The Cambridge Multimedia Document Retrieval (MDR) Project : Summary of experiments

K. Spärck Jones*, P. Jourlin*, S.E.Johnson†, and P.C. Woodland†

Technical Report 517

Computer Laboratory, University of Cambridge
William Gates Building, JJ Thomson Avenue, Cambridge CB3 0FD, England

July 2001

Abstract

This report summarises the experimental work done under the Multimedia Document Retrieval (MDR) project at Cambridge from 1997-2000, with selected illustrations. The focus is primarily on retrieval studies, and on speech tests directly related to retrieval, not on speech recognition itself. The report draws on the many and varied tests done during the project, but also presents a new series of results designed to compare strategies across as many different data sets as possible by using consistent system parameter settings.

The project tests demonstrate that retrieval from files of audio news material transcribed using a state of the art speech recognition system can match the reference level defined by human transcriptions; and that expansion techniques, especially when applied to queries, can be very effective means for improving basic search performance.

*Cambridge University Computer Laboratory
†Cambridge University Engineering Department

1 Introduction

The MDR project has explored spoken document retrieval (SDR) under a range of test conditions. Overall the results

1. confirm the earlier small-scale VMR project findings (*VMR*) showing that well-established retrieval methods are robust under two critical features of the speech case, namely recognition error and a lack of natural, independently-marked document boundaries;

2. show that retrieval strategies exploiting familiar feedback methods in ways adapted to SDR conditions are effective means of compensating for recognition errors and can deliver good performance.

Thus given a stream of broadcast news, passages relevant to the user's information need can be successfully (and preferentially) retrieved.

On the speech side, the work has applied the powerful Cambridge/Entropic HTK speech recognition (SR) engine (*HTK*). On the document retrieval (DR) side the work has exploited the established Probabilistic Model as used in the City/Microsoft Okapi system (*TR446, IPM00*), developing this to enhance searching on the spoken data file with information gathered from a parallel text corpus. Others have found that this general strategy for applying search feedback is useful (*ATT*), and our research suggests that it may be a helpful way of overcoming the problems generated by speech recognition, for instance those associated with a limited vocabulary.

The project has conducted a very large range of SDR experiments, covered in detail in the papers in the publication list (for convenience we refer to these by abbreviations e.g. *TREC-8, SIGIR00*). This report provides a classification for these tests and summarises and illustrates our main results. We have not, however, simply repeated previous figures. *This account of the project work uses a new and systematic series of retrieval runs over our data sets, allowing comparisons in consistent environments.* During the project research we made many small changes to our retrieval system parameters, e.g. to the definition of the indexing baseline, so our detailed test results over time were not strictly comparable. The new tests, covering all the salient data variations and fixed, consistent versions of the main types of retrieval strategy we have considered, allow more robust comparisons. These new tests have not, however, included rerunning the basic speech processing. We modified this in detail over time, e.g. by using distinct vocabularies, by changing the amounts of training data also their epochs (to fit with different test data epochs), applying different processing strategies, changing acoustic and language model sizes, etc. But the generic approach to recognition has throughout been the same, using a 2-pass recogniser and a relatively large vocabulary: 65K words for TREC-6 and -7, 60K and 108K for TREC-8 (and -9), as described in *TREC-8* and *TREC-9*. (Removing commercials and other non-news material from the audio source - as required with the TREC-8 and -9 data - is done before transcription, using acoustic information.) While trying to redo the processing in exactly the same way for all the collections would have been excessive because of the time shifts in the training data, we do not believe these detailed changes in the basic speech processing had any significant impact on retrieval. We thus present the conclusions we draw from our new SDR tests as general, respectably-based findings for our set of test collections.

Our retrieval experiments have been within the overarching framework of the NIST/DARPA Text REtrieval Conference (TREC) evaluations (see e.g. *TREC*). These large-scale community enterprises have provided a general evaluation protocol and, for the SDR track within them, have supplied the essential test collections (documents, requests and relevance assessments). They have also, through successive evaluation specifications, steered our research. Thus as the detailed specifications have been developed by the participating SDR community, we have gained, for example, from the scope for controlled comparisons with work by other SDR teams. However though the scale of the TREC SDR test collections has increased with time, and has proved demanding for speech processing, the file sets are still very small by retrieval standards and all our results need validating on a really large scale, as well as under working conditions with real active users. We participated in TREC-7 (1998), TREC-8 (1999) and TREC-9 (2000). It should be noted that all of these experiments have been for so-called *adhoc* retrieval, designed to find documents relevant to a user's one-off information need as stated in the request input to system processing.

The TREC *data sets*, *retrieval conditions*, *comparison possibilities*, and *evaluation measures* that have provided the framework for our SDR experiments are summarised as follows (for further details see *TREC-SDR*).

1.1 Data sets

We have used test sets from TREC-6, -7, -8 and -9, generally representing increases in document file and request set sizes. For early experiments we supplied our own adhoc requests and assessments for the TREC-6 documents, replacing the original ones that were intended to recover previously-seen items and had been rapidly assembled to allow the TREC-6 evaluation. These new requests and assessments formed the CU60 test collection. The TREC-8 and -9 collections have the same documents but different request sets. The material is all broadcast news, from a variety of sources. The spoken documents are accompanied by text *reference* versions, independent human transcriptions for TREC-6 and -7, and in the form of the news close captions for TREC-8 (and -9). The TREC documents in general are rather short. The requests are simple sentences, with a second 'Terse' set for TREC-9 consisting of only a few key words, contrasting with the normal 'Short' form; the requests are written, not spoken. The relevance assessments are by the request originators. The details of these T6 - T9 data sets are given in Figure 1. As the figure shows, the size of the document file has increased substantially, though in the latest one the documents are shorter. Note that the 21K document set for TREC-8 and -9 defines the data set with known story boundaries - see below - used for our own new experiments. These documents are the news stories contained in the larger initial file which also includes non-news items. This larger data set, flagged * in Figure 1, has been used for some of our project experiments.

In addition we have made use of training data for the speech recogniser consisting, e.g. for TREC-8, of 146 hours of audio and of three text sets for language modelling, the largest with 190M words; and we have used various parallel text corpora for feedback experiments consisting of sets of news material, the main ones for our new tests having 60,000 stories (for further details see the project papers).

1.2 Retrieval conditions

In TREC-7 and -8, retrieval was for whole stories with independently-defined boundaries: the *boundary known (BK)* condition (sometimes labelled story known, SK). However while known boundaries are helpful for evaluation they cannot be assumed in practical SDR, so retrieval with *boundary unknown (BU)*, aka SU(N), was an optional condition for TREC-8 and

TREC data sets

	Requests	Documents	(Hours)	Av word/ request	Av words/ document	Av relevant/ request
TREC-6 CU60	60	1451	43	7.1	276	9.2
TREC-7	23	2866	87	14.7	269	17.0
TREC-8	49	21754	388	13.6	169	37.1
TREC-9	50	21754	388	11.7	169	42.5
terse	50			3.3		
TREC-8 *	50	28048	502			37.1
TREC-9 *	50	28048	502			44.3

Note, the average number of search terms per query is fewer than words
e.g. TREC-8 6.6 terms per query
TREC-9 5.8 and 3.0 terms per query

* full audio material including non-news items

Figure 1: Details of the TREC data sets

mandatory for TREC-9. In operation this results in passage retrieval (usually with in-context display), as in the MDR Demonstration System (*RIAO00b*). However since performance evaluation directly based on passages is complicated and expensive, the TREC BU evaluations have been somewhat artificially grounded in the full documents with their defined boundaries.

1.3 Comparison possibilities

The text transcriptions supply a reference standard for SDR purposes by providing the correct words while maintaining the characteristic discourse properties of speech. TREC SDR has normally required submission of a retrieval run on the text reference version of a collection (R1) as well as the obligatory speech run (S1), using the same retrieval apparatus for both. The retrieval performance comparison between these is quite straightforward for the BK case, using relevance assessments on the defined transcribed stories. The comparison is indirect for the BU case, via the BK assessments. The TREC evaluations have also promoted other comparisons. One is with NIST-supplied common baseline transcriptions, usually two (B1, B2), perhaps including ones from 'no-frills' recognisers, and not necessarily from the same recognisers each year. The other is by cross-recogniser (CR) comparisons between teams, each applying their own retrieval engines to others' transcriptions. As with the R1 versus S1 case, the same retrieval apparatus is used to allow factoring of the respective contributions that a system's recognition and retrieval components make to overall performance. Over successive TRECs there has been a shift from BK to BU as the required test condition (and correspondingly from BU to BK as optional), as noted in Figure 2, and a run on one of the baseline transcriptions has been obligatory as well as S1 and R1 runs.

