SPOKEN DOCUMENT RETRIEVAL FOR TREC-8 AT CAMBRIDGE UNIVERSITY

S.E. Johnson†, P. Jourlin‡, K. Spärck Jones‡ & P.C. Woodland†

†Cambridge University Engineering Department,Trumpington Street, Cambridge, CB2 1PZ, UK.
Email: {sej28,pcw}@eng.cam.ac.uk
‡Cambridge University Computer Laboratory, Pembroke Street, Cambridge, CB2 3QG, UK.
Email: {pj207,ksj}@cl.cam.ac.uk

ABSTRACT

This paper presents work done at Cambridge University on the TREC-8 Spoken Document Retrieval (SDR) Track. The 500 hours of broadcast news audio was filtered using an automatic scheme for detecting commercials, and then transcribed using a 2-pass HTK speech recogniser which ran at 13 times real time. The system gave an overall word error rate of 20.5% on the 10 hour scored subset of the corpus, the lowest in the track. Our retrieval engine used an Okapi scheme with traditional stopping and Porter stemming, enhanced with part-of-speech weighting on query terms, a stemmer exceptions list, semantic 'poset' indexing, parallel collection frequency weighting, both parallel and traditional blind relevance feedback and document expansion using parallel blind relevance feedback. The final system gave an Average Precision of 55.29% on our transcriptions.

For the case where story boundaries are unknown, a similar retrieval system, without the document expansion, was run on a set of "stories" derived from windowing the transcriptions after removal of commercials. Boundaries were forced at "commercial" or "music" changes and some recombination of temporally close stories was allowed after retrieval. When scoring duplicate story hits and commercials as irrelevant, this system gave an Average Precision of 41.47% on our transcriptions.

The paper also presents results for cross-recogniser experiments using our retrieval strategies on transcriptions from our own first pass output, AT&T, CMU, 2 NIST-run BBN baselines, LIMSI and Sheffield University, and the relationship between performance and transcription error rate is shown.

1. INTRODUCTION

The TREC-7 Spoken Document Retrieval (SDR) Track showed that successful retrieval of information where the original source of the documents is audio is possible for small collections [4, 5]. The results showed that although retrieval performance degraded when recogniser performance worsened, the fall off was rather gentle and good retrieval can still be achieved on transcriptions with over 100% Processed Term Error Rate [10], corresponding to 66% Word Error Rate (WER) [11]. Further work has shown that various extensions to our retrieval system can increase performance across the whole range of error rates, with an Average Precision (AveP) of 55.88 obtained on reference transcriptions, 55.08 on our own transcriptions (24.8% WER) and 44.15 on transcriptions from DERA [17] (61.5% WER) on the TREC-7 task [15].

Although by speech recognition standards, the 100 hour test data for TREC-7 represented a large task, the 2866 stories and 23 queries provided only a small collection to test retrieval systems. The conclusions which could be drawn about SDR were therefore limited and a larger collection was needed to confirm the results. The 500 hours of TREC-8 data, with 21,754 stories and 50 queries, represents such a collection and the results presented in this paper show how our methods adapt to a larger task.

An additional feature of our TREC-8 system is that no knowledge about story boundaries is used for recognition, and two retrieval runs are made for each set of transcriptions. For the first run, manual "story" boundaries have been added and commercials have been manually removed (*story-known*) whilst for the second, no such information was used and the retrieval system attempted to find relevant passages in the document collection (*story-unknown*). This led to added challenges in recognition as well as retrieval, with a pre-processing stage being added to remove some data automatically labelled as commercials before recognition began.

This paper firstly describes the TREC-8 SDR tasks and the data used in both development and evaluation of our TREC-8 SDR system. The commercial-detection scheme and the speech recogniser are described in detail in sections 2 and 3 respectively, with the performance of all the sites participating in the cross-recogniser runs also given in the latter. The retrieval engine is then described in section 4, along with a detailed analysis of how the individual retrieval components interacted and affected the overall results. Section 5 focuses on the development of the *story-unknown* system using concatenated TREC-7 data and describes the final evaluation system, giving the results for the TREC-8 task. Cross-recogniser experiments are presented in section 6, where the influence of transcription quality on both the story-known and story-unknown tasks is investigated. Finally, conclusions are offered in section 7.

1.1. Description of TREC-8 SDR Tasks

The TREC-8 SDR track contains two main tasks. The first, story-known (SK) SDR, is similar to the TREC-7 SDR track, with audio from American broadcast radio and TV news programs provided along with a list of manually-generated *story* (or document) boundaries. Natural language *text* queries, such as "`What natural disasters occurred in the world in 1998 causing at least 10 deaths?`" are then provided and participating sites must submit a ranked list of potentially relevant documents after running a recognition and retrieval system on the audio data. Real relevance assessments generated by humans are then used to evaluate the ranked list in terms of the standard IR measures of precision and recall. For TREC-8, sites may also run their retrieval system on a "reference" transcription which uses manually-generated closed-caption data, and on other automatically generated transcriptions from NIST (*baselines*) or from other participating sites (*cross-recogniser*).

The second TREC-8 task assumes no knowledge of the story boundaries at both recognition and retrieval time (story-unknown case). The end points of the shows are given as the start time of the first "story" and end time of the last "story" but no other story information, including the location of commercial breaks within the show, can be used. Retrieval then produces a ranked list of shows with time stamps, which are mapped in the scoring procedure to their corresponding story identifiers (IDs). All but the first occurrence of each story is marked irrelevant, as are commercials, before the standard scoring procedure is applied.

For both tasks in TREC-8, the recognition is an *on-line* task, i.e. for any given audio show, only data and information derived from before the day of broadcast can be used. Therefore, unlike for TREC-7, unsupervised adaptation on the test collection can only use data up to and including the current day. Retrieval however is *retrospective* and can use any data up until the last day of the document collection (June 30th 1998). Further details can be found in the TREC-8 specification [6].

