Report on CLIR Task for the NTCIR-5 Evaluation Campaign

Samir ABDOU, Jacques SAVOY
Institut interfacultaire d'informatique, University of Neuchatel
Pierre-à-Mazel 7, 2000 Neuchatel, Switzerland
{ Samir.Abdou, Jacques.Savoy }@unine.ch

Abstract

This paper describes our second participation in an evaluation campaign involving the Chinese, Japanese, Korean and English languages (NTCIR-5). Our participation is motivated by four objectives: 1) study the retrieval performances of various IR models for these languages; 2) compare the relative retrieval effectiveness of bigram and automatic word-segmenting approaches for Chinese and Japanese languages; 3) propose a new blind-query expansion hopefully capable of improving mean average precision; and 4) evaluate the relative performance of the various merging strategies used to combine separate result lists extracted from a corpus written in English, Chinese, Japanese or Korean.

Keywords: CLIR, MLIR, blind-query expansion, probabilistic IR model.

1 Monolingual IR for Asian languages

1.1 Overview of NTCIR-5 test collection

Table 1 displays various statistics from the fifth NTCIR corpora (for more information, see [5]). In this paper, when analyzing the number of pertinent documents per topic, we only considered rigid assessments and thus only "highly relevant" and "relevant" items are seen as being relevant. A comparison of the number of relevant documents per topic, as shown in Table 1, indicated that for the English collection the median number of relevant items per topic was 33, while for the Chinese corpus it was only 26 and 25.5 for the Korean and 24 for the Japanese. Clearly, the number of relevant articles was greater for the English (3,073) corpus, when compared to the Japanese (2,112), Chinese (1,885) or Korean (1,829) collections.

For the various search models used, the bottom part of Table 1 provides an overview their efficiency, indicating the size of each collection in terms of storage space requirements. For example, the row labeled "# postings" indicates the number of indexing terms (words or bigrams) in the inverted file, followed by the size of this inverted file and the time (user CPU time + system CPU time) needed to build it. For the Chinese and Japanese languages we used both the bigram and an automatic word segmentation approach. To carry out our experiments we used a 2 x Intel Xeon 3.06 GHz (memory: 3.6 GB, swap: 15 GB, disk: 5 x 250 GB). The average query size (expressed in number of tokens following stopword removal) and time (in seconds) required to execute both short (T only) and medium-size (DN) queries is shown in the lower rows (computations made without blind-query expansion). Clearly, the use of bigrams as indexing

	English	Chinese		Japanese		Korean
Size (in MB)	438 MB	1,100 MB		1,100 MB		312 MB
# of documents	259,050	901,446		858,400		220,374
# of topics	49	50		47		50
# rel. items	3,073	1,885		2,112		1,829
Mean	62.73	37.7		44.94		36.58
Median	33	26		24		25.5
Indexing scheme	word	word	bigram	word	bigram	bigram
# postings	494,745	333,017	3,661,338	329,884	909,631	345,751
Inverted file size	278 MB	1,786 MB	3,386 MB	955 MB	1,387 MB	586 MB
Building time	1,150 sec.	2,397 sec.	4,726 sec.	1,650 sec.	2,044 sec.	757 sec.
T query size	4.8 wd/que	5.3 wd/que	6.8 bi/que	4.6 wd/que	8.2 bi/que	7.3 bi/que
Search time	0.218 sec	0.275 sec	0.246 sec	0.232 sec	0.270 sec	0.233 sec
DN query size	69.8 wd/que	94.0 wd/que	173.3 bi/que	68.8 wd/que	100.7 bi/que	140.8 bi/que
Search time	0.409 sec	1.631 sec	1.066 sec	0.712 sec	0.770 sec	0.537 sec

Table□. NTCIR-5 CLIR test collection statistics (rigid evaluation)

strategy required more time to build the inverted file (e.g., for the Chinese corpus, with the time increasing from 2,397 sec. to 4,726 sec., or by 97.2%). The time differences between word-based and bigram searches were not really important.

1.2 Indexing and searching strategies

In analyzing these new test collections and in order to draw some useful conclusions, we considered it important to evaluate the retrieval performance under various conditions. We decided to evaluate a variety of indexing and search models in order to obtain this broader view. First we considered adopting a binary indexing scheme in which each document (or topic) was represented by a set of indexing terms (word or bigram), without assigning any weights (IR model denoted "doc=bnn, query=bnn" or "bnn-bnn"). In order to weight the presence of each indexing term, we might account for the term occurrence frequency ("nnn-nnn") or we might also account for their frequency within the collection (or for *idf*). Moreover, when using cosine normalization, each indexing weight could vary within the range of 0 to 1 ("ntc-ntc" or "*tf idf*").

Other variants might also be created. For example, the *tf* component could be computed as 0.5+0.5·[*tf* / max *tf* in a document] ("atn"). We could also consider that a term's presence in a shorter document provides stronger evidence than in a longer document, leading to more complex IR models; i.e. the IR models denoted by "Lnu" [2] and "dtu" [12]. See the Appendix for details on the exact weighting formulas.