In Figure 2, `x` shows the TREC retrieval conditions covered in the test series presented here, analogously labelled S1, R1 etc. We have not included any CR runs, since they involve complicated external references, and the baseline runs can be taken as good enough indica-

TREC data conditions

			r1	s1	b1	b2	
TREC-7	reqd	BK	x	x	x	x	
TREC-8	reqd	BK	x	x	x	x *	
	opt	BU	x	x	x	x	
TREC-9	opt	BK	x	x	x	x	short & terse requests
	reqd	BU	x	x		x	short & terse requests

* b2 was not required in TREC-8

Figure 2: TREC test conditions

tors of performance when our retrieval methods are combined with other recognisers. Both TREC experience in general (*TREC-SDR*), and our own CR tests in the past (*TREC-7 - 9*), suggest that while the broad level of recogniser performance (whether due to data or system characteristics) naturally affects retrieval performance, for SDR it is more important to have a sensible retrieval strategy than a finely honed recogniser.

The TREC-9 requirement for BU runs, together with the alternative Terse requests, naturally led to a change of emphasis in our research. The TREC-9 requirement for runs with the alternative Terse requests also encouraged us to compare system performance for the two TREC-9 request sets, and to examine the level of performance for the more realistic Terse requests. This has meant that we have not repeated all the device tests done on the T-6, -7 and -8 collections with their Short requests for the T9 requests. We have only done those that are most important for the full cross-collection range of strategy comparisons Thus some of the conclusions drawn about our tests refer primarily to results for T-6 - 8, others to the complete collection set.

1.4 Data irregularities

Quite apart from the fact that our data sets are all small as retrieval test collections, several have particular features that require comment.

The T6 CU60 collection relevance assessments were done on simple search output only, so the real relevance set is very probably larger than the known one. This implies that performance for other retrieval strategies is likely to be artificially low, since real relevant documents are formally labelled non-relevant. The T9 BU assessments include some items not within the BK data, and there were some other minor changes (e.g. in language model normalisation) to the official forms of the news file between TREC-8 and TREC-9, though the material is essentially the same. This means that even though the BU assessments (unrealistically) exploit known story boundaries, our T9 BU tests are not directly comparable with our full range of BK runs, and have their own BK analogues.

The fact that the TREC-8 and TREC-9 reference 'transcriptions' are actually close captions, while TREC-6 and TREC-7 has full human transcriptions, could in principle affect performance comparisons between R1 and S1. But we believe the actual effects to be negligible.

1.5 Performance measures

Speech recognition performance is usually measured by Word Error Rate (WER). However in the retrieval context success in recognising individual words, regardless of word order, is what matters. We have therefore used Term Error Rate (TER), which ignores order and treats substitution errors differently from WER, as better suited to the retrieval situation. We have also used Stopped and Stemmed Term Error Rate (SSTER) as a more specific response to the retrieval emphasis on content rather than function words and use of word normalisation. In general WER and TER follow one another, and the level of retrieval performance is correlated with TER.

For retrieval TREC uses four standard retrieval measures based on Precision and Recall (see Appendix in *TREC*). The single-number (Mean) Average Precision (MAP) measure is widely used, but is a very abstract characterisation of performance and we have therefore also made some use of other TREC measures, notably Precision at Document Rank Cutoff e.g. at rank 15 and also R-Precision. Comparative performance with these measures differs, but only in detail and the broad picture is similar, so we use MAP as the primary measure for our summary analysis here, with only occasional reference to the others.

This summary analysis is deliberately intended to emphasise the main results that hold regardless of variations in other factors, e.g. specific data set, and also concentrates on performance differences that are large enough to be of real practical importance. However as the detailed performance figures are very variable, our main conclusions are qualitative generalisations. *The conclusions are based on a convenient rule of thumb that looks for performance differences of at least 2 points on figures rounded to 2 places (e.g. at least 48% MAP as opposed to 46%),*. But even if performance differences expressed in this way may appear quite large, they are only informal. Significance testing is properly required, and we have therefore used the Sign Test, as the most conveniently applicable and suitable test for genuine retrieval performance differences, to check our main strategy claims.

2 Test classification

Broadly speaking, the project tests can be grouped as follows.

A. Speech (as motivated by the retrieval context)

1. Recogniser variants, e.g. 1 pass or 2 pass;
2. Transcription conditions, e.g. vocabulary size;
3. Task adaptations, e.g. boundary segmentation, commercial elimination.

B. Retrieval (as motivated by the speech context)

1. Retriever modification, e.g. weighting function, constant tuning;
2. Indexing choices, e.g. fixed compound terms;
3. Task strategies, e.g. parallel blind relevance feedback.

In both logic and process, A precedes B. However for presentation here is more convenient to take B first, since the investigations under A are best considered in terms of their effect on retrieval performance. (Note also that e.g. indexing devices are described in logical rather than processing order.)

3 Retrieval studies

Our studies fall into two major blocks, with relevance feedback as the point of division. Thus the first major group of tests covers basic indexing, the second the use of feedback, falling respectively under the headings of indexing choices and task strategies. Our earlier tests explored some retriever modifications, especially non-standard term weighting functions, and we have usually tuned function constants, for example, for different collections. For the tests reported here we have used the standard and well-established Probabilistic Model functions, as defined in *TR446* and *IPM00*. Tuning has some benefit, but we limited it here to choosing settings that would work reasonably across our collections, even if they were not optimal for each, in order to reduce cross-data variation.

Weighting functions

Summarising the main weighting devices for convenience here, we begin by defining the Combined Weight for an individual query term present in a document as a function of term collection frequency, within document frequency, and document length. The first of these is used to define a contributing Collection Frequency Weight for term `i`

```
CFW (i)  =  log N - log n (i)
```

where `N` is the size of (number of documents in) the collection and `n (i)` is the number of documents containing query term `i`. The Combined Weight is then

```
CW (i,j)  =  [ CFW (i) * TF (i,j) * (K1+1) ] /

             [ K1 * ( (1-b) + (b * (NDL (j)) ) ) + TF (i,j) ]
```

where `TF (i,j)` is the number of occurrences of term `i` in document `j`,

```
NDL (j)  =  (DL (j)) / (Average DL for all documents)
```

given `DL (j)` is the length of document `j`, and `K1` is a tuning constant to tone down the effect of term frequency and `b` is a constant to tone down document length (these are clearly collection dependent, but were fixed at `b=0.5` and `K1=1.0` for our tests).

If terms occur more than once in a query we use the query term frequency `QTF (i)` for the Query Adjusted Combined Weight

```
QACW (i,j)  =  QTF (i) * CW (i,j).
```

What constitutes a term is a separate matter. A document's matching score is the sum of weights for the terms shared with the query.

In relevance feedback a query is modified after searching to exploit information about the occurrence of terms in known or assumed relevant documents, at least to change the term weights and possibly also to add extra terms. With the relevance information we can define a term Relevance Weight

```
RW (i)  =  log [ ( (r (i) + 0.5)(N - n (i) - R + r (i) + 0.5) )
           / ( (n (i) - r (i) + 0.5)(R - r (i) + 0.5) ) ]
```

where `R` is the number of known relevant documents for the query and `r (i)` the number of relevant in which term `i` occurs. This is essentially a more refined substitute for the earlier use of simple collection frequency. Then to expand a given query we consider a list of the terms occurring in the relevant documents ranked by their Offer Weight

```
OW (i) = r (i) * RW (i)
```

and select (automatically in our case) the top `t` new terms, say 5 for short queries, to add to the query. All the terms in the new version of the query are now weighted by the Combined Iterative Weight with RW replacing CFW so

```
CIW (i,j)  =  [ RW (i) * TF (i,j) * (K1+1) ] /

              [ K1 * ( (1-b) + (b * (NDL (j)) ) ) + TF (i,j) ]
```

(QACIW is defined by analogy with QACW).

Our experiments with feedback have made use entirely of *blind* (or 'pseudo') relevance feedback, where the best ranked documents from a pre-search on the file are assumed relevant and used to modify the initial query for the real search. This requires a setting for `r`, the number of top ranked documents to assume relevant. The actual numbers we have used are given with the results, since they are collection dependent.

As detailed later it is also perfectly possible to use a document as if it was a query, to apply the relevance feedback mechanism for document expansion.

Organisation of the retrieval test summary

The MAP figures for the tests described in the next section are given in the two main Figures, 3 and 4, for T6 - T8 and for T9 respectively. *We first consider only the results for SDR for our own recogniser output, i.e. S1, for the BK condition, across the collections with their normal sentence-form (short) queries.* We comment later on the T9 Terse queries.

We then summarise the findings for reference performance, i.e. R1, and those for the baseline recognisers B1 and B2, both independently for the various retrieval options and by comparison with S1.

In the following section we examine the results for the BU condition, in similar style. Our BU tests have been relatively limited, partly because the BU condition only figured in later TRECs, and partly because the TREC assessment data and, more importantly, assessment methodology, were somewhat problematic; it was also the case that our treatment of non-news material in the audio file was different in our TREC-8 and -9 experiments, so the search files are not strictly identical. The BU condition is nevertheless far more important for practical applications and the project included substantial work on it.

All this will use informal MAP performance analysis as described earlier. It should be emphasised that the rule-of-thumb difference criterion is quite strong when required to hold across all test collections. While performance differences between devices for individual collections may be much larger, they are far from consistent. So our overall conclusions often have to be qualified as general rather than universal.

We subsequently consider significance test data for key comparisons, and then any particularly relevant other earlier project results outside the new studies framework.

