TREC-5 English and Chinese Retrieval Experiments using PIRCS

K.L. Kwok & L. Grunfeld

Computer Science Dept., Queens College, CUNY,
Flushing, NY 11367. email: kklqc@cunyvm.cuny.edu

ABSTRACT

Two English automatic ad-hoc runs have been submitted: pircsAAS uses short and pircsAAL employs long topics. Our new avtf*ildf term weighting was used for short queries. 2-stage retrieval were performed. Both automatic runs are much better than the overall automatic average. Two manual runs are based on short topics: pircsAM1 employs double weighting for user-selected query terms and pircsAM2 additionally extends these queries with new terms chosen manually. They perform about average compared with the the overall manual runs.

Our two Chinese automatic ad-hoc runs are: pircsCw, using short-word segmentation for Chinese texts and pircsCwc, which additionally includes single characters. Both runs are much better than average, but pircsCwc has a slight edge over pircsCw.

In routing a genetic algorithm is used to select suitable subsets of relevant documents for training queries. Out of an initial random population of 15, the best subset (based on average precision) was employed to train the routing queries for the pircsRG0 run. This ith (= 0) population is operated by a probabilistic reproduction and crossover strategy to produce the (i+1)th and evaluated, and this iterates for 6 generations. The best relevant subset of the 6th is used for training our queries for pircsRG6. It performs a few percent better than pircsRG0, and both are well above average.

For the filtering experiment, we use thresholding on the retrieval status values; thresholds are trained based on the utility functions. Results are also good.

1. ENGLISH AD-HOC RETRIEVAL

1.1 Automatic Query Formulation

TREC-5 ad-hoc retrieval requires experimenting with a short form of the topics (#251-300) having only a description section, and a long form that has additionally a narrative section. The collection consists of Tipster Disk 2 (WSJ, AP, ZIFF, FR) and Disk 4 (FR, CR-E, CR-H, FT). Our experiments are designated as pircsAAS and pircsAAL. When query descriptions are short (in this case averaging about 6 content terms per query for pircsAAS - see Fig.1), the users intention becomes vague and ambiguous, and retrieval is notoriously difficult. Short queries lack both term variety (to describe the intended topics) and

Number of	Number of Queries			
Uniq.Terms	AAS	AAL	AM1	AM2
1	1		1	
2	2		4	
3	6		6	2
4	5		8	2
5	7		8	3
6	9	1	5	8
7	6		5	7
8	6	1	6	6
9	1		1	2
10	4	2	3	4
11	1	3	1	4
12		2		5
13		1		3
14		1		
15	1	4	1	1
16		3		
17		1		1
18		1		1
19	1	1	1	
20		1		
21-42		27		1
74		1		
Average	**6.32**	**22.02**	**5.92**	**9.00**

Fig.1: Distribution of Query Sizes for pircsAAS, AAL and pircsAM1, AM2.

term weight indicating importance (because practically all terms are used once). A comparison of query lengths for this set of queries (#251-#300) is given in Fig.1. It is noted that the long-form queries in pircsAAL have nearly 4 times as much text description as the short-form pircsAAS. We recently introduced a method of automatically weighting terms in short queries that appears to enhance the previous TREC-4 retrieval results [Kwok96], and have applied it in TREC-5. Each query term is assigned a weight given by:

$$w = avtf^{\alpha} * ildf/norm$$

where

avtf = average term frequency
= ColFre/DocFre
ildf = inverse log document frequency with cutoff
= 1/log max(cutoff,DocFre)
α = a parameter set to 1.5
norm = a normalizing factor, so that the total term weights for a document sum to 1.

This is a form of 'tf*idf' weight popularized by Salton. The average term frequency avtf is the default within-document term frequency based on observation from the whole collection, and given to a query term when the short query does not provide differentiating clues. The inverse log document frequency idlf (which was called idf in [Kwok96] and used successfully in our routing experiments) attaches less weight to terms with high document frequency, but not so much less as using the idf. At the low frequency end, a cutoff value of 2000 prevents low frequency terms from getting unusually high weights. α is a parameter that allows one to vary the importance of the two factors: avtf and ildf.