In addition to previous models based on the vector-space model, we also considered probabilistic approaches, such as the well-known Okapi model (or BM25) [8]. As with other probabilistic models, we might apply the Deviation from Randomness (DFR) framework [1], based on two information measures. These are Inf^1 (measuring the informative content of the document with respect to the whole collection), and Inf^2 (measuring the information gain with respect to the *elite* set, the set of documents where the underlying term occurs). To reflect the indexing weight w_{ij} attached to term t_j in document D_i, we have:

$$w_{ij} = \text{Inf}^1{}_{ij} \cdot \text{Inf}^2{}_{ij} = -\log_2[\text{Prob}^1{}_{ij}] \cdot (1–\text{Prob}^2{}_{ij}) \qquad (1)$$

in which $\text{Prob}^1{}_{ij}$ is the probability of having by pure chance tf_{ij} occurrences of the term t_j in a document (various probabilistic models could be used to estimate this probability). On the other hand, $\text{Prob}^2{}_{ij}$ is the probability of encountering a new occurrence of term t_j in the given document, once tf_{ij} occurrences of this term have already been found.

Within this DFR framework, the PB2 model is defined as follows:

$$\text{Inf}^1{}_{ij} = -\log_2[(e^{-\square_j} \cdot \square_j{}^{tf_{ij}})/tf_{ij}!] \qquad (2)$$

$$\text{Prob}^2{}_{ij} = 1 - [(tc_j + 1) / (n \cdot (tfn_{ij} + 1))] \qquad (3)$$

with $tfn_{ij} = tf_{ij} \cdot \log_2[1 + ((c \cdot mean\ dl) / l_i)]$
and $\square_j = tc_j / n$

where tc_j indicates the number of occurrences of term t_j in the collection, l_i the length (number of indexing terms) of document D_i, *mean dl* the average document length, and *c* a constant.

As a variant, the model denoted I(*n*)L2 (used only for the English corpus) is defined as follows:

$$\text{Inf}^1{}_{ij} = tfn_{ij} \cdot \log_2[(n+1) / (df_j+0.5)] \qquad (4)$$

$$\text{Prob}^2{}_{ij} = tfn_{ij} / (tfn_{ij} + 1) \qquad (5)$$

where df_j indicates the number of documents indexed using the term t_j, and *n* the number of documents in the corpus.

In defining these various IR models, we have implicitly admitted that words are our indexing unit. For the English language, finding words in a sentence is usually a simple task. For the Japanese language, each sentence was automatically segmented using the morphological analyzer ChaSen [7], and the Chinese corpus was segmented using Mandarin Tools (freely available at www.mandarintools.com).

For the Asian languages, we also indexed documents by applying an overlapping bigram approach, an indexing scheme found to be effective for various Chinese collections [6], or during previous NTCIR campaigns [3], [11]. Based on this technique for example, the sequence "ABCD EFG" would generate the following bigrams {"AB," "BC," "CD," "EF," and "FG"}. In our work, we generated these overlapping bigrams for Asian characters only, using Latin characters, digits, spaces and other punctuation marks (collected for each language in their respective encoding) to stop bigram generation. Moreover, we did not split any words written in ASCII characters. The most frequent terms may of course be removed before indexing. For the Chinese language, we defined a list of 39 most frequent unigrams, 49 most frequent bigrams and a list of 91 words (used when applying a word-based indexing scheme in Chinese). For Japanese we defined a short stopword list of 30 words and another of 20 bigrams, and for Korean our stoplist was composed of 91 bigrams.

Before generating the bigrams for the Japanese documents, we removed all Hiragana characters, given that these characters are used mainly to write words used only for grammatical purposes (e.g., *doing*, *in*, *of*), as well as inflectional endings for verbs, adjectives and nouns. Moreover, half-width characters were replaced by their corresponding full-width version.

For the English collection, we based the indexing process on the SMART stopword (571 words) and stemmer procedure.

1.3 Evaluation of various IR systems

To measure retrieval performance, we adopted non-interpolated mean average precision (MAP). To determine whether or not a given search strategy would be better than another, we based our statistical validation on the bootstrap approach [9]. In the tables appearing in this paper we have thus underlined any statistically significant differences, with the means serving as baseline amounts (two-sided non-parametric bootstrap test, significance level at 5%).

MAP values obtained by the eleven search models under three query formulations (T, D, DN) are shown in Table 2 (for the English and Japanese collections), where the best performance for any given condition is shown in bold (these values were used as the baseline for our statistical tests in Tables 2 and 3). Table 3 lists performances obtained using the Korean (bigram) and Chinese (bigram or word) corpora.

Surprisingly, this data shows that the best retrieval scheme for short queries is not always the same as that for longer topics. For the Japanese collection (both bigram & word), the best retrieval models were always the PB2 when facing with short queries (T or D) and Okapi when using longer queries (DN). For the Chinese corpus (both bigram & word), the best retrieval model was always the PB2. Based on our statistical testing, the differences in performance were not always significant (e.g., for the Chinese corpus, the difference between Okapi and PB2 models is only significant for the D queries when using bigram indexing scheme). For the Japanese corpus, the word-based indexing scheme seemed to result in better retrieval performance. For example, based on the nine best performing IR models, and using T queries, the word-based indexing schemes shows, in average, a small 4.4% enhancement.

When comparing bigram and word-based representations for the Chinese collection (see Table 3), the performance difference seemed to favor more clearly word-based indexing. For example, based on the six best performing IR models and T queries, the average improvement was around 3.9% and favored the word-based IR schemes.