Finally, we attempt to draw together our various IR findings.

3.1 Basic indexing

Most systems include routine preprocessing, e.g. to remove punctuation, standardise the treatment of abbreviations and cases, etc, and we have done this too. Mainstream indexing practice also usually relies on stopping and stemming. In stopping, function and other 'useless' words are eliminated so operational indexing and searching is confined to content words, i.e. the initial request is replaced by the search *query*. We have used a standard stoplist slightly modified to remove request words like "document" and also now number words. Stemming normalises term form, and we have applied the standard Porter stemmer. The indexing vocabulary after preprocessing, stopping and stemming, when used with CW weights, defines *simple baseline* performance.

Fairly early in the project it appeared it might be useful to add some manual tweaks to counter infelicities in the automatic baseline indexing. This *term mapping* dealt with spelling corrections, stemming exceptions of various sorts, and irregular verbs. More importantly we explored a range of indexing refinements beyond the automatic baseline, primarily motivated by classical arguments like those for multi-word compound terms. These vocabulary refinements included the use of some (manually defined) fixed compounds and of automatically-derived word pairs. They also involved the application of part-of-speech weights, *poswts*, for query term categories obtained by automatic parsing and applied as simple multipliers of the term weight.

These refinements were still seen as baseline indexing, in contrast to the use of feedback. Earlier experiments, notably with the CU60 and TREC-7 data, suggested that while these devices individually did little for performance, they could together lead to a modest improvement (*TREC-7*). However closer study showed that word pairs had detrimental effects, and later tests suggested that poswts had no general value. Our *elaborated baseline* has therefore been stabilised as a combination of the preprocessing, stopping and stemming of the simple baseline along with mapping and fixed compounds.

In general for the earlier collections, as Figure 3 shows, in the comparison:

elaborated baseline vs simple baseline,

we can conclude that:

- the elaborated baseline tends to do modestly better than the simple one, when used as the sole retrieval device, though the performance difference is not always maintained with other strategies like feedback.

We therefore took the elaborated baseline as the initial indexing for our TREC-9 work and for most of the new studies reported here. It would, however, be possible to have a perfectly adequate system with the simple baseline.

We also conducted experiments (cf. *ESCA99, SPCOMM00*) on the use of some manually-defined hierarchical term relations. These were limited to geographic location relationships and unambiguous WordNet hypernym relationships, defining semantic *posets*. Semantic poset indexing (SPI) appeared quite promising for the TREC-7 collection, and Figure 3 indicates that when added to the elaborated baseline they do somewhat better than the baseline for

BK - story boundary known condition

System	T6 CU60		T7				T8			
	r1	s1	r1	s1	b1	b2	r1	s1	b1	b2
Base Simple	65.85	64.26	46.99	45.34	40.90	32.50	42.99	40.27	37.72	37.96
BS + RBRF	65.66	64.56	50.71	50.68	42.07	34.49	50.36	48.13	42.71	44.64
BS + UBRF (X)	67.12	66.80	53.75	52.89	46.78	41.70	46.46	45.53	43.78	44.51
BS + DBRF (X)	67.62	67.87	51.47	50.76	50.19	36.12	41.62	43.76	41.18	41.02
BS + [D+U]BRF(X)	67.08	66.96	54.30	53.33	49.98	41.05	42.42	45.83	44.48	45.40
Base Elaborated	68.63	67.01	49.21	46.90	42.92	33.65	44.40	42.06	39.10	39.50
BE + RBRF	68.61	67.50	52.40	50.96	44.30	34.39	50.98	47.98	45.10	46.50
BE + PBRF (A)	67.10	65.41	52.88	51.18	47.93	39.69	43.86	43.96	41.73	41.86
BE + PBRF (B)	63.23	63.12	52.09	50.04	47.19	40.00	46.06	44.70	42.86	43.17
BE + PBRF (C)	69.80	67.81	52.30	49.60	46.05	40.52	43.96	43.14	41.41	41.59
BE + PBRF (D)	65.28	63.28	48.14	46.77	43.44	37.05	40.55	40.26	37.60	37.80
BE + PBRF (X)	68.34	66.50	51.19	51.30	47.04	41.37	46.47	45.98	43.32	43.69
BE + UBRF (A)	67.93	67.04	53.58	49.73	48.84	39.94	45.16	45.12	43.36	43.38
BE + UBRF (X)	67.00	67.07	53.56	52.31	46.57	40.59	47.90	47.15	45.19	46.17
BE + DBRF1 (X)	64.95	61.25	46.82	45.24	44.27	30.33	39.98	42.14	38.65	35.72
BE + DBRF2 (X)	66.57	65.89	45.21	44.20	40.90	32.97	38.65	41.45	39.05	36.22
BE + DBRF3 (X)	65.42	65.04	45.12	42.50	.....	.....	.....	.....	.....	.....
BE + DBRF (X)	69.41	69.12	52.55	51.73	51.14	36.86	41.99	44.13	42.14	42.05
BE + [P+R]BRF (A)	66.24	64.53	50.87	50.05	46.04	38.71	45.35	46.85	43.09	44.37
BE + [P+R]BRF (B)	62.80	62.58	51.72	50.51	45.13	39.02	47.49	49.21	47.34	47.62
BE + [P+R]BRF (C)	69.79	68.09	52.75	50.85	46.80	40.56	46.51	46.90	45.73	46.01
BE + [P+R]BRF (D)	64.19	63.07	46.28	46.99	43.86	38.75	42.48	42.72	40.29	41.21
BE + [P+R]BRF (X)	68.62	66.87	50.55	49.53	45.70	39.86	48.50	49.91	46.88	47.62
BE + [D+R]BRF (X)	70.57	69.23	53.08	53.92	49.51	37.56	48.45	48.91	46.33	47.74
BE + [D+P]BRF (X)	69.19	68.99	52.54	51.90	49.40	43.12	43.13	45.70	43.78	44.84
BE + [D+P+R]BRF (X)	69.17	67.19	51.97	51.12	48.55	42.10	45.40	50.05	47.20	47.88
BE + [D+U]BRF (X)	68.56	68.47	54.51	53.28	50.50	40.71	43.02	46.40	44.22	46.11
BE + SPI	67.97	66.45	51.50	48.55	44.69	34.89	43.96	41.80	38.84	39.22
BE + SPI + RBRF	69.19	67.50	54.36	53.61	47.37	37.05	50.03	47.07	44.85	45.46
BE + SPI + UBRF (X)	63.52	63.42	53.01	53.13	45.42	40.86	45.21	44.80	42.99	43.09
BE + SPI + DBRF (X)	69.51	69.00	52.52	49.64	46.50	38.18	42.60	43.40	41.29	40.86
BE + SPI + [D+U]BRF	65.97	65.73	51.48	52.27	46.07	40.45	42.85	43.98	41.30	42.14

T6: BRF t=1 r= 1 UBRF t=4 r=11 PBRF t=3 r=10
T7: BRF t=2 r=10 UBRF t=5 r=20 PBRF t=3 r=20
T8: BRF t=2 r=10 UBRF t=5 r=20 PBRF t=3 r=20
All lists (stop words, mappings, compounds) are those used for TREC-9
Base simple (BS): Basic preprocessing (uncaps, abbrevs, special characters, stemming, stopping)
Base elaborated (BE): BS + compounds and mappings
- BRF uses TR446 RW weighting scheme
- DBRF1 adds only new terms, pseudo-queries = 100 terms, t=100, r=10, tf=1
- DBRF2, DBRF3, DBRF add new and existing terms, pseudo-queries = 100 terms
with DBRF1 t=200, r=10, tf=1 ; DBRF2 t=400, r=10, tf=1 ; DBRF t=200, r=20, tf=0.5
Parallel corpus recorded pre test collection, pre-processing same for parallel and test corpus
(A): From 93 to 4/ 96 (pre-trec-6) = 125,489 stories (189,935 terms)
(B): From 94 to 4/ 96 (pre-trec-6) = 93,551 stories (156,018 terms)
(C): From 95 to 4/ 96 (pre-trec-6) = 55,848 stories (114,271 terms)
(D): From 1/ 96 to 4/ 96 (pre-trec-6) = 14,484 stories (56,122 terms)
(X): Multi-parallel : Trec-6 -> 4/ 94 to 4/ 96 (pre-trec-6) = 60,000 stories (119687 terms)
Trec-7 -> 10/ 95 to 5/ 97 (pre-trec-7) = 60,000 stories (121890 terms)
Trec-8 -> 11/ 96 to 1/ 98 (pre-trec-8) = 60,000 stories (129614 terms)
T6-CU60 assessments are sparse