1.2. Description of Data

There are two main considerations when describing the data for SDR. Firstly the audio data used for transcription, and secondly the query/relevance set used during retrieval. Table 1 describes the main properties of the former, whilst Table 2 describes the latter, for the *development* (TREC-7) and *evaluation* (TREC-8) data sets.[1]

	TREC-7 (dev)	TREC-8 (eval)
Nominal Length of Audio	100 hours	500 hours (SU)
Number of Documents	2,866	21,754 (SK)
Approx. Number of Words	770,000	3,900,000 (SK) 4,700,000 (SU)
Average Doc length	269 words	180 words (SK)

Table 1: Description of data used

[1]Only 49 of the 50 queries for TREC-8 were adjudged to have relevant documents within the TREC-8 corpus

	TREC-7 (dev)	TREC-8 (eval)
Number of Queries	23	50
Average Length of Query	14.7 words	13.7 words
Mean # Rel Docs per Query	17.0 docs	36.4 docs (SK)

Table 2: Description of query and relevance sets used

2. AUTOMATIC DETECTION OF COMMERCIALS

To enable both the case of known and unknown story boundary SDR to be investigated, the recognition must be run on all of the 500 hours of audio without using any knowledge of the story boundaries. Since a substantial portion of the data to be transcribed was known to be commercials and thus irrelevant to broadcast news queries, an automatic method of detecting and eliminating such commercials would potentially reduce the number of false matches, thereby increasing the precision of the overall system. Removing commercials early on in processing would also reduce the amount of data that needed to be transcribed and hence speed up the overall recognition system. The first stage of our SDR system was thus a commercial detector designed to eliminate automatically some sections of audio thought to correspond to commercials, whilst retaining all the information-rich news stories.

2.1. Development on TREC-7

The commercial detector was based on finding segments of repeated audio using a direct audio search (described in [12]), making the assumption that (usually) only commercials are repeated. Experiments were performed on the 8.7 hours of TREC-7 SDR data from ABC by searching for segments of similar audio within the data. The results from using 2 sliding window systems with length L and skip S to generate the initial segments are given in Table 3 along with a system which uses the automatically generated wideband segments from our 1997 Hub-4 segmenter [7]. Since the segmentation and commercial detection processes interact, results after both stages are given.

Segments Generation	Cut-Off	Alone		+Segmentation	
		Non-Story	Story	Non-Story	Story
Automatic WB segs	low	31.59%	0.00%	46.27%	1.30%
	medium	36.94%	0.01%	51.62%	1.31%
	high	39.97%	0.24%	54.65%	1.54%
Slide L=10s S=2s	low	**59.41%**	**0.17%**	68.35%	1.38%
	medium	62.45%	0.39%	70.27%	1.56%
	high	64.53%	0.57%	71.83%	1.69%
Slide, L=20s S=4s	low	50.30%	0.05%	58.72%	1.33%
	medium	57.75%	0.38%	65.21%	1.60%
	high	63.92%	1.25%	70.41%	2.44%

Table 3: Proportion of story/non-story rejected by direct search on coded audio for TREC-7 ABC data

A low cut-off threshold on the shorter window-length system was chosen to maximise the rejection of commercials whilst keeping the rejection rate of genuine stories below 0.2%. The effect of relabelling segments shorter than a certain smoothing length, T_s, which appeared between two segments labelled as commercials was investigated, with the results given in Table 4.

This shows that smoothing for up to a minute between detected commercial segments increases the performance of the commercial rejection system.

T_s (s)	Non-story Rejection	Story Rejection
0 (none)	59.41%	0.17%
30	62.31%	0.17%
60	70.90%	0.17%
90	73.34%	0.45%

Table 4: Effects of smoothing on TREC-7 ABC data

These general principles were used in the design of the TREC-8 system, but some changes and additions were made to reflect the different nature of the TREC-8 story unknown task: for example, only data broadcast before the current day can be used to identify commercials in TREC-8.

2.2. The TREC-8 System

In a more realistic scenario, the user is not likely to be interested in retrieving information which has been re-broadcast, (i.e. *repeats*) whether it be a commercial or a news story. However, the TREC-8 evaluation set-up meant it was better to retain segments containing news content even if they were repeats, whilst eliminating those repeated segments which correspond to commercials. Safeguards were therefore added to try to reduce the probability of any matching audio which was not a commercial being falsely rejected during the commercial detection stage.

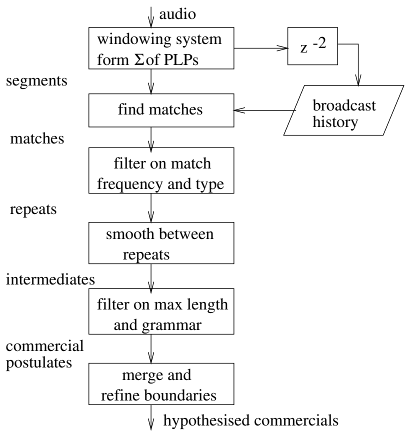

Figure 1: The commercial detection process

A block diagram of the commercial detection process used for the TREC-8 evaluation is given in Figure 1. Audio of the current show was analysed into 5 second windows with a window shift of 1s. Each window was characterised by the covariance matrix of the (wideband) PLP cepstral coefficients as used in the subsequent speech recognition passes. A broadcast history was built up which consisted of the windows for a certain amount of broadcast data (typically 20 hours) from that broadcaster, running up to a few days before the date of the current show. The delay was introduced to reduce the probability of an actual news story occurring in the broadcast history being directly re-broadcast in the current show. The broadcast history was initialised using the January 1998 TDT-2 data and rolled through the TREC-8 SDR evaluation data as the data was processed.

Each segment in the current show was then compared to the segments in the broadcast history. If the arithmetic harmonic sphericity distance [1] between the covariance matrices of the segments was less than a threshold, then the pair was marked as "matching". Note that a non-zero threshold was necessary, even when looking for identical audio, since there is no guarantee that the sampling and window shifts in each case are synchronous with the audio event in question.

For a segment to be marked as a true repeat, the number of matches between the segment and the broadcast history had to be above a given threshold, to reduce the number of false alarms due to similar, but not identical audio (for example for segments which overlapped by say 80%) matching erroneously. The probability of a re-broadcast story being labelled as a repeat was further reduced by defining the number of different days in the broadcast history which must be involved in the match before the segment was accepted as a repeat.

The merging process was then applied which relabelled as *intermediates* any small gaps which occurred between two segments already labelled as *repeats*. The intermediates were then relabelled as commercials, only if the resulting smoothed "commercial" was less than a critical length, the repeats always being relabelled as commercials. For the CNN shows a show "grammar" (constructed from the CNN TREC-7 data) was used to constrain the locations in the audio that could be labelled as commercials. Due to the limited time resolution of the commercial labelling process, conservative start and end points were also used.

