# TREC-6 English and Chinese Retrieval Experiments using PIRCS


K.L. Kwok, L. Grunfeld and J.H. Xu

Computer Science Dept., Queens College, CUNY
Flushing, NY 11367
Email: kwok@ir.cs.qc.edu URL: http://ir.cs.qc.edu



## Abstract

For Trec-6 ad-hoc experiments, we continue to use two-stage retrieval with pseudo-feedback from top-ranked un-judged documents for both Chinese and English. We perform three types of retrieval characterized by queries formed using title only, description only and all sections of the given topics. For short queries mainly derived from title or description section, query terms are weighted by average term frequency avtf introduced previously. For Chinese, we employ a combination of representation (character, bigram and short-word) strategy, returning the highest average non-interpolated precision that is even better than some manual approaches. In English ad-hoc, we try a document re-ranking strategy for the first stage retrieval based on occurrence of selected query term pairs, so as to have better result in the second stage. Performance for English ad-hoc is also highly competitive for both very short and long queries.

In routing, a strategy of combining different methods of query formation and retrieval is used. These include no learning ad-hoc type queries, learning from the more current FBIS5 documents only, queries learnt from selecting the best set of known relevant documents based on a genetic algorithm, and queries that are trained from a back-propagation neural network with hidden nodes. Average precision results are among the best four. In addition, we also participate in high precision and the filtering track.


## 1. Introduction

Results from our PIRCS retrieval engine have been demonstrated to be consistently among the best in previous TREC's, and is again confirmed in TREC-6. As usual we ran the main tasks of ad-hoc and routing retrieval experiments. In addition, we also participated in the Chinese, batch filtering and the high precision tracks. It has been a busy and time-consuming endeavor as many of the ad-hoc and filtering experiments were done three times using different query lengths or to meet different objectives.

PIRCS has been described in previous TREC proceedings and references thereof. It is based on probabilistic indexing and retrieval, conceptualized as a three layer network with adaptive capability to support feedback and query expansion, and operates via activation spreading. The basic model evaluates a retrieval status value (RSV) for each query ($q_a$) document ($d_i$) pair as a combination of a query-focused process that spreads activation from document to query through common terms k, and an analogous document-focused process operating vice versa, as follows:

$$RSV = \alpha*\Sigma_k S(q_{ak} /L_a) * w_{ik} + (1-\alpha)*\Sigma_k S(d_{ik}/L_i) * w_{ak}$$

where $0\le\alpha\le1$, $q_{ak}$ , $d_{ik}$ are the frequency of term k in a query or document respectively, $L_a$, $L_i$ are the query or document lengths, and S(.) is a sigmoid-like function to suppress outlying values. A major difference of our model from other probabilistic approaches is to treat a document or query as non-monolithic, but constituted of conceptual components (which we approximate as terms). This leads us to formulate in a collection of components rather than documents, and allows us to account for the non-binary occurrence of terms in a natural way. For example, in the usual discriminatory weighting formula for term k: $w_{ak}$ = log [p*(1-q)/(1-p)/q], p = Pr(term k present | relevant) is set to a query 'self-learn' value of $q_{ak}$ /$L_a$ , and q = Pr(term k present |~relevant) = $F_k$/M, the collection term frequency of k, $F_k$, divided by the total number of terms M used in the collection. This we call the inverse collection term frequency ICTF and differs from the usual IDF. Moreover, as the system learns from relevant documents, p can be trained to a value intermediate between the basic self-learn value and that given by the known relevants according to a learning procedure [Kwok 1995]. Our system also uses two-word adjacency phrases as terms for representation in addition to single words, and deals with long documents by segmenting them into approximately equal sub-documents of 550 words ending on a paragraph boundary. For the final retrieval list, retrieval status values (RSV) of the top three sub-documents of the same document are combined with decreasing weights to return a final RSV. This in effect favors retrieval of longer documents that contain positive evidence in different sub-parts of it, and differs from the approach that uses document length normalization weighting via training from previous documents [Singal, Buckley & Mitra 1996]. These PIRCS strategies have been in use since TREC-2.