Figure 3: Average Precision results for T6, T7 and T8 test collections

BK - story boundary known condition

		T9 Short queries				T9 Terse queries			
		r1-t8	s1-t8	b1-t8	b2-t8	r1-t8	s1-t8	b1-t8	b2-t8
Base Elaborated		38.06	35.46	34.06	34.09	43.41	40.49	38.37	38.72
BE + RBRF		43.09	37.89	38.55	37.59	47.59	44.91	45.07	42.43
BE + PBRF	(X)	37.96	36.19	34.54	34.36	44.14	42.62	40.55	40.39
BE + UBRF	(X)	40.64	38.30	36.53	37.17	48.10	45.07	43.82	43.40
BE + DBRF	(X)	37.60	36.64	35.55	35.95	42.54	43.20	41.28	41.32
BE + [P+R]BRF	(X)	38.71	35.36	37.15	37.17	42.45	41.02	40.59	40.61
BE + [D+R]BRF	(X)	42.46	37.77	38.86	38.59	45.17	45.67	46.10	42.79
BE + [D+P]BRF	(X)	36.11	36.60	36.27	35.57	42.37	43.73	42.43	42.16
BE + [D+P+R]BRF	(X)	36.73	35.23	37.42	36.90	39.71	42.08	41.71	41.69
BE + [D+U]BRF	(X)	38.27	37.28	36.89	37.14	44.31	46.33	44.56	43.82

T9: BRF t=2 r=10   UBRF t=5 r=20   PBRF t=3 r=20   (same as T8)
    DBRF adds new and existing terms
        pseudo-queries = 100, t=200 r=20, tf=0.5   (same as T8)

X: For Trec-8 -> Nov 96 to Jan 98 (pre-trec-8) = 60,000 stories (129614 terms)

Figure 4: Average Precision results for T9 test collection, two query sets

this data set. However the table shows that in general posets do not improve performance, and the same applies when they are combined with feedback: indeed when parallel collections are used posets are harmful. With the TREC-8 collection they did harm rather than good (*TREC-8*), and were therefore retired.

In general, retrieval research has shown that the kind of manual enhancement to simple indexing we studied can make some, but a minor, contribution to performance, and can require some effort to be useful for large collections. With small collections like ours there is a danger of overfitting to the data. Relevance feedback can have a much more significant effect on performance, and feedback thus became the focus of our retrieval experiments.

3.2 Retrieval with feedback

Feedback with genuine relevance information is well established as helpful. Some, though more modest, performance gain has been obtained with blind feedback (see e.g. *FREFL*). For our SDR case, without genuine user participation, relevance could only be assumed. However, apart from the general reason for applying feedback, it appeared that might be further advantageous in the speech case since it could help to counteract recognition errors by importing missing or associated terms. Following this line of argument suggested that it would also be helpful to base query feedback on a parallel text collection (*ESCA99*). This would not only provide error-free information about term behaviour: in the case where the retrieval file is relatively small and the parallel collection is large, it might provide more reliable comparative frequency information.

Thus conventional or Routine Blind Relevance Feedback (RBRF) is compared with Parallel Blind Relevance Feedback (PBRF) for query modification through term reweighting and addition. A further extension along these lines, originally proposed by Singhal and Pereira (in *ATT*) and of potential value particularly where the retrieval file documents are short, is to expand these by feedback using the parallel collection, i.e. by treating each search file document in turn as a query against the parallel set, and expanding each search file document using terms from the parallel set only. This gives us Document Blind Relevance Feedback, DBRF.

For comparative study these variations of expansion can be treated as completely separate devices which can be tested in any combination, as in *TREC-8* in particular, though when RBRF and PBRF are combined it seems more logical to treat them as one (see below). Our initial experiments with feedback including PBRF on T7 as described in *ESCA99* and *SPCOMM00*, and also with DBRF in *TREC-8*, showed that feedback can be as helpful a device for speech as it has generally been found for text, and that exploiting a parallel collection could be useful. We therefore carried out a systematic series of tests applying individual and combined feedback possibilities, designed to establish the relative values of the different forms of feedback and information source.

When both forms of query expansion are used, these can be combined in slightly different ways. One is to modify the query in two stages, first by PBRF and then by RBRF. The other is to unite the two document files and do a single expansion cycle, for UBRF. We have tested both. However since the UBRF option is more convenient practically, and also seems cleaner intellectually when the document files are comparable (e.g. in document length), we used this alone for TREC-9.

There are also system parameters to set the number of documents deemed relevant, `r`, from which expansion terms are selected, and the number of expansion terms, `t`, to be added to the existing query. It is normal to relate the former to collection size and the latter to query size. Experiments with a range of options reported in *RIAO00a* show that small differences do not normally affect performance, but it is sensible, given noisy speech data and uncertain actual relevance to be conservative in setting the parameter values. Those used in the tests presented here are given in Figures 3 and 4. Note that the number of extra terms added per query is relatively small. Using parallel collections also implies decisions about the base for determining collection frequencies for term weighting. In query expansion with BRF, the collection frequency components of term weights - both for the initial presearch and in the calculation of the relevance weights for the expanded query used for the actual search - are drawn from the document set that is supplying the expansion terms. They are thus drawn from the parallel collection only for PBRF, and from the parallel collection and the search (test) collection combined for UBRF. With document expansion (DBRF) alone, collection frequency components are drawn from the search collection documents that were expanded using the parallel collection, but when DBRF is combined with query expansion, the term weights are those defined by the query expansion base. (The other contributors to term weights, namely within-document term frequency and document length, are of course defined by the search collection.)

Feedback based on parallel collections also requires decisions about the size and age of the parallel corpus. Experiments on the T7 data reported in *SPCOMM00* showed that using a parallel collection was useful even if it was not very large or contemporary with the test file. However it also appeared that larger size was more valuable than recency. We have investigated the effects both of relative size and, for fixed size, of relative recency in a systematic way in the current tests. The feedback tests reported next all used parallel corpora of the same size, but with recency relative to the document set (the results labelled X in Figures 3 and 4). Tests with different parallel corpus sizes are considered separately later. We also compare two same-size corpora, differing (though not much) in relative recency, for the T-9 data, in Figure 5. Details of the various corpora are given in the tables with the results.

The feedback options imply a large number of comparisons. We group these (a) under RBRF alone; (b) PBRF alone or in combination with RBRF; (c) DBRF alone and in combination with query feedback. The full series of combinations uses the elaborated baseline. We conclude with a note on feedback with the simple baseline. The comparisons are for T7, T8 and T9, with the standard form queries: we exclude T6 because of the relevance data limitations mentioned earlier (though the results table includes the performance figures). *The conclusions below are for the S1 case only.*

a) conventional query expansion - RBRF :

RBRF vs baseline alone

In general RBRF enhances performance substantially compared with the baseline indexing on its own.

b) parallel corpus query expansion - PBRF :

PBRF vs baseline

PBRF is usually better than baseline.

PBRF vs RBRF

PBRF is no better than, and sometimes inferior to, RBRF.

PBRF+RBRF vs RBRF; UBRF vs RBRF

The PBRF+RBRF combination of two query expansion information sources is no gain over

RBRF alone. The UBRF version of the combination is the same.

c) parallel corpus document expansion - DBRF :

DBRF vs baseline alone; DBRF vs RBRF

DBRF on its own is superior to baseline, but overall inferior to RBRF.

DBRF+RBRF vs baseline; DBRF+RBRF vs RBRF

As in the previous case, the feedback combination is better than baseline indexing on its own, but not superior to RBRF alone.

DBRF+PBRF+RBRF vs baseline; DBRF+PBRF+RBRF vs RBRF

DBRF+UBRF vs baseline and vs RBRF

These are the most interesting combinations, representing the most comprehensive forms of feedback strategy. But while the results sometimes show much better performance for full DBRF+PBRF+RBRF combination than for baseline indexing, the combination does not do better than RBRF alone, and the form using UBRF is not superior to the three-component one.

The comparative results for the individual collections are somewhat different, but some findings emerge. Thus overall for this large series of informal comparisons, the conclusion is that

- relevance feedback is indeed helpful, even when it is only blind feedback. But this is primarily through the direct use of the search collection only for conventional query expansion, with no material further gain from exploiting a parallel corpus.

The smaller range of comparisons in Figure 3 for feedback when used with the *simple* baseline shows a generally similar picture, with RBRF boosting performance very markedly, and with no consistent additional benefit from alternative or additional devices using a parallel corpus.

Different parallel corpora

Our comparative tests here were to examine first, the effects of increasing corpus size, and in particular whether a really large corpus is much more helpful than a modest one; and second, the effects of corpus recency, given the same corpus size. As noted, earlier tests (*SPCOMMOO*) suggested recency is more important than size.