2.3. Results for the TREC-8 System

Since the audio was eliminated at an early stage and could not be recovered later during processing, a very conservative system, `COMM-EVAL`, which removed 8.4% of the audio, was used for the evaluation. A contrast run, `COMM-2`, which removed 12.6% of the audio, was later made to see the effect of relaxing the tight constraints on the system. The breakdown of data removed using these systems compared to the manually generated story labels is given in Table 5. Note that these "reference" labels are not an exact reflection of the story/commercial distinction, since a few commercials have been wrongly labelled as stories and some portions of genuine news have not had story labels added and hence are erroneously scored as commercials; however they offer a reasonable indicator of the performance of the commercial detector within the context of this evaluation.

The results show that automatic commercial elimination can be performed very successfully for ABC news shows. More false rejection of stories occurs with CNN data, due to the frequency of short stories, such as sports reports, occurring between commercials. The amount of commercial rejection with the VOA data is low, due mainly to the absence of any VOA broadcast history from before the test data. However, overall the scheme worked well, since 97.8% of the 42.3 hours of data removed by the `COMM-EVAL` system (and 95.0% of the 63.4 hours removed by the contrast `COMM-2` run) were labelled as non-story in the reference.

	Broad.	Non-Stories	Stories	Total
COMM EVAL	CNN	26.19hr=35.7%	2822s=0.46%	27.0hrs=11.0%
	ABC	12.78hr=65.5%	28s=0.02%	12.8hrs=20.5%
	PRI	1.93hr=16.6%	297s=0.10%	2.0hrs= 2.2%
	VOA	0.47hr= 5.0%	132s=0.04%	0.5hrs= 0.5%
	ALL	**41.4hrs=36.3%**	**0.9hrs=0.2%**	**42.3hrs=8.4%**
COMM - 2	CNN	43.26hr=59.0%	10640s=1.73%	46.2hrs=18.9%
	ABC	13.78hr=70.6%	107s=0.07%	13.8hrs=22.1%
	PRI	2.60hr =22.4%	416s=0.14%	2.7hrs= 2.9%
	VOA	0.56hr= 6.0%	208s=0.06%	0.62hrs= 0.6%
	ALL	**60.2hrs=52.9%**	**3.2hrs=0.81%**	**63.4hrs=12.6%**

Table 5: Amount of data rejected during commercial elimination

3. THE TREC-8 HTK BROADCAST NEWS TRANSCRIPTION SYSTEM

After the commercial detection and elimination, the data is automatically segmented and classified by bandwidth and gender. The segmenter initially classifies the data as either wideband (WB) speech, narrowband (NB) speech or pure music/noise, which is discarded. The labelling process uses Gaussian mixture models and incorporates MLLR adaptation. A gender-dependent phone recogniser is then run on the data and the smoothed gender change points and silence points are used in the final segmentation. Putative segments are clustered and successive segments in the same cluster are merged (subject to the segment length remaining between 1 and 30 seconds). The TREC-8 segmenter, which ran in approximately 0.75x real time, included a revised mixture model for music and applied new insertion penalties, but is essentially similar to the system described in [7] with the modifications for faster operation from [18].

Since silence, music and noise are discarded during segmentation, it is interesting to note the interaction between this stage and the commercial elimination phase. The results, given in Table 6, show that the proportion of data discarded by the segmenter decreases from 9.5% to 7.4% if applied after the commercial elimination stage.

	before seg.	after seg.
Original	502.4	454.6
Commercial Elim	460.2	$\boxed{426.0}$

Table 6: Number of hours of audio retained during processing

The main transcription system used a continuous mixture density, tied-state cross-word context-dependent HMM system based on the CUHTK-Entropic 1998 Hub4 10xRT system [18]. The speech was coded into 13 static cepstral coefficients (including C0) and their first and second derivatives. Cepstral mean normalisation was applied over each segment. After commercial detection and segmentation, a 2-pass recognition system was applied. The initial transcription pass through the data, denoted `CUHTK-p1`, used gender-independent, bandwidth-specific triphone models, with a 60,000 word 4-gram language model to produce a single best hypothesis. The gender of each segment was then labelled by choosing the most likely alignment of this transcription using male and female HMMs. Top-down covariance-based clustering [9] was then applied on a gender and bandwidth specific basis to all the segments broadcast on a given day and MLLR transforms were generated for these clusters using the first pass transcriptions.

The second pass used the MLLR-adapted gender-dependent triphone models with a 108,000 word 3-gram mixture language model to generate lattices from which a one-best output was generated using a 4-gram model. This transcription, denoted `CUHTK-s1u`, was used for the story-unknown retrieval experiments, whilst the story-known transcription, `CUHTK-s1`, was simply generated by filtering this output using the known story boundaries. The overall system gave a word error rate of 15.7% on the November 1998 Hub4 evaluation data and 20.5% on the 10-hour scored subset of the TREC-8 evaluation data and runs in about 13xRT on a single processor of a dual processor Pentium III 550MHz running Linux.

The HMMs were trained using 146 hours of broadcast news audio running up to 31st January 1998, supplied by the LDC and used for the 1998 Hub-4 task. The gender-independent wideband models were generated initially, then narrowband models were created by single pass retraining using a band-limited (125Hz to 3750Hz) analysis. Gender-specific models were generated using a single training iteration to update the mean and mixture weight parameters.

Three fixed backoff word-based language models were trained, from broadcast news text, newspaper texts and acoustic transcriptions, which were all generated using data from before 31st January 1998. The first model was built using 190 million words of broadcast news text, covering 1992-1996 (supplied by the LDC), Nov. 1996 to Jan. 1998 (from the Primary Source Media Broadcast News collection) and Jan. 1998 (from the TDT-2 corpus transcriptions). The LDC also supplied the 70m words from the Washington Post and Los Angeles Times covering 1995 to Jan. 1998, which were used for the newspaper texts model. The third model was built using 1.6m words from the 1997 and 1998 acoustic training transcriptions and 1995 Marketplace transcriptions. Single merged word based models were created which resulted in effectively interpolating the three models, forming a single resultant language model. The final 60k language model had 6.0m bigrams, 14.6m trigrams and 9.4m 4-grams, whilst the 108k model had 6.2m, 14.8m and 9.4m respectively.

