MDS TREC6 Report

Michael Fuller Martin Kaszkiel Chien Leng Ng Phil Vines Ross Wilkinson
Justin Zobel

Department of Computer Science
RMIT, GPO Box 2476V
Melbourne VIC3001, Australia
msf,cln,martin,vines,ross,jz@mds.rmit.edu.au

1 Introduction

This year the MDS group has participated in the ad hoc task, the Chinese task, the speech track, and the interactive track. It is our first year of participation in the speech and interactive tracks. We found the participation in both of these tracks of great benefit and interest.

2 Full Description of Techniques

In this section of the paper we will give as complete a description as we can of our methodology. We do so by describing the following: term definition, casefolding, stopping, and stemming. This defines the terms that we use. We then give the formula used for matching. After this we give exact descriptions of how we carry out passage retrieval, term expansion, and combination.

A term is a sequence of characters chosen from the alphabet {a–z,A–Z,0–9}. The sequence has a maximum length of 256 but if the string consists solely of numbers a maximum length of 4 applies. All other characters are treated as term delimiters.

To casefold, all uppercase letters are converted to their lowercase equivalents.

To stop, we remove all terms that are in the list given in the appendix.

Terms are stemmed as given in the Lovin's algorithm[4].

Passages are formed by sequences of words that are 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550 or 600 words long. Passages may commence at any 25 word interval. Every such passage is then treated as a document, and matched according to the appropriate retrieval formula.

To match pieces of text we used the same formula for all experiments, a reliable form of the cosine measure:

$$cos(q,d) = \frac{\sum_{t \in q \wedge d}(w_{q,t} \cdot w_{d,t})}{\sqrt{\left(\sum_{t \in q}(w_{q,t})^2 \cdot \sum_{t \in d}(w_{d,t})^2\right)}}$$

with weights that have been shown to be robust and give good retrieval performance [2]: $w_{q,t} = log(N/f_t) + 1$ and $w_{d,t} = log(f_{d,t} + 1)$ where $f_{x,t}$ is the frequency of t in x, N is the number of documents in the collection, and f_t is the number of documents containing t.

To expand query, we first evaluate the original query against the database of documents or passages. The top N documents as determined by the cosine formula are then obtained. The top M terms were determined by using the formula:

$$\frac{Freq.\_in\_top\_N\_docs}{log(K + Freq.\_in\_all\_docs)}$$

Two ranked lists were combined by summing normalized scores. Scores were normalized by ensuring the top-ranked document had a score of 1.0 for each list. Thus the score becomes

$$\alpha \frac{Score1}{Max_{Score1}} + (1-\alpha)\frac{Score2}{Max_{Score2}}$$

Unless otherwise mentioned, $\alpha = 0.5$.

3 Ad Hoc Task

Our main experiments this year have been to do a comprehensive factor analysis of most of the main contributors to successful automatic vector space retrieval. We look at stopping, stemming, passage retrieval, term expansion, methods of combination, and query length. Our methods of passage determination and methods of combination have some novel aspects but our main interest is to report on how all of the above factors interact. We performed no experiments on matching formula, or the use of adjacency information, such as phrases.

Term Experiments

Our first set of experiments were carried out on the TREC5 dataset with TREC5 queries. In these experiments we looked solely at the term definition. We built the TREC5 database 7 times where terms were defined as:

Base All valid strings (up to 256 characters long and not an SGML tag.)

Casefold Each string is converted to solely lower case letters and numbers.

Stop w/0 c.f. Base case with stop words removed.

Stop Stop words removed after casefolding.

Stem w/o c.f. Stem each string in the base case

Stem Stem after casefolding.

Stop & Stem Casefold, stop, and then stem (Standard processing.)

Having built the database, the queries were processed in the same way. The results for the description queries and full queries are given in Tables 1 and 2. The title only runs were very low but gave similar results. As can be seen, there are no surprises, and stopping and stemming English text is again shown to be appropriate.

