate result lists obtained from different collections or distributed information services. As a first approach, we will assume that each collection contains approximately the same number of pertinent items and that the distribution of the relevant documents is similar across the results lists. Based solely on the rank of the retrieved records, we can interleave the results in a round-robin fashion. According to previous studies [24], the retrieval effectiveness of such an interleaving scheme is around 40% below that achieved from a single retrieval scheme working with a single huge collection, representing the entire set of documents.

To account for the document score computed for each retrieved item (or the similarity value between the retrieved record D_i and the query, denoted rsv_i), we can formulate the hypothesis that each collection be searched by the same or a very similar search engine, and that the similarity values would be therefore directly comparable [25]. Such a strategy, called raw-score merging, produces a final list sorted by the document score computed by each collection. However, collection-dependent statistics in document or query weights may vary widely among collections, and therefore this phenomenon may invalidate the raw-score merging hypothesis [26].

To account for this fact, we could normalize the document scores within each collection by dividing them by the maximum score (i.e. the document score of the retrieved record in the first position). As a variant of this normalized score merging scheme, Powell *et al.* [27] suggested normalizing the document score rsv_i according to the following formula:

$$rsv'_i \;=\; (rsv_i \;-\; rsv_{min}) \;/\; (rsv_{max} \;-\; rsv_{min}) \qquad (1)$$

in which rsv_i is the original retrieval status value (or document score), and rsv_{max} and rsv_{min} are the maximum and minimum document score values that a collection could achieve for the current query. In this study, the rsv_{max} is provided by the document score achieved by the first retrieved item and the retrieval status value obtained by the 1,000th retrieved record provides the value of rsv_{min}.

As a fourth strategy, we could use the logistic regression [28], [29] to predict the probability of a binary outcome variable, according to a set of explanatory variables. Based on this statistical approach, Le Calvé and Savoy [30] and Savoy [20] described how to predict the relevance probability for those documents retrieved by different retrieval schemes or collections. The resulting estimated probabilities are dependent on both the original document score rsv_i and the logarithm of the $rank_i$ attributed to the corresponding document D_i (see Equation 2). Based on these estimated relevance probabilities, we sort the records retrieved from separate collections in order to obtain a single ranked list. However, in order to estimate the underlying parameters, this approach requires a training set, which in this case was the CLEF 2001 topics and their relevance assessments.

$$Prob\,[\; D_i \; is \; rel \mid rank_i,\; rsv_i \;] \;=\; \frac{e^{\alpha \;+\; \beta_1 \;\cdot\; ln(rank_i) \;+\; \beta_2 \;\cdot\; rsv_i}}{1 + e^{\alpha \;+\; \beta_1 \;\cdot\; ln(rank_i) \;+\; \beta_2 \;\cdot\; rsv_i}} \qquad (2)$$

where $rank_i$ denotes the rank of the retrieved document D_i, ln() is the natural logarithm, and rsv_i is the retrieval status value (or document score) of the document D_i. In this equation, the coefficients α, β_1 and β_2 are unknown parameters that are estimated according the maximum likelihood method (the required computations were programmed using the S system [31]).

When searching multi-lingual corpora using Okapi, both the round-robin and the raw-score merging strategies provide very similar retrieval performance results (see Table 16). Normalized score merging based on Equation 1 provides an

Table 16. Average precision using various merging strategies based on automatically translated queries

	Average precision				
Query TD	English 42 queries 50.08	French 50 queries UniNEfrBi 51.64	Italian 49 queries UniNEitBi 38.50	Spanish 50 queries UniNEesBi 50.67	German 50 queries UniNEdeBi 42.89
Multiling. 50 queries	Round-robin 34.27	Raw-score 33.83	Eq. 1 36.62	Log $\ln(rank_i)$ 36.10	Log reg Eq. 2 **39.49**
	English 42 queries 50.08	French 50 queries UniNEfrBi2 50.79	Italian 49 queries UniNEitBi2 38.62	Spanish 50 queries UniNEesBi2 50.95	German 50 queries UniNEdeBi2 42.11
Multiling. 50 queries	Round-robin 33.97	Raw-score 33.99	Eq. 1 36.90	Log $\ln(rank_i)$ 35.59	Log reg Eq. 2 39.25

enhancement over the round-robin approach (36.62 vs. 34.27, an improvement of +6.86% in our first experiment, and 36.90 vs. 33.97, +8.63% in our second run). Using our logistic model with only the rank as explanatory variable (or more precisely the $\ln(rank_i)$, performance shown under the heading "Log $\ln(rank_i)$"), the resulting average precision was lower than the normalized score merging. The best average precision was achieved by merging the result lists based on the logistic regression approach (using both the rank and the document score as explanatory variables).