3.1. WER Results from Cross-Recogniser Runs

As well as our own transcriptions (*CUHTK-s1*) we used several alternative sets to assess the effect of error rate on retrieval performance. These came from manually generated closed-captions, both unprocessed (*cc-unproc*) and with some standard text processing of numbers, dates, money amounts and abbreviations (*cc-proc*); two baselines produced by NIST using the BBN Rough'N'Ready transcription system, (*NIST-B1* and *NIST-B2*), including a fixed and dynamically updated language model respectively; transcriptions from recognisers from LIMSI, Sheffield University, AT&T, and Carnegie Mellon University (*CMU*); and the output of the first pass of our system (*CUHTK-p1*).

A 10-hour subset of the TREC-8 (story-known) evaluation data was taken and detailed transcriptions made by the LDC for scoring the recognisers. The results are given in Table 7.

Recogniser	Corr.	Sub.	Del.	Ins.	Err
cc-proc	92.7	2.4	4.9	1.5	8.8
cc-unproc	88.8	4.1	7.1	1.2	12.4
CUHTK-s1	82.4	14.0	3.7	2.9	**20.5**
LIMSI	82.0	14.6	3.4	3.5	21.5
CUHTK-p1	77.3	18.5	4.2	3.9	26.6
NIST-B2	76.5	17.2	6.2	3.2	26.7
NIST-B1	75.8	17.8	6.4	3.3	27.5
AT&T	75.8	20.4	3.8	5.1	29.3
Sheffield	71.9	22.0	6.1	3.9	32.0
CMU	39.6	28.1	32.3	4.0	64.4

Table 7: WER on 10 hour subset of TREC-8 evaluation data

The results show that the `CUHTK-s1` automatic transcriptions are very good, suggesting that the error rate, though some distance from that for the manually-generated closed caption transcriptions, is still low enough not to degrade retrieval performance substantially. It is pleasing to note that the relatively simple `CUHTK-p1` system, which uses a smaller vocabulary, has no adaptation and runs in around 3 times real time, gives a reasonably low word error rate.

4. RETRIEVAL SYSTEM

The basic system we used for SK retrieval in TREC-8 is similar to that presented at TREC-7 [11], but the final system also contains several new devices. These include Semantic Poset Indexing (SPI) and Blind Relevance Feedback for query expansion, both on the test collection itself (BRF) and a parallel corpus (PBRF), all of which have been shown to increase performance on the TREC-7 task [14, 15]. A new technique called Parallel Collection Frequency Weighting (PCFW) is also presented along with an implementation of document expansion using the parallel corpus within the framework of the Probabilistic Model.

4.1. System Description

4.1.1. Preprocessing

A term t_i is a set of words or word sequences from queries or documents which are considered to be a unique semantic unit. We call the first set of operations which define the relationship between terms and their components *preprocessing*. The following preprocessing techniques are sequentially applied on all transcriptions and queries before indexing and retrieval.

The words are first made lower case and some punctuation characters are removed. Hyphens and digital numbers were kept even though they do not occur in the ASR-transcribed documents.[2] Some sequences of words are then mapped to create single compound words. and some single-word mappings are also applied to deal with known stemming exceptions and alternative (possibly incorrect) spellings in the manual transcriptions. The list of compound words and mappings was created manually for our TREC-7 SDR system [11]. A set of non-content (stop) words was removed from all documents and queries, with an additional set also being removed from just the queries, e.g. {`find,documents,..`}. Abbreviations, (in several forms) are mapped into single words, e.g. `[C. N. N. -> cnn]`.

The use of Porter's well-established stemming algorithm [19] allows several forms of a word to be considered as a unique term, e.g. $t_i(train)$ = {train, training, trainer, trains, ...}. Unlike the mapping techniques, this algorithm is not limited by the use of a fixed thesaurus and therefore every *new* word in a test collection can be associated with its various forms.

4.1.2. Indexing

The *index* (inverted) file contains all the information about a given collection of documents that is needed to compute the document-query scores. For the collection, each term t_i in the term-vocabulary has an associated:

- *collection number* $n(t_i)$: the number of documents which at least one of the components of t_i occurs in.
- list of *term frequencies* $tf(t_i, d)$, which is the number of occurrences of all of the components of t_i in document d.

The index file also contains the number of documents in the collection, N, and the length of each document $dl(d_j)$.

Semantic Poset Indexing (SPI) [14] is used to allow $tf(t_i, d)$ and $n(t_i)$ to take into account some semantic relationships between terms. More specifically, semantic poset structures based on unambiguous noun hyponyms from WordNet [2] and a manually-built geographic locations tree were made. A term occurring in a poset is then redefined as the union of itself and all more specific terms in the poset associated with that term, before the statistics are calculated. For example, the term frequency for a term t_i thus becomes the sum of the frequencies of occurrence of itself and all more specific related terms within a given document.

4.1.3. Retrieval

A part-of-speech (POS) tagger is run over the queries and the weight of each query term t_i is scaled by a factor $pos(t_i)$ using the POS weighting scheme from our TREC-7 system [11]. The score for a document with respect to a given query is then obtained by summing the combined weights, $cw(t_i, d_j)$, for each query term t_i according to the following formulae:

$$cw(t_i, d_j) = \frac{pos(t_i) \cdot (\log N - \log n(t_i)) \cdot tf(t_i, d_j) \cdot (K+1)}{K \cdot (1 - b + b \cdot ndl(d_j)) + tf(t_i, d_j)}$$

$$n(t_i) = \sum_{d_i \in D} \begin{cases} 0 & tf(t_i, d_i) = 0 \\ 1 & tf(t_i, d_i) > 0 \end{cases}$$

$$dl(d_j) = \sum_{w \in V} tf(w, d_j) \qquad ndl(d_j) = \frac{dl(d_j) \cdot N}{\sum_{d \in D} dl(d)}$$

[2] One might think that some hyphens should be removed from the manually transcribed documents (e.g. health-related) whereas others should not (e.g. anti-abortion). Because of a lack of preliminary experiments we decided not to remove any hyphens or digits.

where V is the term vocabulary for the whole document collection D; and K and b are tuning constants

4.1.4. Blind Relevance Feedback (BRF)

When the documents in the collection are ranked according to a given query, it is possible to expand the query by adding several terms which occur frequently within the top documents but rarely within the whole collection. The T terms which obtain the highest Offer Weight are added to the query. The Offer Weight of a term t_i is :

$$ow(t_i) = r \cdot \log \left[\frac{(r+0.5)(N-n-R+r+0.5)}{(n-r+0.5)(R-r+0.5)} \right]$$

where R is the number of top documents which are assumed to be relevant; r the number of assumed relevant documents in which at least one component of t_i occurs; n the total number of documents in which at least one component of t_i occurs; and N is the total number of documents in the collection.