The remaining parts of this paper are organized into four main sections describing the Chinese, ad-hoc and high precision, routing and filtering experiments, and our conclusion.

## 2. Chinese Track

### 2.1 Methodology

Chinese ad-hoc experiments consist of 26 new topics (number 29 -54) retrieving against the same 170 MB Xinhua and Peoples' Daily collections of last year. In an effort to study the effect of query lengths on retrieval effectiveness as in English ad-hoc, we perform three experiments using different portions of a given topic as query, namely: title only, description only (desc) and all sections (all). Our convention for naming these runs are: pirc7Ct, pirc7Cd and pirc7Ca respectively, with 7 denoting 1997 and C for Chinese.

We continue to use the short-word segmenter that we developed last year to segment Chinese texts. Short-word means words of 1 to 3 characters long. This procedure involves four steps: (i) lexicon look-up using longest match to segment input texts into smaller chunks; (ii) simple language rules to segment chunks into short-word candidates; (iii) discover new short-words based on frequency filtering; (iv) expand initial lexicon with the new short-words and re-process collection. Last year's initial lexicon of 2K has been enlarged to 27K entries to provide better coverage of common short-words. After adding new words discovered from the collection, the final lexicon size is about 43K. In addition, no stopwords are used as we have shown in [Kwok 1997b] that their effect on retrieval is minimal. On the other hand, accidentally removing a crucial stopword in particular queries may substantially bring down retrieval effectiveness, especially for short ones.

In [Kwok 1997a], we also have studied the behavior of retrieval using three representation types separately, viz: character, bigram and short-word with character. Retrieval using character representation alone is much inferior to the other two, because single characters are ambiguous. Bigrams are much more specific, and they exhaustively cover the Chinese words used in the collection that are two characters in size. However, they suffer from over-generation as one does not have good heuristics to decide which bigram is meaningful, and which ones are not. Moreover, some Chinese words are truly single character, and these would not be represented accurately with bigrams alone. Short-word representation in theory is the best, but can be in error because the segmentation algorithm may give wrong results. We recognize the problem with segmentation, and have used short-words together with characters simultaneously in a document or query for representation last year. Both bigrams and short-word indexing with characters provide similar retrieval effectiveness.

For the TREC-6 topics, using short-word indexing with character gives the following average query lengths: 6 for title, 11.6 for 'desc', and 37.8 for 'all'. With bigram indexing the corresponding figures of: 9.4, 10 and 65.2 respectively. We conjecture that bigrams and short-word indexing with characters may complement each other, and have used a combination strategy in these TREC-6 experiments. The collection was indexed two ways: bigram and short-word with character representation. For each query, two separate retrievals were performed using the two representations, and the resultant document lists are combined using equal weights. This combination strategy is employed for the long and short queries only (i.e. 'all' and 'desc' query types). For the very short queries, we use only short-word indexing, because bigram results for these titles can be quite poor compared to short-words. Two-stage retrieval with pseudo-feedback is employed as the standard retrieval strategy. For the short queries using title or description only, avtf weighting of query terms is employed {Kwok 1996]. In addition, the PIRCS system parameters are tuned as discussed in [Kwok 1997a].

### 2.2 Chinese Ad-Hoc Results

Official results for our Chinese retrieval are tabulated in Table 1, with three data columns for the very short (title), short (description) and long queries (all sections). Percentage changes are measured from values in the 'title' column. It appears that PIRCS continues to perform correctly for processing Chinese, and our strategy of combination pays off nicely. Thus, the average precision value of 0.6263 for query type 'all' is very high, and the 2795 number of relevants retrieved at 1000 documents represents over 94% of the 'pooled' known relevants of 2928. Precision at 10 documents of 0.8737 means close to 9 of the top 10 retrieved are relevant. Even at 100 documents retrieved, over 55% of them are on target. Using shorter queries of the description section only (desc) and title only leads to successfully less performance. However, the average precision of 0.4755 is still quite high for title queries having only on average of 6 terms. It appears that these topics are quite 'easy' for retrieval, as they often involve low level factual terms and few generalized concepts. Moreover, none of the queries has fewer than 16 relevant documents in the collection, and it is often these cases of just a few relevants that can lead to difficult retrievals with low precision values.