Our official and corrected results are shown in Table 17 while some statistics showing the number of documents provided by each collection are given in Table 18. From these data, we can see that the normalized score merging (UniNEm1) extracts more documents for the English corpus (a mean of 24.94 items) than does the logistic regression model (UniNEm2 where a mean of 11.44 documents results from the English collection). Moreover, the logistic regression scheme retrieves more documents from the Spanish and German collections. Finally, we can see that the percentage of relevant items per collection (or language) is relatively similar when comparing CLEF 2001 and CLEF 2002 test-collections.

Table 17. Average precision obtained with our official multilingual runs

Query TD	UniNEm1 Equation 1	UniNEm2 Log reg Eq. 2	UniNEm3 Equation 1	UniNEm4 Log reg Eq. 2	UniNEm5 Equation 1
Corrected	36.62	39.49	36.90	39.25	35.97
Official	34.88	37.83	35.12	37.56	35.52

5 Amaryllis Experiments

For the Amaryllis experiments, we wanted to determine whether a specialized thesaurus might improve the retrieval effectiveness over a baseline, ignoring term relationships. From the original documents and during the indexing process, we retained the following logical sections in our runs: <TEXT>, <TI>, <AB>, <MC>, and <KW>.

From the given thesaurus, we extracted 126,902 terms having a relationship with one or more terms (the thesaurus contains 173,946 entries delimited by the tags <RECORD>...</RECORD>, however only 149,207 entries had at least one relationship with another term. From these 149,207 entries, we found 22,305 multiple entries such as, for example, the term "Poste de travail" or "Bureau poste", see Table 19. In such cases, we stored only the last entry). In building our thesaurus, we removed the accents, wrote all terms in lowercase, and ignored numbers and terms given between parenthesis. For example, the word "poste" appears in 49 records (usually as part of a compound entry in the <TERMFR> field).

From our 126,902 entries, we counted 107,038 TRADENG (English translation) relationships, 14,590 SYNOFRE1 (synonym), 26,772 AUTOP1 relationships and 1,071 VAUSSI1 (See also) relationships (see examples given in Table 19). In a first set of experiments, we did not use this thesaurus and we used the Title and Description logical sections of the topics (second column of Table 20) or the Title, Description and Narrative parts of the queries (last column of Table 20). In a second set of experiments, we included all related words that could be found in the thesaurus using only the search keywords (average precision shown under the label "Qthes"). In a third experiment, we enlarged only document

Table 18. Statistics about the merging schemes based on the top 100 retrieved documents for each query

Statistics \ Language	English	French	Italian	Spanish	German
UniNEm1, based on the top 100 retrieved documents for each query					
Mean	24.94	16.68	19.12	23.8	15.46
Median	23.5	15	18	22	15
Maximum	60	54	45	70	54
Minimum	4	5	5	6	2
Standard deviation	13.14	9.26	9.17	14.15	9.79
UniNEm2, based on the top 100 retrieved documents for each query					
Mean	11.44	15.58	16.18	34.3	22.5
Median	9	14	16	34.5	19
Maximum	33	38	28	62	59
Minimum	1	6	8	10	4
Standard deviation	6.71	7.49	5.18	10.90	11.90
% relevant items CLEF02	10.18%	17.14%	13.29%	35.37%	24.02%
% relevant items CLEF01	10.52%	14.89%	15.31%	33.10%	26.17%

Table 19. Sample of various entries under the word "poste" in the Amaryllis thesaurus

<RECORD> <TERMFR> Analyse de poste <TRADENG> Station Analysis ... <RECORD> <TERMFR> Bureau poste <TRADENG> Post offices <RECORD> <TERMFR> Bureau poste <TRADENG> Post office ... <RECORD> <TERMFR> Isolation poste électrique <TRADENG> Substation insulation ... <RECORD> <TERMFR> Caserne pompier <TRADENG> Fire houses <SYNOFRE1> Poste incendie ... <RECORD> <TERMFR> Habitacle aéronef <TRADENG> Cockpits (aircraft) <SYNOFRE1> Poste pilotage ...	<RECORD> <TERMFR> La Poste <TRADENG> Postal services ... <RECORD> <TERMFR> Poste conduite <TRADENG> Operation platform <SYNOFRE1> Cabine conduite ... <RECORD> <TERMFR> POSTE DE TRAVAIL <TRADENG> WORK STATION <RECORD> <TERMFR> Poste de travail <TRADENG> Work Station <RECORD> <TERMFR> Poste de travail <TRADENG> Work station <RECORD> <TERMFR> Poste de travail <TRADENG> workstations <SYNOFRE1> Poste travail ...