4.1.5. Document Parallel Blind Relevance Feedback (DPBRF)

The method of document expansion described within the Vector Model in [20] at TREC-7, can also be used within the probabilistic framework. By considering a document as a *pseudo-query*, it is possible to expand that document using BRF on a parallel collection. For a given document, the 100 terms with the lowest $n(t_i)$ are used as the pseudo-query. BRF is then applied on the parallel collection (with $R = 10$) and the top 400 terms are added to the original document with a term frequency based on their Offer Weight.

4.1.6. Parallel Collection Frequency Weighting (PCFW)

If the test collection is small or contains many transcription errors, the values of $n(t_i)$ may not be sufficiently reliable to use in the prediction of relevance. It is possible to exploit the larger, higher quality parallel collection to obtain better estimates for $n(t_i)$ (and N), to use within the combined weights formula. The collection number, $n(t_i)$, for a given term is therefore replaced by the sum of the collection number for that term on the test corpus and the parallel corpus; with the number of documents, N, being adjusted accordingly.

4.1.7. The Final System

The index file was made as follows:

1. Preprocess & apply SPI to the test collection to give I_t
2. Preprocess & apply SPI to parallel collection to give I_p
3. Perform DPBRF using the pseudo queries from the test collection documents on I_p and add the new terms into the index file I_t.
4. Replace the collection frequency weights in I_t with the PCFWs derived from I_t and I_p and update N accordingly.

The query file was produced by:

1. Preprocess the original natural language request file and attach a POS weight (POSW) to each query term.
2. Perform PBRF using I_p and add the new terms to the query.
3. Perform BRF on I_t and add the new terms to the query.

4.1.8. The Parallel Collection

The parallel collection used in DPBRF, PBRF and PCFW is composed of $51,715$ stories extracted from the L.A. Times, Washington Post and New York Times over the period of Jan 1st to June 30th 1998. This contains the TREC-8 SDR test collection period (Feb 1st to June 30th 1998).

4.2. Experiments on TREC-8 SK SDR

The AveP results for our final system on all the sets of transcriptions made available is given in Table 13 in section 6. Here we concentrate on the effect on performance of each aspect of the system on our own CUHTK-s1 transcriptions.

4.2.1. Results on the CUHTK-s1 Transcriptions

It is important to try to understand the contribution of each individual device towards the overall performance of the IR system. Table 8 gives the values of AveP we obtain by progressively decomposing the system.

Lines 1 and 2 show that the addition of all these devices together led to a relative increase in AveP of 23%. Lines 3-5 show that adding just PBRF or BRF individually improve the performance over a system with no blind relevance feedback, but applying PBRF alone gives better results than their combination.

Lines 6-11 show that the improvement due to PCFW is reduced by the use of PBRF. BRF degrades the performance even more when PCFW is present. A similar behaviour can be observed on lines 12-15 for POSW, namely that adding POSW increases performance on the basic system, but degrades when all the other devices are also included. However, this is not the case for DPBRF, as lines 16-17 show that including DPBRF when all other devices are present increases AveP by 5.7% relative.

SPI exhibits a rather different behaviour. It has no significant effect on the baseline system (see lines 18-19), but since the parallel corpus was indexed with SPI, all the devices apart from POSW were affected by the use of this technique. Lines 20 and 21 show that AveP reached 56.72% when SPI was not used and thus SPI actually degraded the performance by 2.5% relative. By comparing lines 20 and 22, we can see that the poor contribution of BRF was due to the inclusion of SPI.

In summary, the inclusion of the techniques discussed increased AveP by 23% relative. Some interaction between the devises was found and it was noted that an AveP of 56.72% could be achieved if SPI had not been included. The corresponding AveP on the processed closed-caption data was 57.66%.

5. THE STORY-UNKNOWN (SU) SYSTEM

For the SU evaluation, no knowledge of the manually-labelled story boundaries can be used either in retrieval or recognition. The system must present a ranked list of show:time stamps, which are mapped to the corresponding story (or commercial) IDs before retrieval performance evaluation, with commercials and duplicates scored as irrelevant.

	SPI	DPBRF	PCFW	POSW	PBRF	BRF	AveP	P@30
1	-	-	-	-	-	-	44.96	35.17
2	Y	Y	Y	Y	Y	Y	55.29	40.34
3	Y	Y	Y	Y	-	-	50.63	38.16
4	Y	Y	Y	Y	Y	-	55.69	41.16
5	Y	Y	Y	Y	-	Y	54.27	39.66
6	Y	Y	-	Y	-	-	49.50	37.28
7	Y	Y	Y	Y	-	-	50.63	38.16
8	Y	Y	-	Y	Y	-	55.61	41.43
9	Y	Y	Y	Y	Y	-	55.69	41.16
10	Y	Y	-	Y	Y	Y	55.32	40.41
11	Y	Y	Y	Y	Y	Y	55.29	40.34
12	-	-	-	-	-	-	44.96	35.17
13	-	-	-	Y	-	-	45.90	35.65
14	Y	Y	Y	-	Y	Y	55.49	40.95
15	Y	Y	Y	Y	Y	Y	55.29	40.34
16	Y	-	Y	Y	Y	Y	52.28	38.10
17	Y	Y	Y	Y	Y	Y	55.29	40.34
18	-	-	-	-	-	-	44.96	35.17
19	Y	-	-	-	-	-	44.99	34.90
20	-	Y	Y	Y	Y	Y	**56.72**	**42.72**
21	Y	Y	Y	Y	Y	Y	55.29	40.34
22	-	Y	Y	Y	Y	-	55.99	42.31

Table 8: Breakdown of results on the CUHTK-s1 transcriptions showing different combinations of the retrieval techniques

Two main approaches to the SU task exist, the first consists of labelling story boundaries automatically and then running the standard retrieval engine; whilst the second never explicitly finds the story boundaries, but rather locates the relevant passages in the transcriptions and performs some merging of temporally close relevant passages to reduce the possibility of producing multiple hits from the same story source. We investigated one technique from each approach, namely Hearst's text-tiling [8] for topic boundary detection and a windowing/ recombination system.