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.22	0.19	0.18	0.07	0.079
Casefold	0.23	0.21	0.18	0.08	0.087
Stop w/o c.f.	0.24	0.22	0.19	0.08	0.095
Stop	0.25	0.23	0.20	0.08	0.098
Stem w/o c.f.	0.23	0.20	0.17	0.08	0.087
Stem	0.25	0.22	0.19	0.09	0.094
Stop & Stem	0.25	0.22	0.21	0.09	0.106

Table 1: Baseline Experiments – precision for description queries 251–300

Passage and Expansion Experiments

Our next set of experiments were performed on TREC6 data using TREC6 queries. In these experiments we assume all text is stopped and stemmed and now look at the use of passages and term expansion to improve retrieval performance. These experiments were performed using the description queries. We first examine replacing the query with M terms selected from the top N documents. These expanded queries may have the original terms in them but there is no guarantee. Previous experiments suggest that one should select between 10 and 100 terms from between 10 and 100 documents [6]. In these experiments we fixed the number of terms to be 40, and used either 15 or 30 documents. K is set to 1. The next experiments looked at selecting the best document based on

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.22	0.19	0.18	0.07	0.079
Casefold	0.23	0.21	0.18	0.08	0.087
Stop w/o c.f.	0.24	0.22	0.19	0.08	0.095
Stop	0.25	0.23	0.20	0.08	0.187
Stem w/o c.f.	0.42	0.34	0.27	0.12	0.163
Stem	0.42	0.38	0.32	0.12	0.178
Stop & Stem	0.40	0.39	0.32	0.12	0.192

Table 2: Baseline Experiments – precision for full queries 251–300

the best N word passage in the document. Last year a very wide range of sizes were investigated, so this year only 100, 150, and 200 word passages are reported (a wider range was investigated).

Now it is possible to expand queries using the ranking obtained from either document ranking or passage ranking. Given the improved performance of passage retrieval it may be desirable to use passages to select new terms and then rank passages using these new terms. As we see in Table 3 there is no gain obtained from this method.

Of course it is possible to merge the original query with the expanded query. We show the result of merging the original query and the expanded query taken from the top 30 passages and then matching against 150 word paragraphs. Again the result is not as good as the original passage query.

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.29	0.25	0.19	0.08	0.105
Expand-15	0.31	0.25	0.21	0.07	0.106
Expand-30	0.27	0.22	0.20	0.07	0.101
Passage-100	0.32	0.28	0.25	0.10	0.174
Passage-150	0.34	0.31	0.25	0.09	0.176
Passage-200	0.33	0.30	0.25	0.09	0.167
Expand-15-150	0.28	0.26	0.22	0.08	0.134
Expand-30-150	0.29	0.27	0.24	0.09	0.140
Merge-30-150	0.29	0.29	0.26	0.09	0.159

Table 3: Passage and Expansion Experiments – precision for description queries 301–350

It was subsequently determined that the creators of the description field assumed that the title field would be used jointly. Thus the results for the corresponding runs using title and description queries are shown in Table 4. As can be seen, there is a substantial improvement in the baseline—it is also an improvement over the equivalent baseline for title only shown in Table 8. However the big gains occur using passage retrieval. These gains are huge—about 79%. There is no gain available through expansion.

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.33	0.28	0.25	0.10	0.136
Expand-15	0.30	0.28	0.23	0.08	0.110
Expand-30	0.30	0.27	0.23	0.08	0.111
Passage-100	0.45	0.40	0.33	0.12	0.242
Passage-150	0.50	0.43	0.34	0.12	0.243
Passage-200	0.46	0.42	0.34	0.12	0.238
Expand-15-150	0.36	0.33	0.26	0.09	0.157
Expand-30-150	0.32	0.30	0.28	0.10	0.161
Merge-30-150	0.32	0.30	0.30	0.11	0.194

Table 4: Passage and Expansion Experiments – precision for title+description queries 301–350

Combination Experiments

While some methods that we have described seem to give little improvement, they do provide additional evidence that may be used in combination with other factors. However, combination of evidence works well when there are different factors that give roughly equivalent performance. We consider combining 4 different pieces of evidence—the baseline run, the expansion using 30 documents, the passage run using 150 word passages, and and expansion using 30 passages of 150 words. The results are shown in Table 5.