representatives using our thesaurus (performance shown under column heading "Dthes"). In a last experiment, we accounted for related words found in the thesaurus for document surrogates only and under the additional condition that such relationships could be found within at least three terms (e.g. "moteur à combustion" is a valid candidate but not single term like "moteur"). On the other hand, we also included in the query all relationships that could be found using the search keywords (performance shown under the column heading "Dthes3Qthes").

From the average precision shown in Tables 20 and 21, we cannot infer that the available thesaurus is really helpful in improving retrieval effectiveness, at least as implemented in this study.

However, the Amaryllis corpus presents another interesting feature. The logical sections <TI> and <AB> are used to delimit respectively the title and the abstract of each French scientific article written by the author(s) while the logical section <MC> marks the manually assigned keywords extracted from the INIST thesaurus. Finally, the section delimited by the <KW> tags corresponds to the English version of the manually assigned keywords.

Table 20. Average precision for various indexing and searching strategies (Amaryllis)

	Average precision				
Query	Amaryllis TD	Amaryllis TD Qthes	Amaryllis TD Dthes	Amaryllis TD Dthes3Qthes	Amaryllis TDN
Model	25 queries	25 queries	25 queries	25 queries	25 queries
Okapi–npn	**45.75**	**45.45**	**44.28**	**44.85**	**53.65**
Lnu–ltc	43.07	44.28	41.75	43.45	49.87
dtu–dtc	39.09	41.12	40.25	42.81	47.97
atn–ntc	42.19	43.83	40.78	43.46	51.44
ltn–ntc	39.60	41.14	39.01	40.13	47.50
ntc–ntc	28.62	26.87	25.57	26.26	33.89
ltc–ltc	33.59	34.09	33.42	33.78	42.47
lnc–ltc	37.30	36.77	35.82	36.10	46.09
bnn–bnn	20.17	23.97	19.78	23.51	24.72
nnn–nnn	13.59	13.05	10.18	12.07	15.94

Table 21. Average precision using blind-query expansion (Amaryllis)

	Average precision				
Query	Amaryllis TD	Amaryllis TD Qthes	Amaryllis TD Dthes	Amaryllis TD Dthes3Qthes	Amaryllis TDN
Model	25 queries	25 queries	25 queries	25 queries	25 queries
Okapi–npn	45.75	45.45	44.28	44.85	53.65
5 docs / 10 terms	47.75	47.29	46.41	46.73	55.80
5 docs / 50 terms	**49.33**	48.27	47.84	47.61	**56.72**
5 docs / 100 terms	49.28	48.53	47.78	47.83	56.71
10 docs / 10 terms	47.71	47.43	46.28	47.21	55.58
10 docs / 50 terms	49.04	48.46	48.49	48.12	56.34
10 docs / 100 terms	48.96	**48.60**	**48.56**	**48.29**	56.34
25 docs / 10 terms	47.07	46.63	45.79	46.77	55.31
25 docs / 50 terms	48.02	47.64	47.23	47.85	55.82
25 docs / 100 terms	48.03	47.78	47.38	47.83	55.80

Therefore, using the Amaryllis corpus, we could investigate whether the manually assigned descriptors result in better retrieval performance than the automatically based indexing scheme. To this end, we evaluated the Amaryllis collection using all logical sections (denoted "All" in Table 22), using only the title and the abstract of the articles (denoted "TI & AB") or using only the manually assigned keywords (performance shows under the label "MC & KW").

The conclusions that can be drawn from the data shown in Table 22 are clear. For all retrieval models, the manually assigned keywords result in better average

Table 22. Average precision when comparing manual and automatic indexing procedures