For the Korean corpus, increasing the query size from T to D improves, in average for the nine most effective IR models, the MAP of 8%, and of 28% for the DN over D query formulation.

1.4 Blind-query expansion

It was observed that pseudo-relevance feedback (blind-query expansion) seemed to be a useful technique for enhancing retrieval effectiveness. In this study, we adopted Rocchio's approach [2] whereby the system was allowed to add *m* terms extracted from the *k* best ranked documents from the original query using the following formula:

	Mean average precision								
	English (word, 49 queries)			Japanese (bigram, 47 queries)			Japanese (word, 47 queries)		
Model	T	D	DN	T	D	DN	T	D	DN
I(*n*)L2/PB2	0.3591	0.3548	**0.4556**	**0.2717**	**0.2829**	0.3957	**0.2895**	**0.3120**	0.3925
Okapi-npn	**0.3692**	0.3615	0.4555	0.2660	0.2694	**0.4079**	0.2655	0.2657	**0.4002**
Lnu-ltc	0.3562	0.3551	0.4185	0.2579	0.2648	0.3876	0.2743	0.2814	0.3780
dtu-dtn	0.3577	**0.3748**	0.3949	0.2461	0.2564	0.3660	0.2735	0.2944	0.3514
atn-ntc	0.3423	0.3458	0.3926	0.1799	0.1986	0.3287	0.2109	0.2335	0.3315
ltn-ntc	0.3275	0.3244	0.3608	0.2651	0.2538	0.3200	0.2723	0.2678	0.3115
ntc-ntc	0.2345	0.2522	0.3061	0.1292	0.1289	0.2302	0.1227	0.1343	0.1987
ltc-ltc	0.2509	0.2869	0.3675	0.0992	0.1104	0.2220	0.0945	0.1106	0.2106
lnc-ltc	0.2617	0.2868	0.3951	0.1070	0.1174	0.2354	0.1132	0.1236	0.2475
bnn-bnn	0.2000	0.1277	0.0964	0.1403	0.1422	0.1092	0.1403	0.0977	0.0564
nnn-nnn	0.1048	0.0701	0.0806	0.0981	0.0851	0.0900	0.1055	0.0477	0.0445

Table 2. MAP for various IR models (monolingual English and Japanese)

	Mean average precision								
	Korean (bigram, 50 queries)			Chinese (bigram, 50 queries)			Chinese (word, 50 queries)		
Model	T	D	DN	T	D	DN	T	D	DN
PB2	0.3729	**0.4141**	**0.5022**	**0.3042**	**0.2878**	**0.3973**	**0.3246**	**0.2974**	**0.4136**
Okapi-npn	0.3630	0.3823	0.4940	0.2995	0.2584	0.3887	0.3230	0.2816	0.4135
Lnu-ltc	**0.3973**	0.3962	0.4628	0.2999	0.2644	0.3667	0.3227	0.2910	0.3864
dtu-dtn	0.3673	0.3907	0.4497	0.2866	0.2565	0.3564	0.2894	0.2812	0.3760
atn-ntc	0.3270	0.3489	0.4541	0.2527	0.2378	0.3548	0.2578	0.2585	0.3668
ltn-ntc	0.3708	0.3688	0.4442	0.2886	0.2571	0.3421	0.2833	0.2570	0.3404
ntc-ntc	0.2506	0.2886	0.3666	0.2130	0.2093	0.3138	0.1645	0.1748	0.2741
ltc-ltc	0.2260	0.2638	0.3794	0.1933	0.2056	0.3382	0.1772	0.1931	0.3416
lnc-ltc	0.2414	0.2773	0.4172	0.2053	0.2115	0.3546	0.2189	0.2292	0.3754
bnn-bnn	0.2348	0.1840	0.1078	0.1629	0.1334	0.1139	0.1542	0.0915	0.0613
nnn-nnn	0.1770	0.1287	0.1911	0.1170	0.0911	0.1333	0.0738	0.0527	0.0468

Table 3. MAP for various IR models (monolingual Korean and Chinese)

$$Q' = □ \cdot Q + (□ / k) \cdot \sum_{j=1}^{k} w_{ij} \qquad (6)$$

in which Q' denotes the new query built for the previous query Q, and w_{ij} denotes the indexing term weight attached to the term t_j in the document D_i. In our evaluation, we fixed □ = 0.75, □ = 0.75.

For a new blind-query expansion denoted IDFQE ("IDF Query Expansion"), we adopted the following procedure. First form the root set of search terms composed of all terms included in the original query Q and all indexing terms appearing in the *k* best ranked documents. The weight value for each term in this root set would be computed as follows:

$$w'_j = □ \cdot I_Q(tj) \cdot tf_j + (□ / k) \cdot \sum_{i=1}^{k} I_{D_i}(t_j) \cdot idf_j \qquad (7)$$

$$\text{with } I_Q(t_j) = 1 \text{ if } t_j □ Q, \text{ 0 otherwise}$$

where for term t_j, $idf_j = \ln(n/df_j)$ (the classical idf value) and $I_Q(t_j)$ (or $I_{Di}(t_j)$), an indicator function returning the value 1 if the term t_j belonging to the query Q (or the document D_i), otherwise the value is 0. In this weighting scheme, if a term appeared only in the original query Q, its weight would be $□ \cdot tf_j$, while a term appearing only in one document would have a weight of $(□ / k) \cdot idf_j$.