As can be seen only small gains can be obtained above the top performing run, the passage run. By comparison, for the same set of experiments on disk2 using TREC5 queries, we found that passages gave a 20% improvement on the baseline, but that combining passages with passages on expanded queries gave another 20% gain, and that by combining all forms of evidence gave a further 10% gain. This highlights that combination of evidence works at its best when the forms of evidence are roughly comparable in quality.

The corresponding runs using title and description fields together as the query are shown in Table 6. Due to the large imbalance between the performance of the passage runs and all other runs there is no improvement over the passage runs.

Experiment	5docs	10docs	20docs	200docs	Average
Base+Exp-30	0.28	0.26	0.22	0.09	0.125
Passage-150+Exp-30	0.36	0.28	0.27	0.10	0.180
Base+Expand-150-30	0.36	0.32	0.28	0.09	0.164
Passage-150+Exp-150-30	0.34	0.32	0.28	0.10	0.170
Combine All - mds601	0.35	0.30	0.24	0.09	0.138

Table 5: Combination Experiments – precision for description queries 301–350

In Tables 7 and 8 we show the corresponding runs for full queries and title queries. The expansion algorithm was still under development at the time of these runs and is clearly problematic, particularly for passages. Unlike the description queries and title queries, there were much smaller gains made by passages for full queries.

Conclusions

Our experiments have shown yet again that stopping and stemming work well for English. We have seen that using passages gives very good gains. We have seen that expansion does not improve performance on its own. It is robust to a in terms of the numbers of document and number of terms selected, but it may be quite sensitive to the weighting formula for selecting terms. more work is needed here. When combined with an original query there is a consistent performance improvement. However if some portion of the evidence to be combined is of poor quality this can hurt performance.

Experiment	5docs	10docs	20docs	200docs	Average
Base+Exp-30	0.36	0.30	0.26	0.11	0.147
Passage-150+Exp-30	0.44	0.37	0.32	0.12	0.214
Base+Expand-150-30	0.40	0.37	0.32	0.12	0.188
Passage-150+Exp-150-30	0.40	0.38	0.33	0.12	0.229
Combine All	0.40	0.37	0.32	0.12	0.204

Table 6: Combination Experiments – precision for title+description queries 301–350

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.45	0.38	0.32	0.13	0.196
Exp-30	0.32	0.29	0.28	0.10	0.142
Passage-150	0.45	0.38	0.32	0.13	0.200
Expand-150-30	0.27	0.25	0.21	0.08	0.090
Combine All - mds602	0.42	0.37	0.32	0.13	0.230

Table 7: Combination Experiments – precision for full queries 301–350

4 Chinese Retrieval

First Experiments

For our baseline experiments, we tried indexing each document on characters, words, and bigrams. For character indexing we treated each document as a series of distinct characters, and used these to build our index. Queries consisted of all the characters in the document. For word indexing we parsed documents into words using an online dictionary kindly made available by the Berkeley group. We used *greedy parsing*, in which we matched the longest entry in the dictionary at any point. Although this is not the best strategy, it works reasonably well. For bigrams we used every possible pair of adjacent characters that did not include punctuation. For example a sequence of 7 Chinese characters *abc.def* would generate pairs *ab, bc, de, ef* but not *c., .d, or cd*. We used the *mg* system for our experiments. In calculating query/document similarity we used the standard cosine measure. Results for these experiments are shown in Table 9.