Figure 3 shows the test corpora used: four, A - D, ranging from nearly 200K down to about 15K stories drawn from the same time period, and three, labelled X, of the same 60K size drawn from the time period immediately preceding each TREC document set. The experiments focused on corpus effects for PBRF and PBRF+RBRF: tests with DBRF would have been a very large processing effort for little likely information gain.

a) corpus size :

PBRF (A) vs PBRF (D); PBRF+RBRF (A) vs PBRF+RBRF (D)

These comparisons show that performance with the largest corpus is substantially better than with the smallest.

b) corpus recency :

PBRF (A) vs PBRF (X); PBRF+RBRF (A) vs PBRF+RBRF (X)

These comparisons, on the other hand, show that performance with the recent corpus is the same as, or even better than, that for the larger one.

```
BK - story boundary known condition

T9 Terse queries
```

		r1-t8	s1-t8	b1-t8	b2-t8
Base Elaborated		43.41	40.49	38.37	38.72
BE + RBRF		47.59	44.91	45.07	42.43
BE + PBRF	(X/Y)	44.14/46.67	42.62/44.99	40.55/43.64	40.39/43.51
BE + UBRF	(X/Y)	48.10/49.07	45.07/46.82	43.82/45.33	43.40/45.13
BE + DBRF	(X/Y)	42.54/43.44	43.20/46.20	41.28/44.71	41.32/45.22
BE + [P+R]BRF	(X/Y)	42.45/46.28	41.02/45.23	40.59/45.58	40.61/45.47
BE + [D+R]BRF	(X/Y)	45.17/46.22	45.67/49.02	46.10/49.41	42.79/46.62
BE + [D+P]BRF	(X/Y)	42.37/45.33	43.73/49.59	42.43/48.52	42.16/48.49
BE + [D+P+R]BRF	(X/Y)	39.71/45.29	42.08/49.77	41.71/50.26	41.69/50.12
BE + [D+U]BRF	(X/Y)	44.31/46.49	46.33/50.76	44.56/49.61	43.82/49.66

```
X: For Trec-8 ->  Nov 96 to Jan 98 (pre-trec-8) = 60,000 stories (129614 terms)
Y: SIGIR'00 parallel corpus processed under main runs configuration.
   62,926 documents, 177740 terms, from 1st Jan. to 30th June 98
```

Figure 5: Average precision results, T9 collection, alternative parallel corpora

- The conclusion is that having a recent corpus is more important than having a larger one.

This conclusion is supported by the results for the intermediate sized corpora B and C, which represent more recent selections from the whole A data set, and sometimes perform better than A. We have not been able to test for the relative value of a very large very recent corpus, and recency may also be of particular importance for retrieval from news data.

Figure 5 suggests that this is indeed the case (cf. *TREC-9*). This shows results for the T-9 Terse queries using the same range of devices with two parallel corpora of comparable size, one dated just before the document file, one contemporaneous with it. Performance for the latter is much better than for the former, and is also better for DBRF and UBRF strategies than for simple RBRF alone. We should note that both here and with the corpus size comparisons, the same picture emerges with other, R1 and B1/B2 versions of the document collection for the BK condition. It may be the case that larger file size is important, and that values for `r`, in particular, and `t` should be adjusted to suit size rather than held constant as here. We also did not investigate the quality of the material in the parallel files to see whether this had any effect on performance.

3.3 Reference performance

There are two questions of interest here:
a) whether R1 performance is much above S1 levels; and
b) whether the various devices behave in a similar way in the two cases.

We assume, as mentioned earlier, that the fact that the TREC-8 (and TREC-9) reference data are close caption has no significant effect for these comparisons. Then, as Figures 3 and 4 show, while for the baseline cases S1 performance is somewhat below that for R1, using additional devices like feedback can bring S1 performance up to the R1 baseline level, and can indeed eliminate the difference between S1 and R1 when both use feedback. This is encouraging for SDR, but needs confirmation from tests with much larger document collections.

Making the same series of device comparisons for R1 as was made above for S1 shows that the relative behaviour of the devices is generally similar, with the exception of strategies using a parallel corpus. On the whole, it seems that these are less useful (insofar as they are useful) for R1 than for S1, presumably for the good reason that the document file itself is more reliable in the R1 than in the S1 case.

- Overall, our SDR performance matches that for the text reference level.

3.4 Alternative recognisers

Here the questions are:

a) whether our retrieval strategies when applied to other recogniser outputs behave in the same or similar way as they do with our own; and

b) where recogniser output quality is relatively low, as with T7 B2, whether the retrieval strategies can raise performance in a useful way, and in particular contribute more in this situation than is needed when recogniser performance is good.

Though T9 uses the same documents and B recogniser output as T8, because there were new requests we can treat these as distinct collections. We give T9 results for both B1 and B2, though the official TREC-9 test was on B2 (confusingly relabelled B1) alone. The results for all our collections are shown in Figures 3 and 4.

- In general, the relative behaviour of the retrieval devices we have studied when used with other recognisers follows the same pattern as for our own,

with the interesting exception that for the T7 collection all the parallel corpus-based strategies using B1 and B2 are more or much more effective than they are for S1. This must be attributable to the fact that for the T7 data B1 and B2 performance is much lower than that for S1 and R1; with T8 and T9 B1 and B2 are much nearer to S1 and R1. The parallel corpora thus appear to provide a lever for raising poor recogniser performance, but this needs much more testing.

3.5 Query types

As noted, the query types for all of T6 (with CU60) - T9 are all relatively straightforward sentences or sentence-like, giving an average of 5.8 baseline search terms. The second Terse query set for T9, essentially an alternative user version of the first that mimics web-engine requests consisting just of a few terms or a phrase, has 3.0 terms per query. The TREC-9 tests required runs for both sets (*TREC-9*).

Performance for these Terse queries is shown in Figure 4, compared with that for the Short queries. The figures are of some interest, because performance for the Terse queries is substantially better, across the board of conditions and devices, than for the Short ones. This suggests that the term choice for the Terse queries was better (they were also in practice also not always very brief). However for the Terse queries themselves, relative performance for the various devices, and specifically for S1, is much the same as for the Short queries.

- For Terse queries, as for Short, RBRF performs well compared with other feedback variations.

This conclusion must also be treated with caution because it is based on only one set of information needs and search file.

3.6 Retrieval without document boundaries

For SDR, the BK condition we have considered so far, where retrieval is from a clean file containing distinct, separated documents, represents an artificially created test condition and one unlikely to be encountered in many SDR applications. Even where item boundaries can be reliably identified by routine file preprocessing, the items may not be coherent in content, and it is thus natural to design SDR systems e.g. for news material, to select passages or segments focused on query topics: these are normally defined by fixed length *windows*. This is done in the MDR demonstration system (*RIAO00b, IJST01a*).

In addition it is useful to be able to remove or pass over music, commercials, etc. (These do not all figure in the TREC BK files.) Our BU processing is described in detail in *TREC-8*, *RIAO00a*, *TREC-9* and *IJST01b* which report many experiments in the choice of window sizes, strategies for identifying commercials etc, and is also further discussed in the Speech section below.

The BU processing we have explored is of course generally relevant to retrieval performance. But the form of the TREC BU evaluations presents problems. The retriever delivers matching transcription windows, but in the evaluations these windows were mapped onto BK stories, and retrieval performance was measured by success in retrieving full stories. This presumably has some bearing on performance for the window retrieval that real BU operation would involve, but the precise relation between the two is not well-defined. More importantly, the evaluation specification defined all windows retrieved after the first within a story's boundaries as irrelevant, though in true passage retrieval independent windows might well be viewed and assessed separately. This led us to select just one representative from a group of close windows for output. It is thus possible only to make rather loose inferences, from the artificial TREC form of BU evaluation, as to what true BU retrieval performance would be were it practicable to assess all retrieved windows for relevance.

It is also the case that though the document sets for T8 and T9 are in principle the same, the T8 assessments were on a subset of the data rather than the full data, so the T8 and T9 BU runs are not strictly comparable (see the notes on Figure 6).

We carried out a large number of BU experiments for TREC-9 in particular, geared to the evaluation context (see *TREC-9*). BU tests are very effortful, and both for this reason and because of the evaluation conditions just mentioned as well as that of the assessment data noted earlier, we have not attempted to complete BU tests for the range of devices considered for BK. Our TREC-9 experiments in particular used a parallel collection for query expansion alone, applying the UBRF strategy, and also explored document expansion, both alone and in combination with query expansion. The parallel data was contemporaneous rather than prior news material. The work also used its own tuning constants and expansion parameter settings, especially for document expansion to take account of the BU 'document' length. The choice of settings for the weighting constants `b` and `K1` is unlikely to have had a large effect, but more terms were included in expansion than in the main experiments, which would be likely to affect performance more strongly.

For all of these reasons, the results for BU are not comparable with the BK ones presented earlier, and have to be considered in their own right, along with their separate corresponding BK runs. But this is reasonable given that the experiments covered both the T8 and T9 collections and the Short and Terse query sets for the latter, i.e. three different query sets, albeit for essentially the same search file. The results are given in Figure 6. Exploratory runs for T8 BU suggested that while query expansion is helpful, document expansion is not consistently so, presumably because the initial documents are too ill-defined, so document expansion was not repeated for T9 BU. The T8 and T9 results for S1 show that UBRF is very useful indeed compared with the baseline. (The table also gives the runs for T8 with document expansion, which show why it was not tried for T9.)

For BK under the same conditions, there is the same very substantial gain for both T8 and T9, and for both query sets for T9 (with the Terse queries, as before, somewhat better than Short). These BK runs also show, for both T9 as well as T8, that document expansion alone is inferior to query expansion, but that when they are combined in DBRF+UBRF the result is better than UBRF alone, giving a striking performance gain over the baseline. It seems clear that it is the lack of proper document boundaries that makes document expansion ineffective for BU.