**Query Type**

| | Title % | Desc % | All % |
|---|---|---|---|
| Relv.Ret | 2547 | 2674 5 | 2795 10 |
| Avg.Prec | .4755 | .5423 14 | .6263 32 |
| P@10 | .7115 | .7962 12 | .8737 23 |
| P@20 | .6692 | .7519 12 | .8135 22 |
| P@30 | .6192 | .6974 13 | .7718 25 |
| P@100 | .4327 | .5035 16 | .5542 28 |

**Comparison with Median**

| | > = < | > = < | > = < |
|---|---|---|---|
| Avg.Prec | 8 1 17,6 | 16,1 1 9 | 25,8 0 1 |
| RR@100 | 6 2 18,6 | 15 3 8 | 21,1 3 2 |
| RR@1K | 10,3 7 9,1 | 13,2 8 5 | 18,6 6 2 |

**Table 1: Chinese Ad Hoc Results for 26 Queries**

In the same table, we also have the comparison with the medium from all sites. Under the query type 'all', the Avg.Prec row shows that 25 of our queries are better than medium, with 8 of them being best, and only 1 is below medium. Under other query types, the comparison is less favorable. For example, very short 'title' queries only have 8 above medium, 1 equal and 17 below medium; of these 17, 6 queries return the worst results. This is to be expected since there is only one standard for comparison: results from all query types are used to obtain the medium without differentiation.

As an example of the benefit of combination, we focus on the 'all' query type. The bigram retrieval composing this result has by itself an average precision of .5755 and relevants retrieved of 2735. Similarly, the short-word indexing with character alone has .6031 and 2791 values for these measures. They combine to give values of .6263 and 2795 respectively, an improvement of about 4% in precision from the better one, and a few more in the relevants retrieved.

In almost all cases, two-stage retrieval improves over one stage only, ranging from 0% to nearly 32% in the case of titles using short-word with character representation. Results for avtf weighting of short queries however are not uniform. For bigrams, improvements in all four cases of title and description during 1st and 2nd stage retrieval are observed. For short-word with character, only the first stage retrieval using title representation has slight improvment. Had we combined short-word without avtf retrieval and bigram with avtf, the average precision would have been .502 for the title (versus the official .476), and .566 for description only (versus .542).

For two years in a row Chinese retrieval in the TREC environment has shown much higher effectiveness (50-100% higher) than English for both long and short queries. It is not clear if this is due to the data being much 'easier', or if this is due to some intrinsic properties of the Chinese language. It is of interest to continue further experiments using more diverse collections and queries to throw some light into this phenomenon.

## 3. Ad-Hoc Retrieval and High Precision Track

### 3.1 Methodology

The ad-hoc task requires retrieval of 50 topics numbered 301-350 on the old TREC disk 4 consisting of Congressional Records and Federal Registry, and a new disk 5 of FBIS and LA Times documents. From the given topics, three types of queries can be formed according to what sections of the topics are used to define the queries: very short queries obtained from the title section only, short ones from the description section only (desc), and long queries from all sections (all). The average number of unique terms in each of these query types are: 2.58, 7.12 and 21.26 respectively. For five of these queries, there is only one word in the title query type. Except for one, these are highly specific terms: 'hydroponics', 'polygamy', 'retir', 'agoraphobia', 'metabolism'. Two of these five actually have more words in the title, but some are removed due to frequency considerations. Some of the other titles are also specific, containing two or three words.