Mutual Information

In this experiment we were interested in seeing how well we could segment the text into words without the use of a dictionary, but rather relying on the mutual information contained in the corpus. This

Experiment	5docs	10docs	20docs	200docs	Average
Base	0.28	0.25	0.20	0.09	0.127
Exp-30	0.24	0.22	0.19	0.07	0.087
Passage-150	0.40	0.33	0.28	0.10	0.168
Expand-150-30	0.22	0.21	0.18	0.05	0.076
Combine All - mds603	0.31	0.29	0.26	0.10	0.157

Table 8: Combination Experiments – precision for title queries 301–350

Experiment	5docs	10docs	20docs	200docs	Average
Characters	0.73	0.72	0.69	0.33	0.455
Words	0.78	0.76	0.74	0.35	0.498
Bigrams	0.74	0.72	0.68	0.35	0.494

Table 9: Baseline Experiments – precision for queries 29–54

is similar to the approach used by [3], based on the mutual information idea proposed by [5]. Mutual Information is defined as $I(x,y) = log_2 \frac{p(x,y)}{p(x)p(y)}$ where $p(x)$ and $p(y)$ is the probability of occurrence of characters x and y respectively in the corpus, while $p(x,y)$ is the probability of the two characters occurring together. If the two characters are related the value of $I(x,y)$ will be high, suggesting that the bigram xy may in fact be a word. Thus a two step method is needed. First frequencies are determined from the source text, and bigrams with a mutual information value above a threshold are presumed to be words. We used $I(xy) = 7$ as the threshold, the same as [5]. Second the text is parsed, sentence at a time using the words so gathered. Our results gave an average precision of 0.302 on queries 1–28, which is inferior to the 0.374 figure reported by [3].

Experiment	5docs	10docs	20docs	200docs	Average
Mutual Information	0.76	0.75	0.74	0.35	0.462

Table 10: Mutual Information – precision for queries 1–28

Expansion of characters and words

In these experiments we implemented a simple form of feedback by using the top 30 documents returned for each query in experiments one and two. We did a frequency count of the terms in these documents and ranked them using $\frac{f_i}{log_2(df_i+20)}$ where f_i is the frequency of the ith term the top 30 documents, and df_i is the document frequency. This effectively selects relatively infrequent words, but avoids the undue influence from very rare words by adding 20 to the denominator. Results for these expanded queries are shown in Table 11. Only in the case of bi-grams did the expansion give any improvement over the equivalent baseline. We think that further refinement of our expansion mechanism is required, and are currently investigating this.

Experiment	5docs	10docs	20docs	200docs	Average
Characters	0.73	0.72	0.69	0.33	0.455
Exp-Characters	0.68	0.65	0.54	0.20	0.239
Words	0.78	0.76	0.74	0.35	0.498
Exp-Words	0.82	0.80	0.73	0.31	0.437
Bigrams	0.74	0.72	0.68	0.35	0.494
Exp-Bigrams	0.83	0.77	0.73	0.34	0.506

Table 11: Expanded Queries – precision for queries 29–54

Combination of Evidence

Following the hypothesis that combination of evidence from a number of sources usually improves results, we decided to try combining some of the results of our previous experiments as a final step. We tried values of 0.33, 0.5, and 0.67 for α to test the sensitivity of combination. As well as combining results from two experiments, we found that by again combining the results of these runs we could further improve performance. Our best run on the queries 1-26 was the result of first combining bigrams and expanded bigrams, then combining words and expanded words, both with $\alpha = 0.5$, then combining the results of these two runs using $\alpha = 0.33$. Results of some other combinations are shown in Table 12. Clearly combination of evidence improves retrieval effectiveness.