These runs also show that for BU, S1 performance with UBRF is essentially the same as that for R1, and also slightly superior to that for the alternative recogniser transcription B2; while for BK, S1 is very near R1, and typically slightly superior to B2, in baseline and combined expansion runs. We should also note that, while the form of the evaluation using the BK story boundaries implies that the conclusion can only be tentative, it appears that though with BE alone performance for BU is much below that for BK, using expansion with BU closes the gap when compared with expansion with BK.

TREC-9 illustrates performance using the alternative R-Precision measure: the broad pattern is the same as with MAP, though there are some specific differences.

Given that we only used one document file for restricted BU experiments, some caution about findings is required, even though we did vary the collection through using different request sets, transcriptions, and versions of commercial elimination. But some comments both about BU retrieval and about device performance can be made. Thus

- it appears that where a search file lacks clear document boundaries, a window-based strategy can nevertheless deliver relatively good performance; and
- query expansion exploiting a parallel corpus appears to be a very effective retrieval device for window-based searching.

3.7 Attainable performance

The tests under the BU condition, and the corresponding BK ones, also bring out an important, different point. The table below shows the (simple rounded) performance figures for S1 T8 and T9 BK taken from the main runs tables, Figure 3 and Figure 4, and from the BU-based table, Figure 6, side by side, labelled 'main' and 'new' respectively. The columns in each pair are not directly comparable, because of detailed system and data differences, but are near enough for present purposes (the baseline figures are similar). The relative performance

BU - STORY BOUNDARY UNKNOWN CONDITION

	T8 Short queries			T9 Short queries			T9 Terse queries		
	r1-t8	s1-t8	b2-t8	r1-t9	s1-t9	b2-t9	r1-t9	s1-t9	b2-t9
Base elaborated		30.89			26.17			29.76	
BE + UBRF (X)	51.04	51.15	48.08	40.03	38.83	37.08	44.02	42.99	40.75
BE + DBRF (X)		38.68							
BE + [D+U]BRF (X)		48.94							

BK - STORY BOUNDARY KNOWN CONDITION

	r1-t8	s1-t8	b2-t8	r1-t9	s1-t9	b2-t9	r1-t9	s1-t9	b2-t9
Base elaborated	48.19	46.29	43.31	37.40	34.53	33.67	40.90	37.94	36.39
BE + UBRF (X)		57.41			42.98			47.04	
BE + DBRF (X)		50.76			38.23			42.11	
BE + [D+U]BRF (X)	59.04	60.06	58.15	47.44	46.22	46.55	50.99	49.18	48.56

```
BU UBRF  T8, T9   t= 20, r=26
   DBRF  T8       t=100, r=15

BK UBRF  T8, T9   t=  8, r=22
   DBRF  T8, T9   t=200, r=10

T8 relevance assessments only for 21,754 story subset of the data
T9 relevance assessments for all the data (28,048 items)

T8 BK indexing was only over the 21 K subset
T9 BK indexing was over the 28 K set

T9 BK transcriptions differ from T8 BK in including non-news items
    though some commercials were removed, having different language model
    normalisation etc; but each of the pairs r1-t8 and r1-t9; s1-t8 and s1-t9;
    b2-t8 and b2-t9 gives very similar transcriptions

(X) Parallel corpus Jan 98 to June 98 = about 54,000 stories

BU windows 30 secs, overlap 15 secs; document expansion adds 1 to tf

b2 is same as b1 in official TREC-9 specification
```

Figure 6: Average Precision results for T8 and T9, boundary unknown condition and comparable boundary known

differences for the various devices follow the same overall pattern, with all forms of expansion better than the baseline and UBRF better than DBRF, but the actual performance differences in the 'new' column are *much* greater. Moreover with 'new', DBRF+UBRF is better than DBRF alone, which is not true of 'main'.

BK condition, S1

	T8		T9 Short		T9 Terse	
	main	new	main	new	main	new
baseline	42	46	35	35	40	38
+ U	47	57	38	43	45	47
+ D	44	51	37	38	43	42
+ D+U	46	60	37	46	46	49

These larger differences may be partly attributable to system changes e.g. in language model normalisation, or to choices of expansion parameter settings, but could be partly or even mainly attributable to the difference in parallel corpus and the benefits of contemporaneity. This observation does not make the broad view of *relative* device performance we have developed using the main run results untenable. But it emphasises the need to take context into account when considering *absolute* performance levels. Thus we may say

- query expansion using a parallel corpus can deliver large performance improvements when tailored to collection properties; and
- good retrieval performance can be obtained with simple, fully automatic system devices.

3.8 Significance tests

.

For this report we need only ask whether apparently large performance differences are actually statistically significant. This is to check the informal conclusions drawn earlier. We are not concerned with whether even small differences are in fact significant. Further, we are really only interested in the case where there is at least a 2-point performance difference (as defined earlier) across all the collections.

Unfortunately, as mentioned earlier, this is hard to find, as Figure 7 shows. This illustrates performance differences, using the rule-of-thumb criterion, for selected S1 runs with the BK condition. For T8 and T9 we show comparisons using both figures from the main runs of Figures 3 and 4 and from those used in the BU/BK comparisons ('new') of Figure 6. While there are some cases for the main runs where the required minimum difference does hold across the board, e.g. for BE vs BE+RBRF, there are many others where performance is sometimes different and sometimes the same, and even some where the direction of difference varies across the collections. (The same applies to the R1 and B1/B2 cases, making it very hard to do more than identify a few general tendencies.)

However when we take the differences for BK 'new' and also BU into account, there is a consistent tendency favouring BE+UBRF, with generally material gains over BE alone and much larger ones for the 'new' comparisons based on runs with an appropriately tailored system. On the informal criteria these selected comparisons, given in Figure 8, suggest very large gains for more liberal expansion, given suitable system tuning.

	BK, S1 : T7	T8	T9S	T9T	T8 new	T9S new	T9T new	BU, S1 : T8 new	T9S new	T9T new
BE vs BE+R	<<	<<<	<	<<						
vs BE+P	<<	<<	=	<						
BE+R vs BE+P	=	>	>	>						
BE vs BE+P+R	<	<<<<	=	=						
vs BE+U	<<	<<	<	<<	<<<<+	<<<<	<<<<+	<<<<+	<<<<+	<<<<+
BE+R vs BE+P+R	=	<	>	>>						
vs BE+U	=	=	=	=						
BE vs BE+D	<<	<	<	<	<<	<	<<			
BE+R vs BE+D	=	>>	=	>						
BE vs BE+D+R	<<<	<<<	<	<<<						
BE+R vs BE+D+R	<	=	=	=						
BE vs BE+D+P+R	<<	<<<<	=	<						
BE+R vs BE+D+P+R	=	<	>	>						
BE vs BE+D+U	<<<	<<	<	<<<	<<<<+	<<<<+	<<<<+			
BE+R vs BE+D+U	<	>	=	=						
BE+U vs BE+D+U	<	=	=	=	<	<	<			

R=RBRF, P=PBRF, D=DBRF, U=UBRF   T9S=T9 Short T9T=T9Terse
new from separate BU tables
< = 2 full points, integer rounded MAP values, << = 4 points etc

Figure 7: Magnitude and direction of performance differences, S1 for BK and BU conditions

Unfortunately, this rosy picture is not wholly borne out when significance tests are applied. We limited our tests to the selected comparisons, since these involved the best looking strategies. The significance test results are shown in Figure 8, where the Sign Test values are shown together with the informal comparison values previously shown alone in Figure 7. The significance test figures are shown as percentages, so values larger than 5.0 are NOT significant.

The pairings between the informal comparison values and the actual significance test values are rather instructive. Thus as Figure 8 clearly shows, the informal comparison values, even when quite large, are not always paralleled by a significant Sign Test difference. For example, while an informal difference of $=$ or $<$ is not normally significant, some of $<<<$ are not either. Considering first the selected comparisons for MAP results given in Figures 3 and 4, there is no performance gain over BE, i.e. over the elaborated baseline strategy, for T7, while for T8 only the full expansion of BE+DBRF+UBRF gives a significant gain. There are no significant differences for the T9 Short requests; however with the Terse requests all three selected forms of expansion are a significant improvement over the baseline alone. But perhaps these findings are not altogether surprising, since we deliberately kept collection tailoring to a minimum in order to control cross-collection comparisons.

Indeed the comparisons based on the figures for BU and BK taken from Figure 6 ('new') are more interesting, and serve to emphasise the point that while holding parameter settings constant across collections may be deemed a desirable form of experimental control from one point of view, taking genuine collection differences into account and setting system parameters accordingly is another, equally legitimate form of comparison. Thus the Sign Test values for the BU condition show a clear advantage for query expansion with BE+UBRF compared with BE alone for both T8 and T9, here in line with the very large informal differences. The same holds for BK with either BE+UBRF or BE+DBRF+UBRF for T8 and for T9 with Terse requests, though not for the T9 Short requests; this is again in parallel with the very large informal differences, though as the T9 Short request significance values show, large informal differences do not guarantee corresponding significant performance gains.