We continue to use 2-stage retrieval with pseudo-feedback as in last year. For short queries we try out a document re-ranking procedure on the outputs of the initial retrieval. The purpose is to promote relevant documents to the top in order to enhance the quality of the 'feedback' documents and thereby improving second stage retrieval results. Re-ranking is based on giving additional weights to selected query term pairs that appear in documents as further evidence of relevance. Short queries having few terms have the advantage that most probably all terms are important for relevance judgment, and selection of term pairs may be less prone to error. For longer queries, with numerous term pairs, selecting useful term pairs become more difficult and wrong ones may actually harm retrieval. We decided to use the procedure only on the very short queries from titles only. It turns out that the procedure depresses results slightly for both cases of short and very short queries.

## 3.2 Ad-Hoc Retrieval Results

Our official ad-hoc results are shown in Table 2 for the 3 query types. Most unexpected is the finding that average precision values for very short queries derived from 'titles' actually perform best with .2556, followed with the queries from 'all' sections with .2332. However, the quality of retrieval at the top 10 to 20 documents is better with the 'all' query type (e.g. P@10 = .4260 vs .4020), as well as the number of relevants retrieved at 1000 documents (Relv.Ret = 2674 vs 2384). 2674 is about 58% of the 'pooled' relevants of 4611. Also, the 'all' queries have much more favorable comparison with the medium from all sites (41 better or equal with 5 being best, and 9 below) than the 'title' queries (30 better or equal with 6 best, and 20 below with 3 worst). This rather erratic behavior we believe is due to the very good retrievals of a few very specific, single term title queries, as discussed below.

**Query Types**

| | Title % | Desc % | All % |
|---|---|---|---|
| Relv.Ret | 2377 | 1728 -27 | 2674 12 |
| Avg.Prec | .2556 | .1533 -40 | .2332 -9 |
| P@10 | .4020 | .2940 -27 | .4260 6 |
| P@20 | .3390 | .2450 -28 | .3460 2 |
| P@30 | .3093 | .2133 -31 | .3093 0 |
| P@100 | .2092 | .1330 -36 | .2120 1 |

**Comparison with Median**

| | > = < | > = < | > = < |
|---|---|---|---|
| Avg.Prec | 26,6 4 20, 3 | 26,4 2 22 | 38,5 3 9 |
| RR@100 | 25,10 12 13, 7 | 23,4 10 17,5 | 33,10 11 6 |
| RR@1K | 24,10 15 11,2 | 21,6 9 20 | 37,18 10 3 |

**Table 2: Ad Hoc Results for 50 Queries**

Our results for the 'desc' query type has been found to be erroneous due to some procedures being incompletely run for certain queries, and we did not realize it at the time of experiment and submission. Re-doing it properly gives an average precision of .1928 and relevants retrieved of 1938. Since some of the descriptions lack title terms, we also did an experiment that includes both the title and description section to form queries. The results are .2247 for precision and 2468 for relevants retrieved. The precision is still below that of the title query run, in contrast to some other reports.

A comparison of results among the query types shows that a single highly specific term can give very good results, and that the description section which tries to explain what the title word says in plain and less technical vocabulary (sometimes without the specific term) fails miserably. These plain terms are often of high frequency and ambiguous in meaning. The retrieval engine is not sufficiently intelligent to comprehend the explanation, gets confounded with all the common terms, and retrieve accordingly. Long queries also suffer from the noise introduced. In the following we show the average precision and relevants retrieved results of the five queries that have only one single term:

#312 Title: 'hydroponics'; description: 'the science of growing plants in water or some substance other than soil'.
#316 Title: 'polygamy'; description: 'a look at the roots and prevalence of polygamy, in the world today'.
#318 Title: 'retir'; description: 'aside from the united states, which country offers the best living conditions and quality of life for a u.s. retiree?'
348: Title: 'agoraphobia'; description: 'is the fear of open or public places(agoraphobia) a widespread disorder or relatively unknown?'.
#349 Title: 'metabolism'; description: 'the chemical reactions necessary to keep living cells healthy and/or producing energy'.