Exp.	Method A	α	Method B	5docs	10 docs	20docs	200docs	Average
	Bigrams	0.33	Exp-Bigrams	0.87	0.81	0.76	0.37	0.546
mds607	Bigrams	0.5	Exp-Bigrams	0.88	0.82	0.76	0.37	0.547
	Bigrams	0.67	Exp-Bigrams	0.84	0.80	0.74	0.37	0.539
	Words	0.33	Exp-Words	0.88	0.84	0.79	0.36	0.528
	Words	0.5	Exp-Words	0.86	0.83	0.80	0.36	0.535
	Words	0.67	Exp-Words	0.86	0.83	0.77	0.36	0.536
	Bi/BiX-0.5	0.5	Word/WordX-0.5	0.88	0.84	0.78	0.37	0.560
mds608	Bi/BiX-0.33	0.67	Word/WordX-0.67	0.87	0.82	0.77	0.38	0.560
mds609	Bigrams	0.50	Word Exp	0.85	0.83	0.77	0.37	0.548

Table 12: Statistics of Chinese text collections – precision for queries 29–54

Conclusions

We have seen that using bigrams as a basis of retrieval is quite effective and provides a simple low-cost solution to effective Chinese retrieval. Term expansion and combination of evidence can be used to improve retrieval effectiveness, however we need to do more work to understand how to apply them well in the context of Chinese IR.

5 Interactive Retrieval

Goals

The high-level goal of the Interactive Track in TREC-6 is the investigation of searching as an interactive task by examining the process as well as the outcome. In particular, the task set for the interactive track was the investigation of multi-aspect queries. The RMIT interest in the TREC-6 interactive track was twofold. The primary interest was to develop an interactive system that would use feedback from user to user to cluster and re-rank relevant documents. The secondary interest was to develop an interface to aid users to structure and organize candidate documents in order to compose answers to information needs.

Our goals in this project were therefore three-fold. Firstly, to develop an interactive system that could be paired experimentally with the ZPRISE control system. Secondly, to use that system to interactively cluster and re-order candidate documents to better allow the user to identify documents of interest. Thirdly, to extend that system to permit the user to organize candidate documents or passages such that a structured answer to their information need might be formed.

RMIT WWW/MG system

This was our first attempt to be involved in the TREC interactive track, and, unfortunately, we were only able to complete a partial WWW-based prototype for the experimental phase. Per the interactive track model, we ran four subjects on the ZPRISE control system, and on an WWW/MG-based system. These results can best be considered a comparison of ZPRISE with interactive MG, rather than an evaluation of a new experimental system. Work is continuing on the implementation of the full interactive prototype.

The WWW/MG prototype system was intended to mimic the functionality of the control system. Similarly to the control, it allowed users to issue free text queries, resulting in a list of candidate documents (matching documents identified by MG), but with the addition of the ability to identify and label each aspect found within candidate documents from within the document presentation interface. This can be seen in Figure 1. The prototype was implemented by building a MG database from the FT test collection. The HTML interface was dynamically generated by a set of CGI Perl and JavaScript programs, allowing users' queries to be converted into an MG query, the results of which were then parsed and a synopsis converted to HTML. Requests to view specific documents were also passed to MG, and the resultant document text displayed as HTML. Additional tasks such as tracking user judgments, user-determined topic aspects, and ancillary logging were also performed.

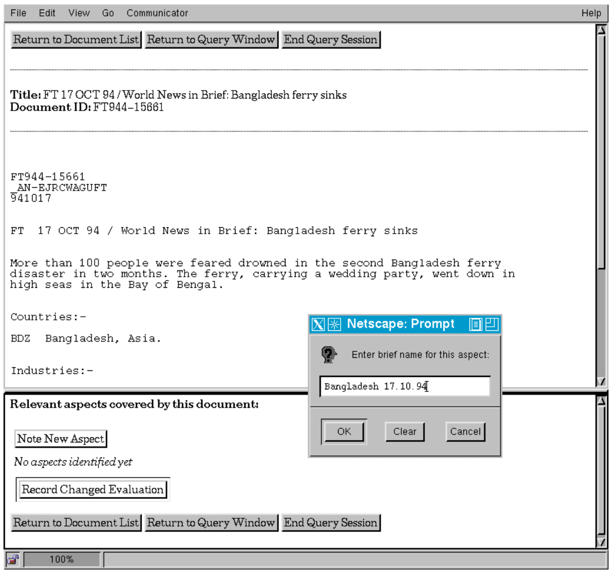

Figure 1: Document Viewing Interface

As the prototype system comprised a relatively simple interface to an existing retrieval system and was intentionally similar to the control system in design, it is not surprising that the experimental results were close to the control results. For the control systems, subjects identified aspects at an average precision of 0.779 and an average aspectual recall of 0.499, in, on average, approximately 17 of the 20 minutes available. For the experimental system, subjects identified aspects at an average precision of 0.805 and an average aspectual recall of 0.466, in, on average, approximately 17 of the 20 minutes available.