The root set elements were then sorted in decreasing order according to their weight. To form the new query Q', we selected the top *m* search terms, and the weights attached to these selected terms in the new query were the same as those used in the root set. We thus used the same weighting scheme to select and weight the new search terms.

We used the two probabilistic models to evaluate this proposition. Table 4 summarizes some results achieved for the English, and Japanese (bigram and word-based indexing scheme) collections, while Table 5 shows some retrieval performances for the Korean (bigram) and Chinese (bigram or word-based indexing) corpora. In these tables, the rows labeled "PB2," (C, J, and K) "I(*n*)L2" (E) or "Okapi-npn" (baseline) indicate the MAP before applying this blind-query expansion procedure. The rows starting with "#doc/#term" indicate the number of top-ranked documents and the number of terms used to enlarge the original query. Finally, the rows labeled "& Rocchio" (or "& IDFQE") depict the MAP following Rocchio's approach (Eq. 6) (or our idf method, Eq. 7), and using the parameter setting specified in the previous row.

From the data shown in Tables 4 and 5, we could infer that the blind query expansion technique improved MAP, and this improvement was usually statistically significant (underlined values in these tables). When comparing both probabilistic models, this strategy seemed to perform better with the PB2 (or I(*n*)L2) than with the Okapi model. Moreover, enhancement percentages were greater for short topics than for longer ones. For example, in the Japanese

	Mean average precision								
	English (word, 49 queries)			Japanese (bigram, 47 queries)			Japanese (word, 47 queries)		
Model	T	D	DN	T	D	DN	T	D	DN
I(*n*)L2/PB2	0.3591	0.3548	0.4556	0.2717	0.2829	0.3957	0.2895	0.3120	0.3925
#doc/#term	15 / 100	15 / 100	10 / 60	15 / 100	10 / 75	10 / 100	15 / 100	20 / 120	10 / 80
& Rocchio	**0.4450**	**0.4625**	0.5027	0.3429	**0.3596**	**0.4240**	0.3479	0.3581	0.3983
#doc/#term	15 / 100	15 / 100	10 / 60	15 / 100	10 / 75	15 / 100	15 / 100	15 / 70	10 / 80
& IDFQE	0.4389	0.4543	**0.5039**	**0.3476**	0.3563	0.4180	**0.3690**	**0.3609**	**0.4071**
Okapi-npn	0.3692	0.3615	0.4555	0.2660	0.2694	0.4079	0.2655	0.2657	0.4002
#doc/#term	15 / 100	15 / 100	10 / 60	10 / 150	10 / 150	10 / 100	10 / 100	15 / 100	10 / 100
& Rocchio	0.4420	0.4478	0.4573	0.3266	0.3212	0.4103	0.3523	0.3433	0.4021
#doc/#term	15 / 100	15 / 100	10 / 60	15 / 100	15 / 100	15 / 100	20 / 100	20 / 100	10 / 100
& IDFQE	**0.4476**	**0.4529**	**0.4994**	**0.3501**	**0.3617**	**0.4307**	**0.3681**	**0.3763**	**0.4378**

Table□4. MAP with blind-query expansion (monolingual English and Japanese)

	Mean average precision								
	Korean (bigram, 50 queries)			Chinese (bigram, 50 queries)			Chinese (word, 50 queries)		
Model	T	D	DN	T	D	DN	T	D	DN
PB2	0.3729	0.4141	0.5022	0.3042	0.2878	0.3973	0.3246	0.2974	0.4136
#doc/#term	15 / 140	5 / 60	5 / 150	10 / 100	5 / 100	5 / 125	5 / 75	10 / 75	10 / 100
& Rocchio	0.3899	0.4719	0.5158	0.3782	0.3616	0.4241	0.3547	0.3822	0.4088
#doc/#term	15 / 100	10 / 100	15 / 100	10 / 75	10 / 125	5 / 125	5 / 125	10 / 75	10 / 100
& IDFQE	**0.4253**	**0.4766**	**0.5228**	**0.3912**	**0.3861**	**0.4288**	**0.3769**	**0.3954**	**0.4400**
Okapi-npn	0.3630	0.3823	0.4940	0.2995	0.2584	0.3887	0.3230	0.2816	0.4135
#doc/#term	15 / 100	5 / 100	15 / 200	5 / 125	10 / 100	5 / 125	5 / 75	10 / 75	10 / 100
& Rocchio	0.4346	0.4563	0.4881	**0.3559**	0.3176	0.3854	**0.3788**	0.3522	0.4252
#doc/#term	15 / 100	10 / 100	15 / 150	5 / 125	10 / 75	5 / 125	5 / 125	10 / 75	10 / 100
& IDFQE	**0.4453**	**0.4667**	**0.5304**	0.3557	**0.3659**	**0.4242**	0.3778	**0.3576**	**0.4479**

Table□5. MAP with blind query expansion (monolingual Korean and Chinese)