	Average precision					
Query Model	Amaryllis T All	Amaryllis T TI & AB	Amaryllis T MC & KW	Amaryllis TD All	Amaryllis TD TI & AB	Amaryllis TD MC & KW
Okapi–npn	**36.33**	**23.94**	**29.79**	**45.75**	**30.45**	**38.11**
Lnu–ltc	34.79	22.74	25.81	43.07	28.22	32.17
dtu–dtc	31.82	23.89	28.51	39.09	27.23	32.29
atn–ntc	35.01	23.32	29.11	42.19	28.16	35.76
ltn–ntc	31.78	20.42	26.40	39.60	24.58	32.90
ntc–ntc	21.55	16.04	17.58	28.62	21.55	24.16
ltc–ltc	25.85	17.42	20.90	33.59	24.44	26.62
lnc–ltc	26.84	16.77	21.66	37.30	26.12	29.29
bnn–bnn	21.03	11.29	22.71	20.17	11.71	19.80
nnn–nnn	8.99	5.12	8.63	13.59	7.39	11.00

precision than the automatic indexing procedure, using the short queries built only from using the Title section or when using both the Title and Description parts of the topics. However, the best performance was achieved by combining the manually assigned descriptors with the indexing terms extracted from the title and the abstract of the scientific articles.

However, the users usually enter short queries and are concerned with the precision achieved after 5 or 10 retrieved articles. In order to obtain a picture of the relative merits of using various indexing strategies within this context, we reported in Table 23 the precision achieved after retrieving 5 or 10 documents using the Okapi probabilistic model. From this table, we can see that the manually assigned keyword indexing scheme (label "MC & KW") provides better results than does the automatic indexing approach (label "TI & AB") when considering the precision achieved after 5 documents. When comparing the precision after 10 retrieved items, both approaches perform in a very similar manner. Finally, when using both indexing approaches (performances given under the label "All"), we achieve the best performance results.

Table 23. Precision after 5 or 10 retrieved documents

Query Title only	Amaryllis All	Amaryllis TI & AB	Amaryllis MC & KW
Model	25 queries	25 queries	25 queries
Precision after 5	71.2%	59.2%	60.8%
Precision after 10	68.8%	54.4%	54.0%

Table 24. Official Amaryllis run descriptions

Run name	Query	Form	Model	Thesaurus	Query expansion	Precision
UniNEama1	TD	auto	Okapi	no	25 docs/50 terms	48.02
UniNEama2	TD	auto	Okapi	query	25 docs/25 terms	47.34
UniNEama3	TD	auto	Okapi	document	25 docs/50 terms	47.23
UniNEama4	TD	auto	Okapi	que & doc	10 docs/15 terms	47.78
UniNEamaN1	TDN	auto	Okapi	no	25 docs/50 terms	55.82

6 Conclusion

For our second participation in the CLEF retrieval tasks, we proposed a general stopword list and stemming procedure for French, Italian, German, Spanish and Finnish. We also tested a simple decompounding approach for German. For Dutch, Finnish and German, our objective was to examine 5-gram indexing and word-based (and decompounding-based) document representation, with respect to their ability to serve as distinct and independent sources of document content evidence, and to investigate whether combining these two (or three) indexing schemes would be a worthwhile strategy.

To improve bilingual information retrieval, we would suggest using not only one but two or three different translation sources to translate the query into the target languages. These combinations seem to improve retrieval effectiveness. In the multilingual environment, we demonstrated that a learning scheme such as logistic regression could perform effectively and as a second best solution, we suggest using a simple normalization procedure based on the document score.

Finally, in the Amaryllis experiments, we compared a manual with an automatic indexing strategy. We found that for French scientific papers manually assigning descriptors result in better performance than does automatic indexing based on the title and abstract. However, the best average precision is obtained when combining both manually assigned keywords and the automatic indexing scheme. With this corpus, we studied various possible techniques in which a specialized thesaurus could be used to improve average precision. However, the various strategies used in this paper do not demonstrate clear enhancement over a baseline that ignores the term relationships found in the thesaurus.

Acknowledgments

The author would like to thank C. Buckley from SabIR for giving us the opportunity to use the SMART system without which this study could not have been conducted. This research was supported in part by the SNSF (Swiss National Science Foundation) under grants 21-58 813.99 and 21-66 742.01.

References

[1] Savoy, J.: Report on CLEF-2001 Experiments: Effective Combined Query-Translation Approach. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 27–43 66, 69

[2] Robertson, S. E., Walker, S., Beaulieu, M.: Experimentation as a Way of Life: Okapi at TREC. Information Processing & Management **36** (2000) 95–108 69, 73

[3] Fox, C.: A Stop List for General Text. ACM-SIGIR Forum **24** (1999) 19–35 69

[4] Savoy, J.: A Stemming Procedure and Stopword List for General French Corpora. Journal of the American Society for Information Science **50** (1999) 944–952 69