As part of our analysis of the results we considered two questions: what agreement was there on aspectual relevance, and what agreement was there on what the aspects were. There was a great deal of disagreement between experimental subjects and the NIST assessors (Table 13). Of the documents considered relevant by the pool of experimental subjects, over half were rejected by the NIST assessors as irrelevant. Conversely, numerous documents considered relevant by either or both the NIST assessors and subjects from other sites were viewed by RMIT experimental subjects and rejected as irrelevant. This occurred both when using the WWW/MG system, and when using the ZPRISE control system. From a local perspective, the same phenomenon occurred: on several occasions, for the same queries, the same documents were viewed by separate subjects, only for one to decide the document contained relevant aspects, and the other that it did not.

Query	Subject Relevant	NIST aspects	NIST Relevant	NIST Irrelevant
303i	22	7	5	17
307i	103	23	54	49
322i	63	9	21	42
326i	47	9	31	16
339i	28	10	7	21
347i	86	26	43	43
Totals:	349	84	161	188

Table 13: Relevant and irrelevant documents

Tables 14 and 15 provide a comparison of the aspects found (or not found) by WWW/MG searchers for queries 303i and 339i.

Table 14 reveals a marked difference between the aspects nominated by the NIST assessors and those listed by the WWW/MG experimental subjects. This is evidenced by the aspects to be found in document FT924-286: the NIST assessor discovered four distinct aspects of the topic, whereas the RMIT subject indicated it covered just a single aspect, "black hole study". Other interesting differences include document FT934-54181, which the NIST assessor described as representing the aspect "generally good, better, better than expected results", but which was viewed and discarded as not relevant by the RMIT subject; and document FT941-17652 for which the reverse was true.

Table 15 has two interesting characteristics. One is that experimental subjects are not accurate readers! Searcher s2 noted that the document was relevant to two aspects, but missed three others; whilst searcher s4 noted three aspects, but not two others. (Searcher s2 had not previously seen documents containing relevant aspects; searcher s4 had previously seen and saved a document containing aspect 4, but not aspect 6.) The other is the difficulty of defining relevance: subjects indicated that the documents FT933-5910 and FT942-17255 contained aspects relevant to the topic, but were rejected as irrelevant by the assessor.

We have seen that in Query 303i there is a very significant difference in how the "intellectual space" is divided into aspects. The consequence is that there can be a very large difference in precision and aspectual recall as a result. In Query 339i there is less issue of the nature of the aspects, but even if subjects agree on relevance, they did not recognise the presence of some aspects, affecting their "personal aspectual recall", but not the system evaluated performance.

	NIST aspects
1	has inspired new cosmological theories
2	study of gravitational lenses
3	more precise estimate of scale, size, and age of universe
4	picture of more distant galaxies/objects
5	generally good, better, better than expected results
6	contradicted existing cosmological theories
7	supported existing cosmological theories

	RMIT aspects
a	black hole study
b	images
c	Hubble,wonder image, universe theory
d	origin of universe
	hubble, fall of contemporary cosmology theories

NIST aspects	Document	RMIT aspect
1, 2, 3, 4	FT924-286	a
5	FT944-15661	b
5	FT934-54181	—
—	FT941-17652	c
3, 4, 5, 6, 7	FT941-17652	d

Table 14: Query 303i: NIST and RMIT aspects

Concluding remarks

We are pleased to have taken part in this year's interactive track. It has raised some philosophical and methodological issues of interest and concern. We plan now to continue development of an interactive prototype, utilizing clustering and feedback to group and re-order the pool of candidate documents dynamically, leading to a system designed to support analysis and synthesis of structured answers to information needs.