	Mean average precision							
	English (word, 49 queries)			Japanese (word or bigam, 47 queries)				
Model	T	D	DN	T	T	D	D	DN
#doc/#term	I(*n*)L2 (wd) 15 / 50 R *0.4425*	I(*n*)L2 (wd) 20 / 70 R 0.4494	PB2 (wd) 15 / 40 I 0.4589	Okapi (wd) 20 / 100 I 0.3681	PB2 (wd) 15 / 100 I 0.3690	Okapi (wd) 20 / 100 I 0.3763	Okapi (wd) 20 / 100 I 0.3763	Okapi (wd) 10 / 100 I 0.4378
#doc/#term	Okapi (wd) 15 / 100 R 0.4420	Okapi (wd) 15 / 100 R *0.4478*	I(*n*)L2 (wd) 10 / 60 R 0.5027	Okapi (bi) 15 / 100 I 0.3501	Okapi (wd) 10 / 100 R 0.3523	Okapi (bi) 15 / 100 I 0.3617	Okapi (wd) 15 / 100 R 0.3433	Okapi (bi) 15 / 150 I 0.4307
Round-rob.	0.4427	0.4514	0.4942	0.3639	0.3729	0.3761	0.3708	0.4405
SumRSV	**0.4544**	0.4573	0.5018	0.3637	0.3742	0.3752	**0.3742**	0.4486
NormRSV	*0.4539*	0.4575	*0.5019*	*0.3734*	*0.3839*	0.3780	0.3681	0.4496
Z-score	0.4540	***0.4581***	**0.5039**	0.3693	**0.3852**	*0.3773*	*0.3692*	***0.4504***
Z-score W	0.4517	0.4572	0.4982	**0.3754**	0.3839	**0.3801**	0.3736	0.4499

Table□6. MAP with various data fusion schemes (English and Japanese corpora)

collection (word-based indexing) using the PB2 model and T topics, blind query expansion improved mean performance, ranging from 0.2895 to 0.3479 (+20.1% in relative effectiveness) with Rocchio's approach or to 0.3690 with IDFQE (+27.5%). With DN query formulation, the MAP improves from 0.3925 to 0.4071 (+3.7%) using our IDFQE scheme.

1.5 Data fusion

For a strategy that would enhance retrieval effectiveness, we can combine two or more result lists. As a first data fusion strategy, we considered the round-robin approach whereby we selected one document in turn from all individual lists and removed duplicates, retaining the highest ranking instances. Various other data fusion operators were suggested [4], however the simple linear combination (denoted "SumRSV") usually seemed to provide the best performance [10], [4], or at least good overall performance [11]. For a given set of result lists i = 1, 2, …, r, this combined operator was defined as SumRSV = $\sum$ RSV_i, being the simple sum of all document scores (RSV_i) obtained by each search model.

As a third data fusion strategy we normalized document scores within each collection through dividing them by the maximum score. As a variant of this normalized score merging scheme (denoted "NormRSV"), we might normalize the document RSV_k scores within the *i*th result list, as follows:

$$NormRSV_k = ((RSV_k – Min^i) / (Max^i - Min^i)) \qquad (8)$$

As a fourth data fusion strategy, we suggest merging the retrieved documents according to the Z-score, computed for each result list. Within this scheme, we would normalize retrieval status values for each document D_k provided by the *i*th result list, as computed by the following formula:

$$\text{Z-score } RSV_k = \square_i \,.\, [((RSV_k\text{-}Mean^i) / Stdev^i)+ \square^i],$$
$$\square^i = ((Mean^i\text{- } Min^i) / Stdev^i\,) \qquad (9)$$

within which $Mean^i$ denotes the average of the RSV_k, $Stdev^i$ the standard deviation, and $\square_i$ (usually fixed at 1), used to reflect the retrieval performance of the underlying retrieval model. When the coefficients $\square_i$ are not all fixed at 1, the data fusion operator is denoted as "Z-score W."

	Mean average precision								
	Chinese (word, bigram, unibigram, 50 queries)					Korean (bigram, 50 queries)			
Model	T	T	D	D	DN	T	T	D	DN
#doc/#term	PB2 (wd) 5 / 75 R 0.3547	PB2 (wd) 5 / 75 R 0.3547	PB2 (wd) 10 / 75 I 0.3954	PB2 (wd) 10 / 75 R 0.3822	PB2 (wd) 10 / 100 I 0.4400	Okapi (bi) 15 / 100 I 0.4453	Okapi (bi) 15 / 100 I 0.4453	Okapi (bi) 5 / 100 R 0.4563	Okapi (bi) 15 / 150 I 0.5304
#doc/#term	PB2 (bi) 10 / 75 I 0.3912	Okapi (wd) 5 / 125 I 0.3778	Okapi (wd) 10 / 75 I 0.3576	Okapi (wd) 10 / 75 I 0.3576		PB2 (bi) 15 / 100 I 0.4253	Okapi (bi) 15 / 100 R 0.4346	PB2 (bi) 10 / 100 I *0.4766*	PB2 (bi) 5 / 150 R 0.5158
#doc/#term	Oka(unibi) 5 / 100 I 0.3620		PB2(unibi) 10 / 100 I 0.3738	PB2(bi) 5 / 100 R 0.3616	PB2(unibi) 10 / 100 I 0.4557				
Round-rob.	0.3780	0.3691	0.3850	0.3734	0.4498	0.4393	0.4463	0.4746	0.5267
SumRSV	**0.4121**	0.3712	*0.4057*	0.3956	0.4618	0.4351	0.4526	0.4892	0.5293
NormRSV	0.4062	0.3800	0.4064	0.4006	0.4592	0.4396	*0.4525*	0.4913	0.5333
Z-score	*0.4076*	*0.3828*	0.4091	***0.4026***	*0.4585*	*0.4395*	**0.4547**	***0.4921***	*0.5362*
Z-score W	0.4050	**0.3837**	**0.4127**	0.3980	**0.4593**	**0.4415**	0.4545	0.4900	**0.5383**

Table□7. MAP with various data fusion schemes (Chinese and Korean corpora)

We could of course combine different document surrogates during the indexing process. For the Chinese corpus for example, we might index the documents (and the queries) using both unigram (or character) and bigram approaches (denoted by the label “unibi” in this paper).