We also add term variety to a query in a small scale. Normally terms above a frequency threshold HiFre (set to 60000 in our case) are removed as stopwords. We believe that they still can carry some content, especially in conjunction with other query terms. In TREC-5, we automatically concatenate them with other less frequent query terms to form 2-word phrases, and these have a chance to match with 2-word adjacency phrases that are captured in documents. This experiment is labeled Ad-hoc Automatic Short: pircsAAS.

For long queries we suppress the term variety addition. The avtf*ildf term weight is still applied to those queries with 14 terms or less; otherwise our default weighting using actual observed term frequencies is performed. This experiment is labeled pircsAAL.

Retrieval is done as before in 2-stages [BSAS95,RWJHG95,KwGL95]. The initial stage uses the raw query enhanced by weighting and limited term variety modifications as discussed above, and a ranked list of documents is obtained. The 40 best-ranked subdocuments are then used for feedback and query expansion with 50 terms. The second stage retrieval uses this expanded query to produce the final retrieval ranked list.

1.2 Manual Query Formulation

Manual queries are formed using the short-form topics as basis as in [KwGr96]. In pircsAM1 (Ad-hoc Manual 1), we double weight certain judged query terms as in TREC4 and the operation is simple and not burdensome, averaging 1/2 minute per query. In pircsAM2 we add some associated terms that either come from the narrative section or based on the opinion of the investigator and takes about 2-3 minutes per query. Table 1 also tabulates their query size distributions. pircsAM1 is shorter than pircsAAS because no manual new terms or automatic term pairs are added. pircsAM2 queries have average length of 9. We manually added about 3 terms to pircsAM1 on average. Retrieval is done similarly as before. This will provide a comparison of our automatic query formulation with respect to manual within our PIRCS system.

1.3 Ad-hoc Results and Discussion

The collection on Disk 2&4 are divided into four subcollections with documents segmented into about 550-word chunks ending on a paragraph boundary as in previous TRECs. They are served by a master lexicon of 664,064 unique terms that include 80,031 entries of our semi-automatic 2-word adjacency phrases. This is about 50% more than our previous phrase list of 55,599.

Comparison Within Pircs

In Fig.2 we have tabulated our 4 official ad-hoc retrieval results (shown *) together with their

QryType pircs	1st-stage Retrieval	2nd-stage QryExp Retrieval
	AUTOMATIC	
AAO	**1763/.1397** (%/%) (% / %)	**2279/.1612** (29.3 / 15.3) (% / %)
AAS	**1990/.1605** (%/%) (12.9 / 14.9)	***2335/.1809** (17.3 / 12.7) (2.5 / 12.3)
AAL	**2452/.2082** (%/%) (39.1 / 49.0)	***2941/.2466** (19.9 / 18.4) (29.0 / 53.1)
	MANUAL	
AM1	**1970/.1610** (%/%) (11.7 / 15.2)	***2354/.1856** (19.5 / 15.3) (3.3 / 15.2)
AM2	**2375/.2033** (%/%) (34.7 / 45.5)	***2801/.2298** (17.9 / 13.0) (22.9 / 42.6)

Fig.2: Ad-Hoc Experiments: Relevants Retrieved @1000 / Non-Interpolated Average Precision Results Before and After Query Expansion for the 5 Initial Query Types Averaged over 50 queries.

corresponding 1st-stage retrievals for comparison. Additionally, another pair of experiments labeled pircsAAO is also shown. This is the Ad-hoc Automatic Original retrieval using the raw short queries and pircs' default weighting (ie. not using avtf*ildf or term variety additions). Percentage increases from this base values (%/%) are shown both horizontally (before and after query training and expansion) and vertically (between query types).