| query | title | desc | desc* | all |
|---|---|---|---|---|
| #312 | .9151/11 | .0016/ 4 | .0016/ 4 | .1580/11 |
| #316 | .6926/26 | .3914/25 | .3914/25 | .4158/25 |
| #318 | .0000/ 0 | .0030/20 | .0030/20 | .0049/21 |
| #348 | .8100/ 5 | .1411/ 3 | .1436/ 5 | .0528/ 3 |
| #349 | .2610/35 | .0098/14 | .0070/16 | .2180/41 |

*corrected

**Average Precision/Relevants Retrieved @1000**

Except for #318 where all query types perform poorly, 'title' gives excellent to medium results. Just the difference of these five between 'title' and 'all' adds 0.0366 to the 'all' average precision result, bringing it to 0.2698 for the latter. Similar large differences can be accounted for when comparing 'title' and 'desc' results. In contrast, the description section of the Chinese queries actually contains related concept terms and are useful for retrieval, giving 14% improvement in average precision when compared with 'title' only queries.

## 3.3 High Precision Track

Initially we intend to participate in this track via manual intervention. Due to time constraints, we found that we could not make the deadline and decided to submit the top documents of our ad-hoc runs. Thus, our high precision results are fully automatic without manual handling. This in effect would provide a lower-bound to this track. As a method to boost precision, we in addition ran our re-ranking procedure described earlier on the final retrieval lists for the 'title' query type. This turns out to depress its value a little bit.

Results are tabulated in Table 3. They are not competitive and far below medium because we lack a human feedback loop. In the comparison of precision @10 with medium, it is interesting to note that there are always some queries (8 for 'title', 1 for 'desc' and 4 for 'all') for which our automatic method can achieve maximum without manual help. They include all four of the highly specific single term queries.

**Query Types**

| | Title % | Desc % | All % |
|---|---|---|---|
| P@10 | .3980 | .3360 -16 | .4260 7 |
| RP@10 | .4163 | .3509 -16 | .4384 5 |
| unrP@10 | .0766 | .0561 -27 | .0574 -25 |

**Comparison with Median**

| | > = < | > = < | > = < |
|---|---|---|---|
| P@10 | 10,8 10 30, 19 | 7,1 12 31, 15 | 8,4 13 29, 6 |

**Table 3: High Precision Results for 50 Queries**

## 4. Routing Retrieval and Filtering track
## 4.1 Methodology
### 4.1.1 Combination of Retrievals

Combination of retrievals have been studied by us as well as other groups (see e.g. [Hull, Pedersen & Schutze 1996] and references thereof). Two different retrievals using different methods, added together will frequently be superior to both of them individually. One reason is that non-relevant documents tend to be more random in the retrieved pool. Another reason for adding retrievals is, that no method performs best in every situation. It may be easier to build small specialist queries and add them after retrieval, favoring the better performer.

A simple specialist query is one developed from the text of the original query. It will perform well compared to a query that is based on relevant documents, if the available relevant documents do not cover all the facets of the query concept, or if the documents contain mainly irrelevant information.

Many of the queries have training data available for as far back as 1987. It is a concern, that the old information is not timely anymore, and will in fact harm the query. A specialist query trained from the current FBIS5 documents only will avoid this problem.

### 4.1.2 Routing Query Pirc7R1

The first routing query submitted is an addition of four retrievals. All of then use the PIRCS retrieval engine, the training documents and the dictionary used are different.

a) Retrieval with no training documents.

This is essentially the same retrieval as used for ad-hoc. This retrieval will performs well if the training documents are noisy or if the original query can be defined by a few well chosen terms. Examples where it did well are query 10003 "Privatization in Peru" and query 78 "Greenpeace activities".

b) Retrieval using the FBIS5 training documents only.

This retrieval will perform well if the query concept is time dependent. Examples where it did well are query 228 "Success stories in recent years concerning environmental recovery", query 100 "Controlling transfer of high technology" and query 1 " A Pending antitrust case".