An informal analysis of the Sign Test figures suggests that what is happening is that where informally strategy X is much better than strategy Y, in the pairwise Sign Test query comparisons when X is better than Y it is much better, but when it is worse it is only very slighly worse, perhaps justifying the conclusion that on the whole, strategy X is to be recommended. But more strictly, *after* applying significance tests, we can conclude that:

- query expansion, and in particular expansion exploiting a parallel corpus, can be helpful for BU as well as BK retrieval, when strategy parameters are appropriately tailored.

3.9 Other experiments

Cross recogniser tests

As mentioned in connection with the TREC evaluation design, participants were encouraged to exchange transcriptions and apply their own retrieval engines to these: this is an extension of the use of the B1/B2 transcriptions. We did this, and our tests with other transcriptions are reported in *TREC-7 -8* and *-9*, also *ICASSP99*. Detailed comparisons are not appropriate here, since different recognisers may have distinct design goals. The important points are that the retrieval devices we found effective with our own transcriptions continued to work with others, but also that, not surprisingly, overall performance varies with recogniser WER (see

BU, S1 :

	T8	T9 S	T9 T
BE vs BE+U	<<<<+	<<<<+	<<<<+
BE vs BE+U	0.00008	0.009	0.0006

BK, S1 :

	T7	T8	T9 S	T9 T	T8 new	T9 S new	T9 T new
BE vs BE+U	<<	<<	<	<<	<<<<+	<<<<	<<<<+
BE vs BE+U	7.84	8.5	56.8	3.28	0.006	15.2	0.03
BE vs BE+D	<<	<	<	<	<<	<	<<
BE vs BE+D	28.6	88.5	47.1	4.44	47.1	20.3	0.66
BE vs BE+D+U	<<<	<<	<	<<<	<<<<+	<<<<+	<<<<+
BE vs BE+D+U	9.3	2.1	39.2	4.44	0.00152	11.9	0.26
BE+U vs BE+D+U	<	=	=	=	<	<	<
BE+U vs BE+D+U	13.4	66.5	8.54	67.2	24.3	100.0	48.0

BE=baseline, D=DBRF, U=EBRF   T9 S = T9 Short, T9 T = T9 Terse

comparisons based on Table~\ref{resultsBU} for BU condition and BK labelled 'new', otherwise on Tables~\ref{results6-8} and \ref{results9}

< = 2 full points, integer rounded MAP values, << = 4 points etc

Sign Test values as percentages; values greater than 5% are NOT significant

Figure 8: Significance tests for key comparisons, BU and BK conditions, using the Sign Test

also *TREC-SDR*). However this variation is slight and SDR performance over the successive TRECs tended to converge, presumably because the same generic DR technology was being used, and this has more effect than some difference in SR performance.

Out of vocabulary (OOV) terms and feedback

It is evident that the expansion devices can also provide means for overcoming matching problems due to out-of-vocabulary (mis)recognition. In recognition, all the input is mapped to whatever vocabulary is in use, so input words not in the vocabulary are replaced by others, which themselves then have false occurrences. Query expansion in particular can provide alternative good terms to increase the chance of relevant document matches. (We consider the OOV problem further under Speech studies, see Section 4)

The project work thus included tests with 5 recogniser vocabularies from small (3K words) to quite large (55K), on the T8 BK data (*SIGIR00*). The effect on retrieval performance in general is considered under speech factors later. Here we concentrate on the compensation effect of feedback.

Working with the elaborated baseline would require handtailoring of the baseline elaboration for each test vocabulary, and also complicate the vocabulary size picture, so all the *SIGIR00* experiments with the TREC-8 data were done with the *simple* baseline, i.e. with indexing without word pairs. We studied expansion using RBRF and UBRF for queries, and DBRF for documents both alone and with UBRF.

The results, illustrated in Figure 9, showed that all forms of expansion were valuable, with UBRF much superior to RBRF and RBRF much superior to the simple base. It is noteworthy that UBRF helped most with the smaller vocabularies, RBRF with the larger. Using document expansion was also helpful, with the combination DBRF+UBRF superior to UBRF alone and the latter much better than DBRF alone, even though DBRF was still better than the simple baseline.

It thus appears that a parallel collection can be more helpful in small vocabulary situations than when a large one is available, though this needs further exploration. There may, in particular, be a complicated mixture of effects since with smaller vocabularies it is the case both that more words are missing and that more words are maltreated. Thus as *SIGIR00* suggests, expansion may be especially helpful where there is a high Word Error Rate in transcription.

3.10 Retrieval assessment

Drawing together the project findings for retrieval, it is evident that, in the SDR as in the text case, feedback technologies are effective. This is particularly so for query expansion. It has not been fully demonstrated for document expansion, where the nature of the parallel corpus may have significant impact and the lack of independent document boundaries may make expansion difficult to control. It also appears that a relatively straightforward application of well-founded methods works as well as more complex strategies. Thus as far as retrieval devices are concerned

- query expansion is advantageous;
- document expansion *may* be helpful.

More generally, our tests confirm that for speech data as for text,

- the probabilistic model is a good general approach to retrieval.

At the same time our range of tests as a whole emphasises the point that while the model is fairly robust, performance does benefit from setting parameters to suit collection characteristics; and these characteristics may also mean that individual devices are not always effective (or indeed not appropriate, cf. document expansion for BU). This general observation is not novel, but is worth repeating for the speech case:

- parameters and devices should be chosen for collections.

Even so, relative performance varies considerably for different collections. Absolute performance as in the general text case, also declines with increase in collection size.

In relation to the interaction between the SR and DR components of the system as a whole, it is evident that given a reasonable level of SR performance and well-founded retrieval methods, it is possible to reach SDR performance levels competitive with text retrieval. The presumption was that SDR would be relatively impervious to detailed SR failures, and that has been confirmed both by our own tests and more generally for TREC (*TREC-SDR*). We conclude that

- retrieval from speech data can reach performance levels at least as good as those for the corresponding text reference standard; and also
- parallel text collections appear to be useful for expansion for spoken document retrieval, at least for broadcast news data.

4 Speech studies

The HTK large vocabulary speech recognition system used for the project is a well-established, leading-edge engine (for full details see *HTK*). It uses hidden Markov modelling with N-gram language models up to 4-grams and, ordinarily, a 65,000 word vocabulary. Processing uses multiple passes and automatically adapts to the speaker and acoustic conditions.

From the speech processing point of view, the Broadcast News data used for the project presented problems through both signal variation and noise. Recording and speech conditions could vary over narrow/broad band, monologue/dialogue, formal/informal, indoor/outdoor, female/male speaker, no music/music background/just music etc etc. Thus stories, i.e. stretches of material on the same topic, could be carried over several changes of speaker and environment, implying that the quality of transcription could also vary. At the same time it could not be assumed that any such change would necessarily signal a content change. (The detailed consequences for the BK and BU tests were somewhat different, since for BK content changes were independently marked and for BU had to be determined, but the general ones were the same.)

The system was developed and tuned for Broadcast News primarily for the independent DARPA CSR evaluations, so the TREC-8 version was superior to the earlier ones. It included segmentation for single speaker/acoustic conditions, with subsequent segment clustering to support adaptation; general rather than data-specific acoustic models; and data-specific language models derived from very large text corpora. This final project system, using a fast decoder from Entropic Ltd and developed jointly with Entropic, used 2 passes over the data with 108K vocabulary 4-grams for language modelling, and operated at 10 times real time, with a Word Error Rate (WER) of about 20% for the SDR material. (For further detail see *HTK* and *TREC-8*.)

Here we consider only the studies directly relevant to DR. These were on
a) vocabulary size effects, and
b) commercial elimination.

4.1 Vocabulary size

We considered this earlier in connection with expansion strategies. As noted there, in transcription *all* of the input (that survives initial filtering to remove e.g. music and possibly some commercials) is mapped to *some* word or other in the dictionary. Where the mapping is incorrect this has two effects: it increases the frequency of the 'wrong' word, and decreases that of the 'right' word (in fact things are more complicated because boundaries may also be misplaced). The special case is where an input word is not in the dictionary so there is a forced mapping to the wrong word: the system cannot say, as with text, that an input word is out of vocabulary (OOV).

The size of dictionary used can therefore be expected to affect retrieval performance, since a smaller dictionary is likely to imply more mismatches. However the impact on retrieval is influenced by whether a mismatch is for a function or a content word, whether a query term is a mismatch, and whether a mismatched word is rare or common (locally or globally). Mismatches for function words are clearly not serious for DR, mismatches for document words are less serious than for query words (though explicit vocabulary checks for query words can be easily made in practice), and mismatches for rare words perhaps more serious than for common. However it may be that, because query words tend to be more common words, mismatches have little real impact, even with a quite small vocabulary, and that expansion helps to counteract this. It is also the case that word form recognition interacts with stemming, so the net effect of misrecognition on retrieval may not be easily predicted from OOV rates, WER, etc.

As noted earlier for our retrieval tests, expansion is effective. Overall, the experiments exploring OOV effects reported in *SIGIR00* showed that, for the simple baseline on the T8 data, performance improves as the vocabulary grows from 3K to 27K words, rising from 22.2 to 43.0 MAP, with hardly any further improvement for a 55K word vocabulary: the full set of results, taken from *SIGIR00*, is shown in Figure 9.