The results confirm what has been reported previously in [Kwok96] where the experiments were done with known answers and not blind, and in TREC-4 [KwGr96]. In particular, the 2-stage expanded query retrieval outperforms single stage retrieval on average in all cases, by between 12.7% to 18.4% in precision and 17.3% to 29.3% in relevants retrieved. It is noted that retrieval by the raw short queries with default weighting (pircsAAO) is not recommended. The queries should either be automatically weighted via the avtf*ildf scheme (pircsAAS, 12.3% to 14.9% improvement in precision compared to pircsAAO), manually double weighting some user-selected important terms (pircsAM1, 15.2% improvement), or additionally manually add some associated content

pircs	Number of Queries Better	Equal	Worse
AAS (compared to short topic runs)			
Av.P	32(3)	3	15
rel.ret@100	33(8)	6	11
rel.ret@1000	35(10)	6	9
AAL (compared to long topic runs)			
Av.P	38(11)	9	3
rel.ret@100	36(19)	8	6
rel.ret@1000	41(26)	4	5
AM1 (compared to manual runs)			
Av.P	22(3)	5	23
rel.ret@100	18(8)	7	25
rel.ret@1000	26(8)	2	22
AM2 (compared to manual runs)			
Av.P	22(1)	3	25
rel.ret@100	25(8)	7	18
rel.ret@1000	37(13)	1	12

(.) denotes number equally the best results.

Fig.3: Comparison of Ad-Hoc Results with the Median from All Sites

terms (pircsAM2, over 42.6% to 45.5% improvement). The automatic weighting results for short queries pircsAAS are quite comparable to the manual double-weighting method pircsAM1. However, the best performance is obtained with the original long queries that include the narrative section (pircsAAL, about 49% to 53.1% improvement). Apparently the collective set of terms (average 22.07) provided by the topic composer gives clues to the retrieval system that are lacking in short descriptions.

Also, adding term variety in initial retrieval based on our simple proceudure of paring high frequency terms (that are removed) with existing low/medium frequency terms is not recommended. Removing this operation actually improves our pircsAAS results by about 6% in 2nd stage retrieval.

Comparison with Other Sites

Fig.3 shows how our results compare with respect to the median from all sites. The automatic runs, whether short or long, perform very well. However, the manual runs are about median.

As in previous years we have also calculated the MAXI-system performance by selecting the best results among all sites for each of the 50 queries. Three measures (average precision, relevants retrieved @100 and @1000) are used. Fig.4 shows the comparison of our system to this performance. Our automatic results for long queries achieve between 78.9% to 86.7% of these maximum values. Short query results pircsAAS, like manual pircsAM2, achieve about 54% to 70% of their respective maxima. PircsAM1 performed worse. It is interesting to note that the MAXI-system for short queries actually outperformed that for long queries in two measures: average precision and relevants retrieved @100. Our simple manual manipulation of the queries are not sufficient to achieve good results compared with what are done in other sites.

	Av.P.	RR@100	RR@1000
MAXI (short)	.3312	1413	3338
pircsAAS	.1809 (54.6%	864 61.1%	2335 70.0%)
MAXI (long)	.3124	1343	3392
pircsAAL	.2466 (78.9%	1069 79.6%	2941 86.7%)
MAXI (manual)	.4371	1909	4023
pircsAM1	.1856 (42.5%	848 44.4%	2354 58.5%)
pircsAM2	.2298 (52.6%	1011 53.0%	2801 69.6%)

Fig.4: Comparison of PIRCS's Performance to the MAXI-System

2. CHINESE AD-HOC RETRIEVAL

2.1 Word Segmentation

As is well known, a Chinese written sentence consists of a string of characters between punctuations and without natural word boundaries. A crucial step to Chinese IR is to segment a sentence into meaningul content words. This operation is hard because different adjacent character sequences can be meaningful depending on context, and it is not obvious which sequence to choose. It is usual to consider the longest meaningful character sequence breakups as the correct segmentation of a sentence. For large scale IR however, this may not be fruitful because long words can lead to problems of evaluating partial matchings between query and document words. A number of studies and operational systems for Chinese word segmentation has been done including: [WuTs95,Chie95,LiLY96,PoCr96,NMSU].