These two retrievals use only statistics available from the FBIS5 collection and their combination comprised the retrieval used for the filtering track

c) Two retrievals using genetic algorithms to select training documents.

Many queries have a large number of evaluated training documents of varying quality. Retrieval can be improved if only a subset of training documents are used. We use genetic algorithms search to find this subset for the 39 queries that were repeated from TREC5. (For the others we used the top sub-documents for each relevant document as usual).

Genetic algorithms [Gol89] is a search procedure based on the survival of the fittest. The algorithm can be summarized as follows [Gre88]

```
Procedure GA
begin
        initialize population P(0)
        evaluate P(0)
        t=1
        repeat
```

select P(t) from P(t-1)
recombine P(t)
evaluate P(t)
until (termination condition)
end

We use genetic algorithms in this search space by performing the following steps.

1. Create training and testing query region. The top 1500 retrieved documents from the fbis 5 collection plus all short judged documents from the other collections are divided into 2 equal parts for this purpose.
2. Run GA for 6 generations to select best query
3. Swap training and testing data
4. Run GA for 6 generations to select best query

This procedure yields two more retrievals.

### 4.1.3 Routing Query Pirc7R2

We have recently began experimenting with utilizing backpropagation neural network for information retrieval. Our starting point was NevProp, a publicly available c program maintained by Phil Goodman of the University of Nevada. The results by itself were not good, but it invariably enhance the pircs retrieval, when the two retrievals were combined.

The input layer of the backprop neural network consists of about 100 terms selected by the pircs system, plus 60 positive term pairs plus 6 negative term pairs. The hidden layer has 3 nodes and the output layer has one node. This parameters were selected after experimentation with a very limited number of queries. The training data was the same as the training data for the genetic algorithms search.

The second routing query adds a retrieval by a backpropagation neural network to the first routing query. While the bp retrieval by itself was not as good as pircs, it is sufficiently different from it to make addition beneficial.

### 4.1.4 Performance based addition of retrievals.

Retrievals are combined using the equation

$$RSV_j = \sum_i C_i * RSV_{ji}$$

The RSV of the j-th document is the sum of the RSVs for all retrievals I, multiplied by a constant Ci for retrieval I. We may attempt to predict the performance of a method by doing a retrieval on the training data. In general the overall performance is better if a higher value is assigned to the better retrieval.

For trec4 we experimented with setting Ci to the 11 point average and normalizing retrieval weights by dividing all RSVs by the highest ranked RSV. There are some problems with this method. Although the 11pt average is an indication of the strength of the retrieval, it may not be the most optimal coefficient. The normalization also presents a problem. Bad retrievals frequently have a flat RSV curve, the normalization procedure would result with all documents RSV being close to 1.0000.

To overcome these problems, we do not normalize and we determine the Ci by trying to maximize the 11pt average on the training database.

## 4.2 Routing Retrieval Results

| | pircs7R1 | | | pircs7R2 | | |
|---|---|---|---|---|---|---|
| | > | = | < | > | = | < |
| avg prec | 36(2) | 1 | 10 | 36(6) | 3 | 8 |
| rel ret @100 | 34(6) | 4 | 9 | 36(6) | 3 | 8 |
| rel ret @1000 | 34(12) | 6(3) | 7 | 34(12) | 7(3) | 6 |

**Table 4.1 Comparison of routing results with mean. Number in parenthesis is number of best values.**

Performance of the official runs was well above median. (see table 4.1) . The strategy of adding retrievals appears to have paid off.

Looking at individual runs (Table 4.2) further underscores the value of combining specialist retrievals. The backprop run although weak by itself improved the pircs7R1 resulting in pircs7R2 our best result, similarly the query only run (no 1) when added to the fbis only run (no2) resulted in the filtering run, an over 10% improvement.