This performance gain is correlated with a decrease in the story-averaged Term Error Rate from 68.8 for 3K to 36.7 for 27K, as measured on a file subset. (The Term Error Rate, defined in *ICASSP99*, characterises recogniser error in a more appropriate way for DR purposes that the standard simple Word Error Rate.) More importantly, the OOV rate for terms, and in particular query terms, fell from 21.4 to 1.9 for this file subset suggesting that, except with very small vocabularies, OOV is not a major problem. Indeed, as *SIGIR00* shows, expansion can raise performance for a small vocabulary to the baseline for a larger one: thus the combination of query and document expansion with the 7K vocabulary is noticeably better than the baseline performance for the 55K vocabulary.

Overall the conclusion on vocabulary size is

- a medium-sized vocabulary is adequate, and retrieval devices like expansion can com-

BK - story known condition

Vocabulary	Baseline	RBRF	UBRF	DBRF	DBRF +UBRF
S1					
3K	22.2	24.4	33.3	27.9	37.0
7K	33.8	37.5	44.3	38.6	46.6
14K	41.4	47.6	51.8	46.7	53.4
27K	43.0	49.7	53.8	48.4	55.8
55K	43.5	50.3	54.6	48.7	56.6
R1					
55K	47.9	54.4	56.1	51.1	56.3

Figure 9: Retrieval performance with different recognition vocabularies, for TREC-8 (taken from *SIGIR00*)

pensate for a small vocabulary.

4.2 Commercial elimination

Eliminating commercials is helpful both as a way of reducing the size of the file to be processed and in removing material that might generate unwanted matches in searching. We tried a number of strategies, exploiting different kinds of information, for doing this (*TREC-8, TREC-9*). The most effective was found to be looking for re-broadcast stretches of signal, using the method described in *ICASSP00*, since commercials typically recur, at least over a series of broadcasts from the same show if not within a single broadcast. However since real stories may also be re-broadcast over different bulletins, steps had to be taken to avoid removing broadcast content by accident. (Because the TREC evaluations treated repeated stories as separate items, story repeats were not removed, though in reality this might be appreciated by the user.) The experiments with the TREC-9 data showed that it was possible to remove over 50% of the commercials for a negligible loss of less than 1% of the real content, while reducing the file by about 13%.

Our experiments thus show that

- quite straightforward processing strategies can remove unwanted commercials from news data;
- removing unwanted material helps retrieval effectiveness as well as reducing recogniser effort.

Removing commercials can also be used to identify story boundaries in the BU case, since commercials typically occur between rather than within news stories, but more work is needed to explore this in combination with other means of obtaining natural content units in searching.

5 The MDR Demonstration System

The project work has included building a demonstration system (*RIAO00b, IJST01a*). This is a Web-based application for searching automatically-generated transcriptions of online news broadcasts, based on the TREC-8 SR and DR models. However since the system is interactive, semantic posets are exploited to offer the user additional terms for their query, and relevance feedback can be applied 'for real' via the user's assessments of search results. The ranked system output is displayed as extracts with query terms highlighted, and the user can also listen to the corresponding audio or view the full transcript.

6 Conclusion

The project successfully achieved both its specific goals and the more general one of demonstrating SDR. The main limitations were in the size of collection used, in the abstraction from a fully operational context, and in the impossibility of exploring, with the TREC data, the issue of full multimedia retrieval.

However, subject to these limitations, the project showed that

- Retrieval from transcribed speech can be done as effectively as from corresponding correct text; performance is relatively impervious to speech data imperfections and speech system conditions like recogniser vocabulary size.
- Retrieval strategies using expansion, and particularly query expansion, are valuable; and they may be assisted by exploiting information about words drawn from large parallel text files.

Acknowledgements

This work was supported by the UK Engineering and Physical Science Research Council under Grant GR/L49611. We are also very grateful to the relevant staff at NIST, especially Ellen Voorhees and John Garofolo, for all their TREC SDR effort.

References

ATT: A. Singhal and F.Pereira, 'Document expansion for speech retrieval', *SIGIR-99, Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, 34-41.

ESCA99: P. Jourlin, S.E. Johnson, K. Spärck Jones and P.C. Woodland, 'General query expansion techniques for spoken document retrieval', *Proceedings of the ESCA Workshop on Extracting Information from Spoken Audio*, Cambridge, 1999, 8-13.

EVAL: see results appendices in TREC.

FREFL: K. Spärck Jones, 'Further reflections on TREC', *Information Processing and Management*, 36, 2000, 37-85.

HTK: see the publications listed at
`http://htk.eng.cam.ac.uk/docs/cuhtk.shtml`

ICASSP99 S.E. Johnson, P. Jourlin, G.L. Moore, K. Spärck Jones, and P.C.Woodland, 'The Cambridge University spoken document retrieval system', *Proceedings of ICASSP '99*, Vol. 1, 1999, 49-52.

ICASSP00 S.E. Johnson and P.C. Woodland, 'A method for direct audio search with applications to indexing and retrieval', *Proceedings of ICASSP 2000*, Vol. 3, 2000, 1427-1430.

IJST01a A. Tuerk, S.E. Johnson, P. Jourlin, K Spärck Jones and P.C. Woodland, 'The Cambridge University multimedia document retrieval demo system', *International Journal of Speech Technology*, 2001 (in press).

IJST01b S.E. Johnson, P. Jourlin, K Spärck Jones and P.C. Woodland, 'InformationrRetrieval from unsegmented broadcast news audio', *International Journal of Speech Technology*, 2001 (in press).

IPM00: K. Spärck Jones, S. Walker and S.E. Robertson, 'A probabilistic model of information retrieval: development and comparative experiments, Parts 1 and 2', *Information Processing and Management*, 36, 2000, 779-840.

RIAO00a: S.E. Johnson, P. Jourlin, K. Spärck Jones and P.C. Woodland, 'Audio indexing and retrieval of complete broadcast news shows', *RIAO '2000, Content-Based Multimedia Information Access, Conference Proceedings*, Paris: Centre de Hautes Etudes Internationales d'Informatique Documentaire (CID), 2000, 1163-1177.

RIAO00b: A. Tuerk, S.E, Johnson, P. Jourlin, G.L. Moore, K. Spärck Jones and P.C. Woodland, 'The Cambridge University Multimedia Document Retrieval Demo System', *RIAO '2000, Content-Based Multimedia Information Access, Conference Proceedings*, Paris: Centre de Hautes Etudes Internationales d'Informatique Documentaire (CID), 2000, Vol. 3, Applications, 14-15.

SIGIR00: P.C. Woodland, S.E. Johnson, P. Kourlin and K. Spärck Jones, 'Effects of out of vocabulary words in spoken document retrieval', *SIGIR-2000, Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000, 372-374.

SPCOMM00: P. Jourlin, S.E. Johnson, K. Spärck Jones and P.C. Woodland: 'Spoken document representations for probabilistic retrieval', *Speech Communication*, 32, 2000, 21-36.

TR446: K. Spärck Jones, S. Walker and S.E. Robertson, *A probabilistic model of information retrieval : development and status*, Technical Report 446, Computer Laboratory, University of Cambridge, 1998.

TREC: E.M. Voorhees and D.K. Harman (Eds), *The Eighth Text REtrieval Conference (TREC-8)*, NIST Special Publication 500-246, National Institute of Standards and Technology, Gaithersburg MD, 2000.

TREC-7: S.E. Johnson, P. Jourlin, G.L. Moore, K. Spärck Jones and P.C. Woodland, 'Spoken document retrieval for TREC-7 at Cambridge University', *The Seventh Text REtrieval Conference (TREC-7)* (Ed. Voorhees and Harman), NIST Special Publication 500-242, National Institute of Standards and Technology, Gaithersburg MD, 1999, 191-200.

TREC-8: S.E. Johnson, P. Jourlin, K. Spärck Jones and P.C. Woodland, 'Spoken document retrieval for TREC-8 at Cambridge University' *The Eighth Text REtrieval Conference (TREC-8)*, (Ed. Voorhees and Harman), NIST Special Publication 500-246, National Institute of Standards and Technology, Gaithersburg MD, 2000, 197-206.

TREC-9: S.E. Johnson, P. Jourlin, K. Spärck Jones and P.C. Woodland, 'Spoken document retrieval for TREC-9 at Cambridge University' *Proceedings TREC-9*, National Institute of Standards and Technology, Gaithersburg MD, in press.

TREC-SDR: J.S. Garofolo, C.G.P. Auzanne and E.M. Voorhees, 'The TREC spoken document retrieval track: a success story', *RIAO '2000, Content-Based Multimedia Information Access, Conference Proceedings*, Paris: Centre de Hautes Etudes Internationales d'Informatique Documentaire (CID), 2000, Vol. 3, Applications, 1-20.

VMR: G.J.F. Jones, J.T. Foote, K. Spärck Jones and S.J. Young, 'Retrieving spoken documents by combining multiple index sources', *SIGIR-96, Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996, 30-38.

for further publication details see
http://www-svr.eng.cam.ac.uk/Research/Projects/Multimedia_Document_Retrieval/