NIST aspects		RMIT aspects	
1	Alcav	1	alcar
2	pivacetam	2	piracetam
			Piracetam of UCB Belgium
3	oxiracetam	3	oxiracetam
			oxiracetam of SmithKline Beecham
4	tacrine - Cognex	4	cognex
			Warner-Lambert, Cognex,
			Good evaluator of effect of drug to alz
5	physostigmine	5	physostigmine of Forest Lab US
6	Aviva	6	aviva
7	velnacrine - Mentane	7	velnacrine (Mentane) of Hoechst Germany
8	selegiline (Eldepryl)	8	selegiline of Sandoz Switz
9	Zofran (ondansetron)	9	(not found)
10	denbuflline	10	denbufylline
		a	silicon intake to prevent alz
		b	tacrine

NIST aspects	Document	RMIT aspect
1, 3, 10	FT922-1565	1, 3, 10
2, 3, 4, 5, 6	FT922-715	searcher s2: 2, 6
		searcher s4: 2, 3, 5
4	FT924-8306	4
4	FT931-2434	4
4, 7, 8	FT932-7262	7, 8
—	FT933-5910	a
—	FT942-17255	b

Table 15: Query 339i: NIST and RMIT aspects

6 Speech Retrieval

We participated in the full SDR track. Our speech experiments explored the use of phoneme sequences as matching units instead of words. Phonemes were extracted from the speech tracks and triphones created to perform retrieval. For comparison, the transcripts were also translated to phoneme sequences. The translation to phonemes used the Ainsworth algorithm [1].

MG, developed at RMIT and the University of Melbourne, is the retrieval engine used. The recognition engine used is HTK which is developed at Cambridge.

The reference experiment consisted of retrieval based on the textual documents provided (LTT). The first baseline experiment used the transcribed documents provided by IBM (SRT). For both experiments, documents and queries were stemmed and casefolded. In addition, the queries were stopped. The average length of the queries was 5 words. Results for the reference run is shown as MDS612 and the first baseline run is shown as MDS613 in Table 16.

The second baseline experiment investigated the performance of phoneme retrieval when recognition is assumed to be perfect. The documents from the reference run were translated to phonemes and then transformed to triphones before indexing. The queries were not stopped prior to the transformation. Results for this run are shown as MDS615.

We used triphones because it has a higher noise tolerance than words. In addition, word boundaries becomes less important.

A simple phoneme model was built for 61 phones. The phoneme model was trained using the speech training documents provided. The training data did not contain explicit details about phone boundaries within and between words. Explicit segment and section times were provided for the training process instead. We had about 1400 test and 500 training documents. For our experiments, most of the longer training and test documents were not used. Initial recognition results indicate about 16% recognition accuracy of the phonemes. A reason for such poor performance may be due to the lack of information on phoneme boundaries. There may also be an error in the training which we have yet to find.

Post-trec experiments included the addition of another 86 documents from the i-disk (i960606, i960610, i960611). These were excluded because of difficulties in processing the larger speech documents. The reference experiment was repeated as well as the second baseline run. The results are shown as ref-new-s and ref-new-ph-l respectively. The stoplist may have been too aggressive for this document collection. The results of queries which were not stopped are shown as ref-new-l. Some of the stop terms were useful in retrieving relevant documents. This was indicated by an improvement in mean reciprocal.

The transcribed documents (SRT) were translated and transformed to triphones. Results are shown as srt-new-ph-l. The effects of an imperfect recogniser contributed to the retrieval of many irrelevant documents.

For triphone retrieval, important textual terms were lost but retrieval was possible using part of the terms at triphone level. It was found that unimportant terms at the word level became important at the phoneme level. For example, the term classic is translated to klasik which is transformed to "... kls las asi sik ...". The triphone "las" helps retreive the relevant document.