For development, the 100 hours of TREC-7 SDR test data was used. This did not exactly model the TREC-8 SU task, since the commercials had already manually been removed from the data, but offered a reasonable basis to compare the different systems. Two methods of scoring were used, the first is the official evaluation scoring procedure, where all instances of a story other than the first one are scored as irrelevant (named *dup-irrel*). The second, by removing all duplicates before scoring, was more lenient and provided an indication of the "best" performance that could be achieved if a perfect merging system (that removed duplicates, but did not re-score or re-order the ranked list) were added after retrieval. This was named *dup-del* and represents a reasonable indication of the potential of any given system.

A simple experiment was conducted to compare a text-tiling system with a windowing system. Text-tiling was originally designed to group paragraphs in long textual reports together and therefore is not ideally suited to the SU-SDR task, since the transcriptions contain no case, sentence or paragraph information. "Pseudo" paragraphs of 10s of speech were made for each show and the default text-tiling parameters [8] were used along with some additional abbreviations processing, to obtain the "tile" boundaries. Our standard retriever, similar to our TREC-7 system [11], was then used to produce the final ranked list. The windowing system made pseudo-stories of a given length and skip before running the retriever as before. The results are given in Table 9. The windowing system seemed to offer greatest potential and hence the basis of the SU system was chosen to be a sliding window of length 30 seconds and skip 15 seconds.

System	dup-irrel	dup-del	#"Stories"
TREC-7 Story-known	50.3	50.3	2866
Text-tiling	23.2	25.3	4195
Windowing - 120s@60s	28.2	34.0	5226
Windowing - 120s@30s	24.7	35.5	10181
Windowing - 30s@15s	$\boxed{33.9}$	$\boxed{46.1}$	18669
Windowing - 30s@10s	27.7	44.0	27890

Table 9: AveP for simple SU systems on the TREC-7 data

The standard retrieval engine was then replaced by a more complicated system, similar to the one described in [14], and *forced-breaks* were added during the windowing to prevent windows being formed over gaps of more than 5 seconds in the audio. Any very short windows (<8 seconds or ≤16 words) were removed at this stage. The results are given in Table 10. The increase in performance due to a more sophisticated retrieval engine, which includes SPI and relevance feedback, is clearly shown. Forcing breaks at gaps in the audio did not have much effect on the TREC-7 data (which contained no commercials), but it was hoped that these gaps (generally formed by music/silence removal in the segmentation, or commercial elimination for the TREC-8 system) would offer a good indication of story boundary for the TREC-8 data, and hence should be enforced as hard breaks.

System	dup-irrel	dup-del
Baseline from Table 9	33.9	46.1
Improved Retriever	36.5	51.2
Improved Retriever + forced-breaks	36.0	51.6

Table 10: SU AveP improvements on the TREC-7 data

Post-processing the retrieval output in order to prevent multiple hits of the same story was then examined. Smoothing was added such that for any given query, any stories which were returned as relevant and originated from within a certain time, T_{merge}, in the same broadcast were pooled, with only the highest scoring window being retained. The others were placed in order at the bottom of the ranked list. The results for different values of T_{merge} are given in Table 11.

The results show that the best performance using the TREC-8 evaluation measure (dup-irrel) for the TREC-7 data is obtained with a smoothing time of 105s. This is surprisingly high, but it was thought that the probability that two temporally close windows both being retrieved for a given query but not being

T_{merge}	0	15	30	45	60
dup-irrel	36.0	45.0	45.6	46.2	46.9
dup-del	51.6	51.6	51.1	50.7	50.6
T_{merge}	75	90	105	120	
dup-irrel	47.5	47.8	$\boxed{48.1}$	48.0	
dup-del	50.6	50.3	50.3	50.0	

Table 11: AveP for different merge times for post-processing on the TREC-7 data

from the same story was quite low. Since the TREC-8 collection contained more data and had a greater proportion of CNN broadcasts, which generally produce shorter stories, the parameter T_{merge} was set to the sub-optimal, but shorter 75s for the TREC-8 evaluation.

Attempts were made to modify the score from the retriever of any window which represented a merged group of windows, before re-ordering during the post-processing phase, but this proved not to be beneficial for the TREC-7 data. Finally hard breaks, as defined by a certain length gap in the audio, were also enforced in the post-processing phase, so that no merging could take place over such a break. The results are given in Table 12 for a T_{merge} of 75 seconds and 120s.

Audio Gap	T_{merge}=75s		T_{merge}=120s	
for Boundary	dup-irrel	dup-del	dup-irrel	dup-del
100s or ∞	47.51	50.62	48.03	50.03
15s	47.46	50.61	48.05	50.11
10s	47.49	50.63	48.08	50.13
5s	47.46	50.64	$\boxed{48.34}$	50.45

Table 12: Effect of enforcing hard boundaries in post-processing on TREC-7 data

No real benefit is shown for the TREC-7 data when the smoothing is relatively conservative, but for the case of T_{merge}=120s, when the smoothing time is greater than optimal value, the enforcement of boundaries for audio gaps of 5s does increase performance slightly. Since the problem of over-smoothing was thought to be greater for TREC-8 as the commercials had not been manually removed, the enforcement of boundaries at 5s gaps in the audio was maintained.

The final system, summarised in Figure 2, gave an AveP of 41.47 (R-prec=41.98) on our own transcriptions on the TREC-8 task. A more detailed analysis of the SU results for TREC-8 can be found in [13].

6. CROSS-RECOGNISER EXPERIMENTS

Several sets of transcriptions from other participating sites were offered to allow comparisons to be made between retrieval using different recognition systems. The detailed breakdown of the word error rate of these transcriptions is given in Table 7 in section 3.1. The AveP for both the SK and SU runs, along with

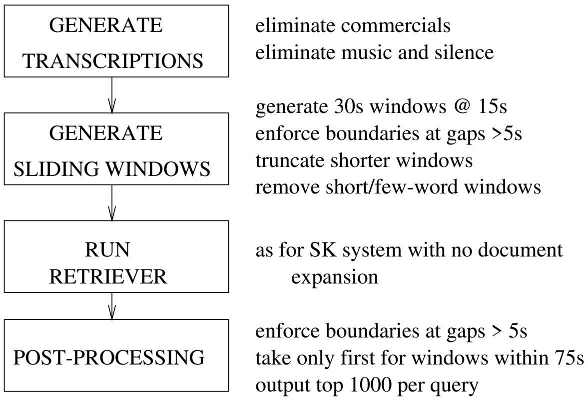

Figure 2: The TREC-8 SU system

the term error rate [10] after stopping and stemming (SSTER) and word error rate (WER) is given in Table 13. The AveP for a benchmark system with no relevance feedback, document expansion or parallel collection frequency weights (BASE) is given as a comparison. [3]