Table 6 shows the MAP obtained for the English and Japanese collections, for each of the T, D and DN queries. Table 7 lists the same information for the Chinese and Korean corpora, in which the best performing single IR scheme served as a baseline for our statistical testing.

From this data, we could see that combining two or more IR models might sometimes improve retrieval effectiveness (differences with the best single system were however not statistically significant except three cases with the Japanese corpus). Moreover the Z-score scheme tended to produce the best performance. It is difficult however to predict which data fusion operator would produce the best result, even when a particular data fusion scheme improved performance during single runs. Current and past experiments tend to indicate that combining short query results provides more improvement than does combining longer topics [11].

Results from some of our official monolingual runs are indicated in italics in shown in Tables 6 and 7. Given that we introduced a bug in our IDFQE blind-query expansion scheme, our official results depicted usually a lower MAP than the corrected version (differences given in Table 8).

	Official MAP	Corrected MAP
UniNE-J-J-DN-01	**0.4480**	0.4504
UniNE-J-J-T-02	0.3705	0.3734
UniNE-J-J-D-03	**0.3823**	0.3773
UniNE-J-J-T-04	0.3815	0.3852
UniNE-J-J-D-05	0.3717	0.3692
UniNE-C-C-DN-01	**0.4419**	0.4585
UniNE-C-C-T-02	0.4104	0.4076
UniNE-C-C-D-03	0.3846	0.4057
UniNE-C-C-T-04	0.3806	0.3828
UniNE-C-C-D-05	0.4002	0.4026
UniNE-K-K-DN-01	0.5313	0.5362
UniNE-K-K-T-02	0.4494	0.4395
UniNE-K-K-D-03	0.4845	0.4921
UniNE-K-K-T-04	0.4468	0.4525
UniNE-K-K-D-05	0.4748	0.4766

Table 8. Official and corrected results

For the Japanese monolingual task for example, the UniNE-J-J-T-02 was based on the “NormRSV” operator combining two Okapi runs (the first was a word-based indexing scheme and the second based on bigrams) with an official MAP 0.3705 and a corrected MAP of 0.3734.

2 Bilingual IR

As explained in our last NTCIR campaign paper [11], we translated each topic written in English into the three Asian languages using freely available resources on the Web. In this study, we chose four different machine translation (MT) systems and three machine-readable bilingual dictionaries (MRDs) to translate the topics:

SYSTRAN	www.systranlinks.com
WORLDLINGO	www.worldlingo.com
ALPHAWORKS	www.alphaWorks.ibm.com
APPLIEDLANGUAGE	www.appliedLanguage.com
DICT	www.dicts.info
ECTACO	www.ectaco.co.uk/free-online-dictionaries
BABYLON	www.babylon.com

For the bilingual dictionaries, we submitted search keywords word-by-word after lemmatizing (e.g., “weapons“ will be replaced by “weapon“). In response to each word submitted, the MRD system provided not only one but several translation terms (in an unspecified order). In our experiments, we decided to pick the first available translation (e.g., labeled “Babylon 1” or “Dict 1”), the first two (e.g., “Babylon 2”) or the first three (e.g., “Dict 3”).

Table 9 shows MAP when translating English topics employing the four MT systems, the three MRDs and the Okapi model. This table also contains the retrieval performance for manually translated topics, with the first row (“Okapi-npn”) being used as a baseline. Compared to our previous work with European languages [10] and also to manually translated topics, machine translated topics generally provided poor performance levels. Based on the T queries and the best single query translation resource (the Alphawork MT system in this case), the resulting performance was only 40.3% that of a monolingual search for the Chinese language (0.1208 vs. 0.2995), 56.6% for the Korean language (0.2055 vs. 0.3630) or 69.7% for the Japanese (0.1855 vs. 0.2660). Moreover, differences in mean average precision were always statistically significant and favored the manual topic translation approach.

The Alphawork MT system seemed to produce the best translated topics for all languages. Morevoer, MT systems tended to result in better performance level than MRDs approaches. The poor overall query translation performance seemed to be caused by including proper names in numerous topics (e.g, 15 topics had a person’s name, 4 a geographical name, 7 had other proper names such as “Linux,” “Anthrax” or “Mir”), and these names were usually not properly translated by the MRDs or MT systems.

3 Multilingual IR

In this section, we will investigate situations in which users submit a topic in English in order to retrieve relevant documents in English, Chinese, Japanese and Korean (CJKE). The different collections were indexed separately and, once the original or translated request (see Section 2) was received, a ranked list of retrieved items was returned. From these lists we needed to produce a unique ranked result list, using a merging strategy described further on in this section.