Table 4.3 shows the performance of various retrievals on the FBIS5 collection. As can be seen the run which only used the FBIS training documents performed much better than all the others. This resulted in inflated values for the addition coefficients for this run at the combination phase. Although we limited the maximum coefficient to .6, further limits should have been place on this run, since it was purely retrospective as opposed to the others which used all the training data.

| run no | name | avg | pct imp | rel retr | pct imp |
|---|---|---|---|---|---|
| 1 | query only | 0.2487 | 0% | 4648 | 0% |
| 2 | fbis only | 0.2992 | 20% | 5120 | 10% |
| 3 | all docs | 0.3114 | 25% | 4793 | 3% |
| 4 | ga 1 | 0.3362 | 35% | 5082 | 9% |
| 5 | ga 2 | 0.3431 | 38% | 5087 | 9% |

| | | | | | |
|---|---|---|---|---|---|
| 6 | backprop | 0.2417 | -3% | 4611 | -1% |
| add runs | | | | | |
| 1+2 | filtering | 0.3325 | 34% | 5109 | 10% |
| 4+5 | ga | 0.3461 | 39% | 5097 | 10% |
| 4+5+6 | ga+np | 0.3501 | 41% | 5124 | 10% |
| 1+2+4 +5 | pircs7R1 | 0.3605 | 45% | 5284 | 14% |
| 1+2+4 +5+6 | pircs7R2 | 0.3783 | 52% | 5288 | 14% |

**Table 4.2 Routing runs and combinations on FBIS6.**

| run no | name | avg | pct imp |
|---|---|---|---|
| 1 | query only | 0.2199 | 0% |
| 2 | fbis only | 0.6231 | 183% |
| 3 | all docs | 0.3831 | 74% |
| 4 | ga 1 | 0.4220 | 92% |
| 5 | ga 2 | 0.4145 | 88% |
| 6 | backprop | 0.3297 | 50% |
| add runs | | | |
| 1+2 | filtering | 0.6389 | 191% |
| 4+5 | ga | 0.4336 | 97% |
| 4+5+6 | ga+np | 0.4780 | 117% |

**Table 4.3 Routing runs and combinations on FBIS5.**

Since the ga training was only run for the 38 old queries, we separated performance statistics into old and new queries in table 4.4. The gain over the 38 old queries for the ga run ( add 4+5) was 12% over run 3 which is the pircs run using all top relevant subdocuments.
Another interesting detail is that for the 9 new queries, there is very little gain made by training from relevant documents, but combination of retrievals averted disaster. Comparing runs 2 and 3 shows the importance of term statistics. All runs except for runs 1 and 2 used the old trec5 dictionary and statistics. For the old 38 queries the run 2 using all training docs did much better than run 3, which had access only to the fbis 5 training docs, for the new 9 where the training documents were the same, run 3 did much better. The decision not to include the additional words and statistics from fbis5, which was made for time saving reasons, probably hurt overall performance.

| run no | name | avg old 38 | pct imp | avg new 9 | pct imp |
|---|---|---|---|---|---|
| 1 | query only | 0.2422 | 0% | 0.2765 | 0% |
| 2 | fbis only | 0.2943 | 22% | 0.3012 | 9% |
| 3 | all docs | 0.3341 | 38% | 0.2156 | -22% |
| 4 | ga 1 | 0.3648 | 51% | | |
| 5 | ga 2 | 0.3733 | 54% | | |
| 6 | backprop | 0.2664 | 10% | 0.1376 | -50% |
| add runs | | | | | |
| 4+5 | ga | 0.3771 | 56% | | |
| **1+2+4+5 | pircs7R1 | 0.3678 | 52% | 0.3294 | 19% |
| **1+2+4+5+6 | pircs7R2 | 0.3921 | 62% | 0.3201 | 16% |

**Table 4.4 Routing runs and combinations on fbis 6 separated into old and new queries**.
** note: for the avg new 9 col the runs don not include runs 4 and 5