[5] Sproat, R.: Morphology and Computation. The MIT Press, Cambridge (1992) 69

[6] Lovins, J. B.: Development of a Stemming Algorithm. Mechanical Translation and Computational Linguistics **11** (1968) 22–31 69

[7] Porter, M. F.: An Algorithm for Suffix Stripping. Program **14** (1980) 130–137 69

[8] Figuerola, C. G., Gómez, R., Zazo Rodríguez, A. F., Berrocal, J. L. A.: Spanish Monolingual Track: The Impact of Stemming on Retrieval. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 253–261 69

[9] Kraaij, W., Pohlmann, R.: Viewing Stemming as Recall Enhancement. In Proceedings of the ACM-SIGIR 1996. The ACM Press, New York (1995) 40–48 70

[10] Monz, C., de Rijke, M.: Shallow Morphological Analysis in Monolingual Information Retrieval for Dutch, German, and Italian. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 262–277 70

[11] Chen, A.: Multilingual Information Retrieval Using English and Chinese Queries. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 44–58 70

[12] Molina-Salgado, H., Moulinier, I., Knutson, M., Lund, E., Sekhon, K.: Thomson Legal and Regulatory at CLEF 2001: Monolingual and Bilingual Experiments. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 226–234 70

[13] Chen, A.: Cross-Language Retrieval Experiments at CLEF 2002. In *this volume* 72

[14] Buckley, C., Singhal, A., Mitra, M., Salton, G.: New Retrieval Approaches Using SMART. In Proceedings TREC-4. NIST, Gaithersburg (1996) 25–48 73, 74

[15] Singhal, A., Choi, J., Hindle, D., Lewis, D. D., Pereira, F.: AT&T at TREC-7. In Proceedings TREC-7. NIST, Gaithersburg (1999) 239–251 73

[16] Braschler, M., Peters, C.: CLEF 2002: Methodology and Metrics. In *this volume* 73

[17] Brand, R., Brünner, M.: Océ at CLEF 2002. In *this volume* 73
[18] McNamee, P., Mayfield, J., Piatko, C.: A Language-Independent Approach to European Text Retrieval. In: Peters, C. (ed.): Cross-Language Information Retrieval and Evaluation. Lecture Notes in Computer Science, Vol. 2069. Springer-Verlag, Berlin Heidelberg New York (2001) 131–139 73
[19] McNamee, P., Mayfield, J.: JHU/APL Experiments at CLEF: Translation Resources and Score Normalization. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 193–208 73
[20] Savoy, J.: Cross-Language Information Retrieval: Experiments Based on CLEF-2000 Corpora. Information Processing & Management (2002) to appear 74, 82
[21] Amati, G., Carpineto, C., Romano, G.: Italian Monolingual Information Retrieval with PROSIT. In *this volume* 74
[22] Nie, J. Y., Simard, M.: Using Statistical Translation Models for Bilingual IR. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.): Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science, Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 137–150 77
[23] Gachot, D. A., Lange, E., Yang, J.: The SYSTRAN NLP Browser: An Application of Machine Translation Technology. In: Grefenstette, G. (ed.): Cross-Language Information Retrieval. Kluwer, Boston (1998) 105–118 77
[24] Voorhees, E. M., Gupta, N. K., Johnson-Laird, B.: The Collection Fusion Problem. In Proceedings of TREC-3. NIST, Gaithersburg (1995) 95–104 81
[25] Kwok, K. L., Grunfeld, L., Lewis, D. D.: TREC-3 Ad-hoc, Routing Retrieval and Thresholding Experiments Using PIRCS. In Proceedings of TREC-3. NIST, Gaithersburg (1995) 247–255 82
[26] Dumais, S. T.: Latent Semantic Indexing (LSI) and TREC-2. In Proceedings of TREC-2. NIST, Gaithersburg (1994) 105–115 82
[27] Powell, A. L., French, J. C., Callan, J., Connell, M., Viles, C. L.: The Impact of Database Selection on Distributed Searching. In Proceedings of ACM-SIGIR'2000. The ACM Press, New York (2000) 232–239 82
[28] Flury, B.: A First Course in Multivariate Statistics. Springer, New York (1997) 82
[29] Hosmer, D. W., Lemeshow, S.: Applied Logistic Regression. 2nd edn. John Wiley, New York (2000) 82
[30] Le Calvé, A., Savoy, J.: Database Merging Strategy Based on Logistic Regression. Information Processing & Management, **36** (2000) 341-359 82
[31] Venables, W. N., Ripley, B. D.: Modern Applied Statistics with S-PLUS. Springer, New York (1999) 82