Our strategy is to detect common short words of 2 or 3 characters long (with some proper names of 4 characters also), and use exact matching for retrieval. Our purpose is to generate useful features for retrieval, and not necessarily for human consumption. Readability most probably implies good features, but not necessary vice versa. To achieve segmentation for a sentence in a relatively efficient manner, we implement a four-step procedure with a mixture of expert knowledge and statistical processing:

1) facts - lookup on a manually created lexicon list of about 2000 items. Each item is tagged as useful, useless (stopword), numeric, punctuation and a few other codes. Longest match is performed when searching on this lexicon, and this results in breaking a sentence into smaller chunks of texts. This procedure ignores other possible breakup paths. However, we conjecture that its effect may not be too significant because quite often the ambiguities are with character sequences that form stopwords anyway, and also the topics may be sufficiently long that these errors may be remedied in other parts of the query.

2) rules - common usage ad-hoc rules, also manually determined, are then employed to further split the chunks into short words of 2 or 3 characters. Examples of rules that serves to segment text are: a) any two adjacent similar characters XX; b) AX, where A is a small set of special characters; c) a remaining sequence of even number of characters are segmented two by two.

3) filtering - a first pass through the test corpus using steps 1 and 2, and frequency of occurrence is used to threshold the captured short words to extract the most common occurring words.

4) iteration - expand the initial lexicon list in step 1) based on step 3) to about 15000 in our case and re-process the corpus. No additional training collection is needed in these four steps.

Our procedure is like data-mining common short words in the domain of the collection. Naturally no segmentation algorithm is perfect and our case is no exception. Wrong segmentations are made, but for IR we hope the error rates may be tolerable. We evaluate our result with a manual short-word segmentation on the topics and found that we achieve 91.7% in recall and 84.3% in precision. Some examples of our segmentation result with all stopwords kept are given in the Appendix.

The words detected in the above 4-step procedure are used for document and query representation directly in pircsCw. For pircsCwc, we further break up each word into characters and use both words and characters for representation. Because we use exact matching for retrieval, pircsCwc may retrieve documents that may have been missed by pircsCw. 2-stage retrieval is done similar to Section 1.1. However, a more conservative query expansion is done with 30 terms from the best 40 ranking documents of the 1st stage retrieval.

2.2 Ad-hoc Chinese Retrieval Results

We also segment our long Chinese documents into subdocuments of 550 characters long chunks ending on a paragraph boundary as in English. The initial 24,988 Xinhua and 139,801 People's Daily collections produce a total of 231,527 subdocuments. Using our word segmentation procedure, pircsCw processing accumulates a lexicon list of 492,293 entries. With pircsCwc where single characters from identified words are also used for representation, the lexicon size becomes 494,288.

Comparison Within Pircs

Fig.5 tabulates our 2 official Chinese ad-hoc retrieval results (shown *) together with their corresponding 1st-stage retrievals for comparison. Percentage

QryType pircs	1st-stage Retrieval	2nd-stage QryExp Retrieval
AUTOMATIC		
Cw	**1858/.3548** (%/%) (% / %)	***1997/.3989** (7.5 / 12.4) (% / %)
Cwc	**1925/.3726** (%/%) (3.6 / 5.0)	***2019/.4226** (4.9 / 13.4) (1.1 / 5.9)

Fig.5: Chinese Ad-Hoc Experiment: Relevants Retrieved @1000 / Non-Interpolated Average Precision Results Before and After Query Expansion for the 2 Initial Query Types Averaged over 28 queries.