The term error rate after document expansion (DETER) is also given in Table 13 as a comparison. To calculate this measure, pre-processing, poset mapping and then document expansion are performed on both the reference and hypothesis transcriptions before the standard term error rate is calculated.[4]

	Error Rate on 10hr subset			Average Precision		
Recogniser	WER	SSTER	DETER	SK	BASE	SU
cc-proc	8.8	14.2	30.82	54.93	48.54	——
cc-unproc	12.4	18.0	83.07	**52.32**	48.93	——
CUHTK-s1	20.5	27.8	45.89	55.29	46.04	41.47
LIMSI	21.5	29.1	47.25	54.12	45.19	40.19
CUHTK-p1	26.6	36.5	56.13	54.51	44.84	41.50
NIST-B2	26.7	35.0	51.56	53.02	43.64	38.70
NIST-B1	27.5	36.1	81.12	**49.63**	43.25	38.62
AT&T	29.3	38.6	55.73	52.75	43.89	——
Sheffield	32.0	44.7	60.66	52.85	42.47	38.24
CMU	64.4	77.8	103.52	39.36	31.37	——

Table 13: AveP for SK and SU cross-recogniser evaluation conditions with corresponding transcription error rates

Figure 3 shows the relationship between stopped-stemmed term error rates (SSTER) and AveP. Whilst the benchmark (BASE) performance can be predicted reasonably well from SSTER, there is more, seemingly unpredictable, variation for the case of the complete SK system. In particular, the AveP for the `NIST-B1`

[3]The unprocessed version of the closed caption transcriptions `cc-unproc` is not included in all the subsequent analysis since it does not reflect the standard output format

[4]Since there is no guarantee that the terms added to the reference transcriptions during document expansion will be "good" terms for the subsequent retrieval runs, the new "reference" transcriptions may no longer represent the ideal case, but it was hoped that this measure would allow the effects of document expansion to be seen and in particular to show up any major problems which occurred during the document expansion process.

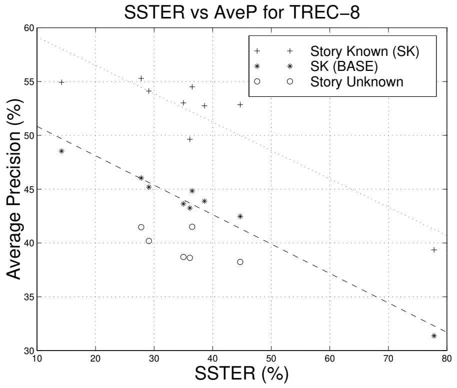

Figure 3: Relationship between AveP and SSTER

and `cc-unproc` runs is much worse than that predicted by the SSTER. However, the DETER for both these cases is unusually high, suggesting the problem for these runs lay in the document expansion process.[5]

It is interesting to note that the best-fit lines for both the complete SK system and the benchmark SK cases are almost parallel, (gradients -0.26 and -0.27 respectively), showing that the inclusion of relevance feedback for query and document expansion and parallel collection frequency weights improves the overall AveP by around 8.5% absolute across the complete range of transcription error rates.

The SU results follow a roughly similar pattern, suggesting that generally transcriptions which work well for the SK case also work well for the SU case. It is pleasing to note that the output from the first pass of our system, CUHTK-p1, does better than might be predicted from its error rate. This is due in part to the reduction in false alarms because of the elimination of commercials in the system. This is confirmed by the results given in Table 14, which show that the AveP on CUHTK-p1 transcriptions would have fallen by 0.5% if the commercial detector had not been used, whereas the performance on LIMSIs transcriptions increases by over 0.5% when the detected commercials are filtered out during the post-processing stage (see [13] for more details).

Run	No Commercials removed	`COMM-EVAL` removed
CUHTK-p1	41.00%	41.50%
LIMSI	40.19%	40.75%

Table 14: Effect on AveP for the SU case when automatic commercial detection is included

6.1. New TERs to Predict Performance

Term Error Rates were introduced in [11] to model the input to the retriever more accurately than the traditional word error rate.

[5] It was found that a disk filling up during the document expansion process for NIST-B1 was responsible for the relatively poor performance for this case. When rectified, the AveP for NIST-B1 was 52.81

If knowledge about the retrieval process itself is known in advance, then the TER can be modified to exploit this information to model the retrieval process more closely and therefore hopefully provide a better predictor of the performance of the final system. An example of this is using SSTER, where the stopping, mapping and stemming processes used in the first stage of indexing the transcriptions, is incorporated into the error rate calculation.

If more information is known about how the scores are generated within the retriever for a given term, then new TERs can be defined which incorporate this information. The generic TER function thus becomes:

$$TER = \frac{\sum_w [f_w(|R(w) - H(w)|)]}{\sum_w [f_w(R(w))]}$$

where f_w is some function which generally depends on the word w, R is the reference and H the hypothesis. This can be seen to reduce to the standard TER when f is the identity function. Some other possibilities for the function f_w which allow the collection frequency weighting (inverse document frequency),[6] or the combined weights formula to be included directly are:

$$\begin{aligned} f_w(x) &= x/n \\ f_w(x) &= x\,[log(N) - log(n_w)] \\ f_w(x) &= \frac{x\,[log(N) - log(n_w)]\,(K+1)}{x + K[1 - b + b\,\mathrm{ndl}]} \end{aligned} \quad (1)$$

where N, K, b, n and ndl have the same meaning as in section 4.1.3. It is also possible to include the frequency of each term in the *query* as a scale factor within f_w if the queries are known, but this makes the score query-dependent, which may be undesirable, and care must be taken in defining the query terms if relevance feedback is used for query expansion.

The TERs using (1), including stopping, stemming, mapping, posets, document expansion and parallel collection frequency weights within the combined weighting formula are given in Table 15. Unfortunately these numbers do not appear to offer a better predictor for our AveP results. This may be because the words added to the "reference" during document expansion may not be the best in terms of retrieval performance, or that only the query terms themselves should be taken into account, or simply the overall performance on the entire 500 hour collection cannot be predicted well using the scored 10 hour subset.