## 4.3 Filtering Track

The filtering query is a standard pircs query based on the fbis 5 documents only, added to a query that is based on the original text of the query. Thresholds were developed by finding the rsv for which the maximum value is reached for each of the three utility measures. The second set lowers the threshold by 10%, based on the observation that if the retrieval is retrospective the relevant documents have a lower rank and the maximum rsv is reached earlier.

| run | > | = | < |
|---|---|---|---|
| pirc7f11 | 15(4) | 12(2) | 20 |
| pirc7f12 | 17(3) | 13(2) | 17 |
| pirc7f21 | 19(4) | 4 | 24 |
| pirc7f22 | 23(5) | 5 | 19 |
| pirc7a1 | 16(1) | 15 | 16 |
| pirc7a2 | 23(1) | 14 | 10 |

**Table 4.5 Comparison of filtering results with mean. Number in parenthesis is number of best values**.

We report here on the corrected retrieval submitted after the trec conference. The results for f1 and f2 utility values are a bit less then the median, while for the asp utility value is above. Lowering the thresholds improved the asp run, but made the f1 and f2 runs worse.

## 5. Conclusion

TREC-6 Chinese experiments continue to confirm TREC-5 observations that accurate word segmentation is not a pre-requisite for effective IR. Using a combination of simple representations, our PIRCS engine is able to return the best and very high effectiveness for another set of 26 queries. Our ad-hoc English retrieval is also among the best two submissions for both very short and long queries. Single word queries of a very specific and unambiguous nature performs like database retrieval and can give very good results. Longer descriptions only serve to confuse the retrieval engine under such circumstances. However, most information needs cannot be described by this type of representation. Our routing experiments introduce combination of new retrieval methods and also perform admirably among the top four submissions. It appears that PIRCS can perform consistently good under various circumstances.

## Acknowledgments

This work is partially supported by a grant from the Department of Defense.

## References

Goldberg, D. E. Genetic Algorithms in Search Optimization & Machne Learning. Addison Wesley 1989.

Hull, D.A., Pedersen, J.O. & Schutze, H (1996). Method combination for document retrieval. In: Proc. 19th Ann. Intl. ACM SIGIR Conf. On R&D in IR. Frei, H.P., Harman, D. Schauble, P. & Wilkinson, R. Aug 18-22, 1996. Pp.279-287.

Kwok, K.L. (1997a). Comparing representations in Chinese information retrieval. In: Proc. 20th Ann. Intl. ACM SIGIR Conf. On R&D in IR. Belkin, N.J., Narasimhalu, D. & Willett, P. (eds). July 27-31, 1997. Pp.34-41.

Kwok, K.L. (1997b). Lexicon effects on Chinese information retrieval. In: Proc. 2nd Conf. On Empirical Methods in NLP. Cardie, C. & Weischedel, R. (eds). Aug 1-2, 1997. Pp.141-148.

Kwok, K.L. (1996). A new method of weighting query terms for ad-hoc retrieval. In: Proc. 19th Ann. Intl. ACM SIGIR Conf. On R&D in IR. Frei, H.P., Harman, D. Schauble, P. & Wilkinson, R. Aug 18-22, 1996. Pp.187-195.

Kwok, K.L. (1995). A network approach to probabilistic information retrieval. ACM Transactions on Office Information Systems. 13:325-353.

Kwok, K.L. & Grunfeld, L (1997). TREC-5 English and Chinese experiments using PIRCS. In: The Fifth Text REtrieval Conference (TREC-5). To be published.

Kwok, K.L. & Grunfeld, L (1996). TREC-4 ad-hoc, routing retrieval and filtering experiments using PIRCS. In: The Fourth Text REtrieval Conference (TREC-4). Harman, D.K. (Ed.). NIST Special Publication 500-236, pp.145-152.

Singhal, A., Buckley, C. & Mitra, M (1996). Pivoted document length normalization. In: Proc. 19th Ann. Intl. ACM SIGIR Conf. On R&D in IR. Frei, H.P., Harman, D. Schauble, P. & Wilkinson, R. Aug 18-22, 1996. Pp.21-29.