	Mean average precision								
	Chinese (bigram, 50 queries)			Japanese (bigram, 47 queries)			Korean (bigram, 50 queries)		
Model	T	D	DN	T	D	DN	T	D	DN
Okapi-npn	0.2995	0.2584	0.3887	0.2660	0.2694	0.4079	0.3630	0.3823	0.4940
Babylon 1	0.0505	0.0486	0.1059	0.0987	0.1161	0.1467	n/a	n/a	n/a
Babylon 2	0.0433	0.0516	0.0943	0.1250	0.1137	0.1375	n/a	n/a	n/a
Babylon 3	0.0438	0.0480	0.1113	0.1191	0.1212	0.1329	n/a	n/a	n/a
Ectaco 1	n/a	n/a	n/a	n/a	n/a	n/a	0.0632	0.0392	0.0500
Dict 1	0.0411	0.0329	0.0249	0.0570	0.0248	0.0366	0.0473	0.0373	0.0287
Dict 2	0.0700	0.0495	0.0552	0.0736	0.0341	0.0411	0.0644	0.0715	0.0615
Dict 3	0.0715	0.0540	0.0630	0.0745	0.0314	0.0407	0.0780	0.0767	0.0781
WorldLing	0.1055	0.1252	0.2256	0.1417	0.1597	0.2637	0.1988	0.2113	0.3418
AlphaW	**0.1208**	**0.1663**	**0.2526**	**0.1855**	**0.2021**	**0.3037**	**0.2055**	0.2117	0.3363
AppliedLg	0.1052	0.1255	0.2269	0.1417	0.1609	0.2642	0.1988	**0.2118**	**0.3421**
Systran	0.1052	0.1255	0.2269	0.1417	0.1609	0.2642	0.1988	0.2113	0.3415
Combined	indexing : bigram only			Systran / WorldLingo / AlphaWorks					
with Okapi	0.1317	0.1689	0.2713	**0.1927**	0.2039	**0.3056**	0.2396	0.2557	0.3914
with PB2	0.1355	**0.1946**	**0.2816**	0.1925	0.2214	0.2937	0.2503	0.2848	0.4060

Table 9. MAP for various query translation approaches (Okapi model)

	Mean average precision				
	T	T	D	D	DN
English	Oka& DFR	Okapi	Oka& DFR	DFR	Oka& DFR
(out of 49 queries)	0.4540	0.4420	0.4572	0.4494	0.5019
Chinese	Oka& PB2	Okapi(wd)	PB2 & PB2	PB2 (wd)	PB2 & PB2
(out of 50 queries)	0.2417	0.2360	0.2751	0.2363	0.2904
Japanese	Oka& Oka	Okapi(wd)	Oka& Oka	Okapi(wd)	Oka& Oka
(out of 47 queries)	0.2631	0.2631	0.2878	0.2728	0.3379
Korean	Oka& PB2	Okapi(bi)	Oka& DFR	PB2 (bi)	Oka& PB2
(out of 50 queries)	0.3374	0.3289	0.3586	0.3677	0.4120
Merging strategy CJKE					
Round-robin (baseline)	0.2244	0.2169	0.2548	0.2410	0.2839
Raw-score	0.2165	**0.2332**	0.2364	0.2169	0.2823
MaxRSV	0.2248	0.2102	0.2468	0.1979	0.2830
NormRSV (Eq. 8)	0.2256	0.2259	*0.2475*	0.2322	0.2830
Biased RR E,K=1/C,J=2	0.2036	0.1965	0.2328	0.2172	0.2600
Z-score (Eq. 9)	**0.2333**	*0.2261*	**0.2698**	***0.2578***	**0.2950**
Z-score W E,K=1/C,J=1.25	*0.2113*	0.1965	0.2475	0.2316	*0.2695*

Table 10. MAP of various merging strategies for CJKE collection (official in italics)

As a first approach, we considered the round-robin method. As a second merging approach, we took the document score into account, denoted as RSV_k for document D_k. Known as raw-score merging, this strategy, produced a final list sorted by document score, as computed by each collection. As a third scheme, we could either normalize the RSV_k by using the document score of the retrieved record in the first position ("MaxRSV") or using Eq. 8 ("NormRSV").

As a fifth merging scheme, we would suggest a biased round-robin approach which extracted not just one document per collection per round, but one document from both the English and Korean collections and two from the Japanese and Chinese. This type of merging strategy exploited the fact that the Japanese and Chinese corpora contain more articles than do the English or the Korean corpora. Finally, we applied our Z-score (see Eq. 9) and then under the "Z-score W" label we assigned a weight of 1.25 for the Japanese and Chinese result lists, and 1 for the English and Korean runs.

The data depicted in Table 10 also indicates that resulted in retrieval effectiveness that could be viewed as statistically superior to that of the round-robin baseline. As a first approach, the simple and normalized merging schemes ("MaxRSV" or "NormRSV") provided reasonable performance levels. Also, our biased round-robin scheme did not perform better when compared to the simple round-robin version (it was difficult a priori to know whether a given corpus would really contain more relevant items than another). The Z-score provided statistically better performance levels than did the round-robin approach.