increases from base values (%/%) are shown horizontally (before and after query training and expansion) and vertically (between query types). Results shows that the 2-stage expanded query retrieval works also in the Chinese environment, outperforming initial stage retrieval in both cases, by between 12.4% to 13.4% in precision and 4.9% to 7.5% in relevants retrieved. Using characters in addition to short words for representation in pircsCwc has a slight edge of a few perecent over pircsCw which uses short words alone. pircsCwc only outperforms pircsCw in exactly half of the topics, but the amount is larger. For example, in queries #14, 17, 22, 25, pircsCwc returns average precisions that are better than pircsCw by over 0.15. The reverse is true only on one occasion with query #23.

(a) 死亡 related to 死, 死伤

character shared: 死

(b) 灭绝 related to 灭种

character shared: 灭

(c) 爱滋 unofficial form of 艾滋

character shared: 滋

We can see why use of characters can help achieve better performance in some circumstances. In the box above, example (a) is a word from query #22 meaning

'dead' and can be expressed in related forms as shown. Another example is (b) meaning 'extinct' query #25. If perfect segmentation is performed but alternative forms are used in the query and a document, there will be no matching value between query and document even though they talk about the same concept. However, if the chinese words are broken up into single characters, some character(s) are shared and non-zero matching values result. Chinese characters do carry meaning though often imprecise; they probably lie between alphabets and words in this capacity.

A different situation occurs in topic #14 and shown in example (c) for the different transliteration of 'AIDS'. Again correct word segmentation leads to no matching with many documents that use the official form. However, when single characters are used, the second character happens to be the same in both transliterations and some matching between query and relevant documents is restored. Of course one also has to balance this with wrong matchings due to characters being used in multiple senses.

Comparison with Other Sites

Fig.6 shows that our two official runs perform very well when compared with the median from all sites. In particular, pircsCwc returns the best average precision of 0.4226, and out of 28 queries, 25 queries are better and 1 equals the medium, while 2 are worse. Out of these 25, 8 returns the best.

	Number of Queries		
pircs	**Better**	**Equal**	**Worse**
Cw			
Av.P	22(6)	1	5
rel.ret@100	22(8)	3	3
rel.ret@1000	27(16)	0	1
Cwc			
Av.P	25(8)	1	2
rel.ret@100	23(10)	3	2
rel.ret@1000	27(19)	0	1

Fig.6: Comparison of Chinese Ad-Hoc Results with the Median from All Sites

3. ROUTING EXPERIMENTS

Training Document Selection

Retrieval results can be improved by an intelligent selection of training documents. We found [KwGr94] that short documents and high ranking subdocuments are particularly effective. Short documents can be extracted without the need for a retrieval and they include information that would not be found by the retrieval system. High ranking documents contain information that the system considers close to the query concept.

Selection and weighing of training data became subject of current research in machine learning. Breiman [Brei96] discovered, that bagging improves the performance of CART, a decision tree learning procedure. During bagging the training set is sampled with replacement, thus perturbing the resulting decision tree. After a number of trials the results are added together using a voting procedure. Breiman observes that this procedure is effective for an unstable learning method, where a small change in training data can have a major effect on the classifier.

Schapire et al. [FrSc96] proposed another method called boosting. In boosting the misclassified training cases are given additional weight. This procedure is iterated until there is no more improvement. Quinlan [Quin96] applied both of these methods to the C4.5 decision tree system and found significant improvement for both methods.

Preliminary experiments revealed, that bagging was not helpful for the pircs system as expected, since the pircs network model is based on bayesian statistical retrieval, thus it is a stable learning method. Boosting was beneficial in some cases, but in most cases it was not. However we found that applying a genetic algorithm search to the document selection problem was beneficial.