Experiments have shown that without being given more explicit boundary information, the recognition result can be improved to approximately 30%. This was accomplished by segmenting the training documents into smaller segments of about 30 seconds. However, this recognition model resulted in degraded retrieval effectiveness. This could mean the model is not recognising at all or the test documents have to be segmented as well.

	Mean Rank	Mean Reciprocal
MDS612	5.31	0.7036
ref-new-s	5.48	0.6899
ref-new-l	13.10	0.7238
MDS613	10.11	0.5207
MDS614	229.20	0.0046
MDS615	8.71	0.7316
ref-new-ph-l	11.47	0.7340
srt-new-ph-l	23.49	0.5472

Table 16: Full Speech Experiments

References

[1] William A. Ainsworth. A system for converting English text into speech. *IEEE Transactions on Audio and Electroacoustics*, AU-21(3):288–290, Jun 1973.

[2] C. Buckley, G. Salton, and J. Allan. The effect of adding relevance information in a relevance feedback environment. In W.B. Croft and C.J. van Rijsbergen, editors, *Proceedings of the 17th Annual International Conference on Research and Development in Information Retrieval*, pages 292–300, Dublin, Ireland, July 3–6 1994. Springer–Verlag.

[3] A. Chen, J. He, L. Xu, F.C. Gey, and J. Meggs. Chinese text retrieval without using a dictionary. In N. Belkin, D. Narasimilau, and P. Willett, editors, *Proceedings of the 20th Annual International Conference on Research and Development in Information Retrieval*, pages 42–49, Philadelphia, U.S.A., August 27–31 1997. ACM.

[4] J.B. Lovins. Development of a stemming algorithm. *Mechanical Translation and Computation*, 11(1–2):22–31, 1968.

[5] R. Sproat and C.L. Shih. A statistical method for finding word boundaries in Chinese text. *Computer Processing of Chinese and Oriental Languages*, 4(4):336–351, 1990.

[6] R. Wilkinson. Using combination of evidence for term expansion. In J. Furner and D.J. Harper, editors, *Proceedings of the BCS-IRSG 19th Annual Colloquium on IR Research*, Aberdeen, Sotland, 1997. Springer-Verlag.

Appendix A – Stopwords

0 1 2 3 4 5 6 7 8 9 a about above across after afterwards again against all almost along already also although always am among amongst an and another any anybody anyhow anyone anything anywhere ap apart are around as at b be became because become becomes been before beforehand behind being below beside besides best better between beyond both but by c can cannot cant co could d described did do does doing done down during e each eg eight eighth either else elsewhere end ended ending ends enough et etc even evenly ever every everybody everyone everything everywhere ex except f far few fifth first five for four fourth from furthermore g great greater greatest h had has have having he hence her here hereafter hereby herein hereupon hers herself high higher highest him himself his hither how howbeit however i ie if in inasmuch indeed insofar instead into inward is it its itself j just k l large largely last later latest latter latterly least less lest long longer longest m many me meanwhile more moreover most mostly mr mrs much my myself n namely neither never nevertheless newer newest next nine ninth no nobody non none noone nor not nothing now nowhere o of off often oh old older oldest on once one ones only onto or other others otherwise our ours ourselves out over overall p per perhaps possible q que quite r rather really s same second secondly self selves seven seventh several shall she should since six sixth small smaller smallest so some somebody somehow someone something sometime sometimes somewhat somewhere still such t than that the their theirs them themselves then thence there thereafter thereby therefore therein thereupon these they thing things third this those though three through throughout thru thus ten tenth to together too toward towards turn turned turning turns twice two u under unless until unto up upon us v very via viz vs w was we were what whatever when whence whenever where whereafter whereas whereby wherein whereupon wherever whether which while whither who whoever whole whom whose why will with within without would x y yet you your yours yourself yourselves z zero