Rec.	HTK	cc-proc	HTK-p1	LIMSI	NIST-B2
Error	55.51	37.93	67.86	57.20	62.46
Rec.	Sheff	AT&T	cc-unproc	NIST-B1	CMU
Error	72.80	66.79	104.75	91.90	121.68

Table 15: Term error rate modelling stopping, stemming, mapping, posets, document expansion and PCFW with combined weighting on the scored 10 hour subset

[6] Another method of modifying the TER to model retrieval weighting more closely can be found in [20]

7. CONCLUSIONS

This paper has described the systems developed at Cambridge University for the 1999 TREC-8 SDR story known and story unknown evaluations.

A new method of automatically detecting commercials has been shown to work well, with 97.8% of the 42.3 hours of data automatically labelled as commercials being marked as non-story information by humans. By automatically eliminating these "commercials" at an early stage, the computational effort required during speech recognition was reduced by 8.4% and the Average Precision for the story unknown task was increased by 1.2% relative.

Two HTK-based transcription systems were made. The first ran in 3 times real time and gave a word error rate (WER) of 26.6% on the scored 10 hour subset of the data. The second ran at 13 times real time and included a second pass with a 108k vocabulary and speaker adaptation, giving a WER of 20.5%, the lowest in the track by a statistically significant margin.

Several extensions to our retriever have been described and shown to increase Average Precision on our best transcriptions for the story-known case by 23% relative, giving a final value of 55.29%. These included semantic poset indexing, blind relevance feedback, parallel blind relevance feedback for both query and document expansion and parallel collection frequency weighting.

The system developed for the case where story boundaries were not known included automatic detection and elimination of commercials, windowing using the segmentation information, retrieval using all the strategies developed for the story-known case except document expansion, and post-filtering to recombine multiple hits from the same story. The final system gave an average precision of 41.5% on both sets of our transcriptions.

Finally, experiments were described using other transcriptions and the relationship between transcription error rate and performance was investigated. The results from TREC-7 showing that the degradation of performance with increasing error rate was fairly gentle were confirmed on this significantly larger data set.

Acknowledgements

This work is in part funded by an EPSRC grant reference GR/L49611. Thanks to Tony Robinson for the initial idea that repetitions of audio could help to indicate the presence of commercials.

8. REFERENCES

[1] F. Bimbot & L. Mathan *Text-Free Speaker Recognition using an Arithmetic Harmonic Sphericity Measure*. Proc. Eurospeech '93, Vol. 1. pp. 169-172, 1993

[2] C. Fellbaum *WordNet: An Electronic Lexical Database* MIT Press, ISBN 0-262-06197-X, 1998

[3] M.F.G. Gales & P.C. Woodland *Mean and Variance Adaptation Within the MLLR Framework* Computer Speech & Language, Vol. 10 pp. 249-264, 1996

[4] J.S. Garofolo, E.M. Voorhees, C.G.P. Auzanne, V.M. Stanford & B.A. Lund *The 1998 TREC-7 Spoken Document Retrieval Track Overview and Results* Proc. TREC-7, pp. 79-89, 1999

[5] J.S. Garofolo, E.M. Voorhees, C.G.P. Auzanne & V.M. Stanford *Spoken Document Retrieval: 1998 Evaluation and Investigation of New Metrics* Proc. ESCA Workshop on Extracting Information from Spoken Audio, pp. 1-7, 1999

[6] J.S. Garofolo, C.G.P. Auzanne, E.M. Voorhees, K. Spärck Jones 1999 TREC-8 Spoken Document Retrieval (SDR) Track Evaluation Specification
http://www.nist.gov/speech/sdr99/doc/sdr99_spec.htm

[7] T. Hain, S.E. Johnson, A. Tuerk, P.C. Woodland & S.J. Young *Segment Generation and Clustering in the HTK Broadcast News Transcription System* Proc. DARPA Broadcast News Transcription and Understanding Workshop, pp. 133-137, 1998

[8] M.A. Hearst *TextTiling: Segmenting text into multi-paragraph subtopic passages* Computational Linguistics, Vol. 23. pp. 33-64, 1997 (Source code *http://elib.cs.berkeley.edu/src/texttiles/*)

[9] S.E. Johnson & P.C. Woodland *Speaker Clustering Using Direct Maximisation of the MLLR-Adapted Likelihood* Proc. ICSLP 98, Vol. 5 pp. 1775-1778, 1998

[10] S.E. Johnson , P. Jourlin, G.L.Moore, K. Spärck Jones & P.C. Woodland *The Cambridge University Spoken Document Retrieval System* Proc ICASSP'99, Vol. 1, pp 49-52, 1999

[11] S.E. Johnson , P. Jourlin, G.L.Moore, K. Spärck Jones & P.C. Woodland *Spoken Document Retrieval for TREC-7 at Cambridge University* Proc. TREC-7, pp. 191-200, 1999

[12] S.E. Johnson, P.C. Woodland *A Method for Direct Audio Search with Applications to Indexing and Retrieval* To appear in ICASSP 2000

[13] S.E. Johnson , P. Jourlin, G.L.Moore, K. Spärck Jones & P.C. Woodland *Audio Indexing and Retrieval of Complete Broadcast News Shows* To appear in RIAO 2000

[14] P. Jourlin, S.E. Johnson , K. Spärck Jones & P.C. Woodland *General Query Expansion Techniques for Spoken Document Retrieval* Proc. ESCA Workshop on Extracting Information from Spoken Audio, pp 8-13, 1999

[15] P. Jourlin, S.E. Johnson , K. Spärck Jones & P.C. Woodland *Improving Retrieval on Imperfect Speech Transcriptions* Proc. SIGIR '99 pp. 283-284, 1999

[16] C.J. Leggetter & P.C. Woodland *Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models* Computer Speech & Language, Vol. 9 pp. 171-185, 1995

[17] P. Nowell *Experiments in Spoken Document Retrieval at DERA-SRU* Proc. TREC-7, pp. 353-362, 1999

[18] J.J. Odell, P.C. Woodland & T. Hain *The CUHTK-Entropic 10xRT Broadcast News Transcription System* Proc. 1999 DARPA Broadcast News Workshop. pp 271-275, 1999

[19] M.F. Porter *An Algorithm for Suffix Stripping* Program, 14 pp. 130-137, 1980

[20] A. Singhal & F. Pereira *Document Expansion for Speech Retrieval* Proc. SIGIR '99, pp. 34-41, 1999