Genetic Algorithms and IR

Genetic algorithms [Gold89] is a search procedure based on natural selection, the survival of the fittest. The population consists of strings of data of some alphabet, often consisting 1's and 0's, they may be identified as genes. The initial population is generated randomly and the fitness of each individual is

calculated The strings reproduce, creating the next generation. The most common operations used for reproduction are selection, crossover and mutation. At the time of selection higher ranking strings have a greater chance of being selected, this is how the fitness function guides the search. Crossover is the process of combining two strings into one. Mutation causes random changes to the genes. The new population selected forms the next generation. New generations are formed until the fitness stops improving or some other stopping criteria is reached. Genetic algorithms (GA) have been applied to the IR relevence feedback domain by a number of researchers [Chen95] [YaKoRa93].

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
```

Genetic algorithms are known to perform well in many complicated search spaces, however their computational cost is greater then the cost of other common search methods.

When applying GA search to the IR problem the strings may represent the presence or absence of terms, in which case the result is the best subset of terms for the query. Or it can be an encoding of weights for each term. An interesting variation is when they represent Boolean connectors and terms, then the result is a Boolean query [KPBS94]. In our experiment the goal is to find the best training subset, therefore the string represented presence or absence of training documents from the full set for the query. The population was initialized randomly and it also included the string representing all documents. The fitness function used was the TREC evaluation package. The training documents were the top 3000 short (less than 550 words) evaluated documents.

Genetic algorithms often require hundreds or even thousands of generations to reach their maximum fitness. In our case since a complete training and retrieval is required for the fitness function, we limited the number of generations to six and the population to 15. Mutation was not used. To compensate for the limited number of evaluations, we used an elitist selection strategy giving high ranking individuals very high probability of selection. In genetic algorithms this often leads to gene loss and premature convergence to a local maximum. Even with this strategy the fitness value reached at training continued to increase and the limit of six generations was imposed due to time constraints.

	base	**gen 0**	**gen 6**
avg	0.3233	0.3142	0.3402
%chg		-2.81%	5.26%

Fig.7: Comparison of Routing Queries to Baseline.

gen	**0**	**1**	**2**	**3**	**4**	**5**	**6**
avg	.3177	.3343	.3255	.324	.3369	.3389	.3399
%chg		5.23	2.46	1.98	6.04	6.67	6.99

Fig.8 Performance of the best member in each generation.

	pircsRG0			**pircsRG6**		
	>	**=**	**<**	**>**	**=**	**<**
av. prec:	38(3)	3	4	41(3)	3	1
rel_ret @ 100:	34(9)	9(2)	2	39(11)	6(2)	0
rel_ret @ 1000:	34(15)	6(5)	1	38(18)	7(5)	0

(figure in paranthesis is number of queries equaling the best values)

Fig.9: Comparison of Routing Results with Median.

The standard crossover operation used in a GA consist of cutting the string of the two parents at a random point and exchanging one part to create the new offsprings. In our scheme each gene was added probabilistically. Thus if both parents had a 1 in the same position the offspring could have a 0, 1 or 2.