Conclusion

Based on our evaluations, we may infer that for both the Chinese or Japanese language, using a good automatic word-segmentation procedure seems to produce slightly better retrieval performances than an bigram indexing scheme (average difference between 3.9% and 7%, see Tables 2 and 3). Based on our evaluation of the various IR models, we can obtain the best retrieval performance levels using the PB2 probabilistc model before blind-query expansion, and using Okapi after blind-query expansion for the Japanese and Korean languages (Tables 2 through 5).

Compared to Rocchio's query expansion (Eq. 6), better performance may be obtained from our idf-based model (see Eq. 7). The performance differences with an approach without query expansion are usually statistically significant and in favor of a query expanded approach. To further improve retrieval effectiveness, a data fusion approach could also be considered, although this technique would require additional computational resources with and uncertain improvement (Tables 6 and 7).

From an analysis of bilingual search performances, the number and quality of freely available translation resources were questionable. When translating the topics from English into Chinese, Japanese or Korean language, overall retrieval effectiveness decreases by more than 30% for the Japanese language, compared to more than 50% for Chinese and Korean (see Table 9).

When evaluating various merging strategies (see Table 10), it appears that the Z-score merging procedure produces better retrieval performance when result lists provided by separate collections are merged.

Acknowledgments

This research was supported in part by the Swiss NSF (Grant #200020-103420).

References

[1] Amati, G., & van Rijsbergen, C.J. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM-TOIS*, 20(4):357-389, 2002.

[2] Buckley, C., Singhal, A., Mitra, M., & Salton, G. New retrieval approaches using SMART. *Proceedings of TREC-4*, pp. 25-48, 1996.

[3] Chen, A., & Gey, F.C. Experiments on cross-language and patent retrieval at NTCIR-3 workshop. *Proceedings of NTCIR-3, 2003.*

[4] Fox, E.A., & Shaw, J.A. Combination of multiple searches. *Proceedings TREC-2*, pp. 243-249, 1994.

[5] Kishida, K., Chen, K.-H., Lee, S., Kuriyama, K., Kando, N., Chen, H.-H., & Myaeng, S.H. Overview of CLIR Task at the Fifth NTCIR Workshop. *Proceedings of NTCIR-5*, Tokyo, 2005.

[6] Luk, R.W.P., & Kwok, K.L.. A comparison of Chinese document indexing strategies and retrieval models. *ACM-TALIP*, 1(3): 225-268, 2002.

[7] Matsumoto, Y., Kitauchi, A., Yamashita, T., Hirano, Y., Matsuda, H., & Asahara, M. Japanese morphological analysis system ChaSen. Technical Report NAIST-IS-TR99009, NAIST, 1999 (available at http://chasen.aist-nara.ac.jp/).

[8] Robertson, S.E., Walker, S., & Beaulieu, M. Experimentation as a way of life: Okapi at TREC. *IP&M*, 36(1), 95-108, 2000.

[9] Savoy, J. Statistical inference in retrieval effectiveness evaluation. *IP&M*, 33(4):495-512, 1997.

[10] Savoy, J. Combining multiple strategies for effective monolingual and cross-lingual retrieval. *IR Journal*, 7(1-2):121-148, 2004.

[11] Savoy, J. Comparative study of monolingual and multilingual search models for use with Asian languages. ACM TALIP, 4(3), 2005.

[12] Singhal, A., Choi, J., Hindle, D., Lewis, D.D., & Pereira, F. AT&T at TREC-7. *Proceedings of TREC-7*, 239-251, 1999.

Appendix

bnn	$w_{ij} = 1$	npn	$w_{ij} = tf_{ij} \cdot \ln[(n-df_j) / df_j]$	ltn	$w_{ij} = [\ln(tf_{ij}) + 1] \cdot idf_j$
nnn	$w_{ij} = tf_{ij}$	lnc	$w_{ij} = \frac{\ln(tf_{ij})+1}{\sqrt{\sum_{k=1}^{t} \left(\ln(tf_{ik})+1\right)^2}}$	ntc	$w_{ij} = tf_{ij} \cdot idf_j \Big/ \sqrt{\sum_{k=1}^{t} \left(tf_{ik} \cdot idf_k\right)^2}$

atn	$w_{ij} = idf_j \cdot [0.5+0.5 \cdot tf_{ij}/maxtf_{i.}]$	dtn	$w_{ij} = [\ln[\ln(tf_{ij})+1] + 1] \cdot idf_j$
ntn	$w_{ij} = tf_{ij} \cdot idf_j$		
Lnu	$w_{ij} = \frac{\left[\left(1+\ln(tf_{ij})\right) \Big/ \left(\ln(\text{mean tf})+1\right)\right]}{(1-\text{slope}) \cdot \text{pivot} + \text{slope} \cdot nt_i}$	ltc	$w_{ij} = \left(\ln(tf_{ij})+1\right) \cdot idf_j \Big/ \sqrt{\sum_{k=1}^{t} \left(\left(\ln(tf_{ik})+1\right) \cdot idf_k\right)^2}$
Okapi	$w_{ij} = \left((k_1+1) \cdot tf_{ij}\right) \Big/ \left(K+tf_{ij}\right)$	dtu	$w_{ij} = \frac{\left(\ln\left(\ln(tf_{ij})+1\right)+1\right) \cdot idf_j}{(1-\text{slope}) \cdot \text{pivot} + \left(\text{slope} \cdot nt_i\right)}$

Table A.1. Weighting schemes