This was done to allow for training documents to have weight.

Two routing queries were submitted. PircsRG0 is the best member of generation 0, pircsRG6 is the best of generation 6. In our tests they both outperformed the baseline query, which was similar to our pircsL query submitted for TREC-4.

3.1 ROUTING RESULTS

Generation 6 improved on the baseline measure but generation 0 did not. See Fig.7.

A closer look at the results however reveals that most of the gain was in one query. [q 207] . The gain for the 39 queries that have 6 or more relevant documents is only 1.2%.

	Precision oriented	Balanced	Recall oriented
Relevant	5808	5808	5808
Retrieved	861	1703	3056
Rel. Retr.	576	977	1465
Mean Prec.	.215	.246	.423
Mean Recall	.043	.091	.162
Total Prec.	.669	.574	.479
Total Recall	.099	.168	.252
Total Score	-279	251	2804

Fig.10: Filtering results.

	Number of Queries		
	>	=	<
Precision oriented	13(5)	28(15)	8
Balanced	24(12)	18(12)	7
Recall oriented	28(11)	10(3)	11

(figure in paranthesis is number of queries equaling the best values)

Fig.11: Comparison of Filtering Results with Median.

Fig.8 shows that there is an upward trend as the generations advance. These runs were run with query expansion factor of 180 as opposed to 300 on the official run. Fig.9 compares our routing results to the median from all sites.

4. FILTERING EXPERIMENTS

The filtering track experiment used the same approach that we used for TREC-4. We find the point where the maximum occurs for the utility function during a retrospective retrieval. If there is more than one maxima we use the later one. The RSV for this point is used for the cutoff value on the test collection. The cutoff values were extrapolated from a retrieval on the WSJ and AP collections from disk 2. Since that was a retrospective retrieval the cutoff points may have been too conservative. In the future we will consider other possible predictors too, such as rank and number of relevants in the training document. It is also possible that using more data that just disk2 will result in more accurate prediction. The results are summarized in the Fig.10 and 11.

5. CONCLUSION

Our PIRCS system has been extended to handle Chinese documents (GuoBiao coded). We have also implemented a Chinese word segmentation algorithm for IR. Otherwise, processing is similar as in English. It appears that except for manual ad-hoc runs which we did not expand too much effort, all the automatic ad-hoc (including Chinese), routing and filtering runs are much better than the overall average from all sites. We attribute this to the highly effective PIRCS retrieval engine and its learning capabilities which have consistently been demonstrated to give exemplary performance.

ACKNOWLEDGMENTS

This work was partially supported by a PSC-CUNY grant and a Tipster program grant from DARPA . The following are especially acknowledged for their effort in developing the Chinese dictionary and part of the programs for Chinese processing: Yen Yen Lee, Eric Rong, Jun Wang and Mark Young.

REFERENCES

[BSAS95] Buckley, C, Salton, G, Allan, J & Singhal (1995). Automatic query expansion using SMART: TREC 3. In: Overview of the Third Text REtrieval Conf. (TREC-3). Harman, D.K.(ed.) NIST SP 500-225; pp.69-80.

[Brei96] Breiman, L (1996). Bagging predictors, Machine Learning, to be published.

[Chen95] Chen, Hsinchun (1995). Machine Learning for information Retrieval: Neural Networks, Symbolic Learning, and Genetic Algorithms, JASIS 46(3) 194-216.

[Chie95] Chien, L.F (1995). Fast and quasi-natural language search for gigabytes of Chinese texts. In: Proc. 18th ACM SIGIR Conf. on R&D in IR. Fox, E., Ingwersen, P. & Fidel, R. (eds.) ACM:NY, NY. pp.112-120.

[FrSc96] Freund Y, Schapire R,E (1996). A decision theoretic generalization of on-line learning and an application to boosting. 1996 , ("http:/www.research.att.com /orgs/ssr/people/schapire"}

[Gold89] Goldberg, D. E. Genetic Algorithms in Search Optimization & Machne Learning. Addison Wesley 1989.

[KPBS94] Kraft, D.H, Petry, F.B, Buckles, B.P, Sadasivan, T (1994). The use of genetic Programming to Build Queries for Information Retrieval. IEEE Symposium on Evolutionary Computation, Orlando, 1994.

[KwGL95] Kwok, K.L., Grunfeld, L. TREC-3 ad-hoc, routing retrieval and thresholding experiments using PIRCS. (1995). In: The Overview of the Third REtrieval Conference TREC-3. Harman, D.K. (ed.) NIST SP 500-225: Gaithersberg, MD. pp.247-255.

[KwGr94] Kwok K.L, Grunfeld L (1994). Learning from relevant documents in large scale routing retrieval. In Proc ARPA Human Language Technology
Workshop,Mar 8-11,1994 Plainsboro NJ.

[KwGr96] 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.

[Kwok96] Kwok, K.L (1996). A new method of weighting query terms for ad-hoc retrieval. Proc. 19th Annual Intl. ACM SIGIR Conf. on R&D in IR. pp.187-195.

[LiLY96] Liang, T., Lee, S.Y & Yang W.P (1996). Optimal weight assignment for a Chinese signature file. Info. Proc. Mgmnt. 2:227-237.

[NMSU] 'Algorithm of Chinese segmenter' from CRL, New Mexico State University.

[PoCr96] Ponte, J & Croft, W.B (1996). USeg: a retargetable word segmentation procedure for information retrieval. In: Symposium on document analysis and information retrieval (SDAIR '96).

[Quin96] Quinlan, J.R. Boosting, Bagging and C4.5, Draft of paper to be given at AAAI 96 ("ftp:/ftp. cs.su.oz.au/pub/ml/q.aaai96.ps")

[RWJHG95] Robertson, S.E, Walker, S, Jones, S, Hancock-Beauliew, M.M & Gatford, M (1995). Okapi at TREC-3. In: The Overview of the Third REtrieval Conference TREC-3. Harman, D.K. (ed.) NIST SP 500-225: Gaithersberg, MD. pp.109-126.

[WuTs95] Wu, Z & Tseng, G (1995). ACTS: An automatic Chinese text segmentation system for full text retrieval. J. ASIS, 46:83-96.

[YaKoRa93] Yang, J. Korfhage, R.R. Rasmussen, E (1993). Query improvement in information retrieval using genetic algorithm:a report on the experiments of the TREC project. In Text Retrieval Conference (TREC-1) pp 31-58, Gaithesburg, MD 1993.

Appendix

Query 002: Original Chinese

中共对于中国统一的立场
中国，一国两制，台湾，和平统一，经贸合作，两岸关系，
科技、文化交流

相关文件必须提到中共如何经由实现一国两制来达到台湾
与大陆统一的目的.如果文件只是外国政府重申支持中共对
台湾拥有主权或提到中共与其他国家之经贸、科技、文化交流，
则为不相关文件，

Segmented Result:
中共 | 对于 | 中国 | 统一 | 的 | 立场 | 中国 | 一国 | 两 | 制 |

台湾 | 和平 | 统一 | 经贸 | 合作 | 两 | 岸 | 关系 | 科技 | 、 |

文化 | 交流 | 相关文件 | 必须 | 提到 | 中共 | 如何 | 经由 | 实现 | 一国 |

两 | 制 | 来 | 达 | 到 | 台湾 | 与 | 大陆 | 统一 | 的 |

目的 | 如果 | 文件 | 只 | 是 | 外国 | 政府 | 重申 | 支持 | 中共 |

对 | 台湾 | 拥有 | 主权 | 或 | 提到 | 中共 | 与其 | 他 | 国家 |

之 | 经贸 | 、 | 科技 | 、 | 文化 | 交流 | 则 | 为 | 不 | 相关文件 |

Query 014: Original Chinese

中国的爱滋病例
中国,云南,爱滋病,HIV,高危险群患者,注射器,病毒

相关文件应当包括中国那些地区的爱滋病例最多,爱滋病毒在中国是如何传播
的,以及中国政府如何监测爱滋病并控制它的传染.

Segmented Result:
中国 | 的 | 爱滋 | 病例 | 中国 | 云南 | 爱滋病 | ,HIV | 高 | 危险 |

群 | 患者 | 注射 | 器 | 病毒 | 相关文件 | 应 | 当 | 包括 | 中国 |

那些 | 地区 | 的 | 爱滋 | 病例 | 最 | 多 | 爱滋 | 病毒 | 在 |

中国 | 是 | 如何 | 传播 | 的 | 以及 | 中国 | 政府 | 如何 | 监测 |

爱滋病 | 并 | 控制 | 它 | 的 | 传染 |

'^' denotes missing separator; '-' dentoes spurious separator