ANU/ACSys TREC-5 Experiments

David Hawking, Paul Thistlewaite and Peter Bailey
Co-operative Research Centre For Advanced Computational Systems
Department Of Computer Science
Australian National University
{dave,pbt,peterb}@cs.anu.edu.au

February 1, 1997

Abstract

A number of experiments conducted within the framework of the TREC-5 conference and using the Parallel Document Retrieval Engine (PADRE) are reported. Several of the experiments involve the use of distance-based relevance scoring (spans). This scoring method is shown to be capable of very good precision-recall performance, provided that good queries can be generated. Semi-automatic methods for refining manually-generated span queries are described and evaluated in the context of the adhoc retrieval task. Span queries are also applied to processing a larger (4.5 gigabyte) collection, to retrieval over OCR-corrupted data and to a database merging task. Lightweight probe queries are shown to be an effective method for identifying promising information servers in the context of the latter task. New techniques for automatically generating more conventional weighted-term queries from short topic descriptions have also been devised and are evaluated.

1 Introduction

The work reported here comprises a number of text retrieval experiments conducted within the framework of the TREC-5 task and addressing questions of interest in the following research areas: Applications of parallelism in information retrieval; Distance-based relevance scoring; Distributed IR; and Automatic query generation.

In TREC-5, ANU/ACSys runs were submitted in Automatic Adhoc, Manual Adhoc, DB Merging, Confusion and VLC categories.

1.1 Hardware and Software Employed

A 128-node, distributed-memory Fujitsu AP1000 (16 MBytes of RAM per node, 2 gbytes in total) running PADRE [6] software was used for all retrieval runs reported here. Most runs used the Super Dictionary (SD) method (using disk-resident inverted files) [1] rather than the Full Text Scanning (FTS) method used in previous ANU TREC submissions.

1.2 Statistical Testing of Differences Between Runs

Throughout this paper, many comparisons are made between pairs of runs. In making these comparisons, apparent differences between means have been tested for statistical significance using the well-known t-test. Two-tailed tests with a 5% confidence level were used.

Table 1: Comparison of ANU TREC-5 Automatic Adhoc runs with that of TREC-4.

	`padreA`(TREC-4)	`anu5aut1`	`anu5aut2`
Recall	35%	34%	36%
Average Precision	.1453	.1537	.1538
Exact Precision	.1954	.1948	.1980
Precision at 10 docs	.2949	.2540	.2280
Precision at 20 docs	.2857	.2160	.2100
Precision at Recall .10	.3186	.2869	.2898
Precision at Recall .20	.2521	.2399	.2378
Precision at Recall .40	.1722	.1760	.1778
Precision at Recall .50	.1300	.1557	.1575
No. "best" on ave prec	-	6	3
No. "best" on recall	-	8	6

2 Automatic Query Generation

Automatic AdHoc, Official Runs `anu5aut1` and `anu5aut2`

The main activities in ANU/ACSys participation in the Automatic AdHoc category were directed toward:

1. Replacing the semantics-based techniques used in the TREC-4 runs for determining phrases and for selecting and weighting query terms, with techniques based on term frequency alone.

2. Attempting to determine from a topic whether to restrict documents to being USA-only, foreign-only, etc.

3. Developing and evaluating a new stemming algorithm.

Only the short form of the topic descriptions was used.

2.1 Relevance Scoring Method in Automatically Generated Queries

The relevance ranking algorithm was the same as that used by ANU in the TREC-4 Automatic Adhoc task, but with a different term weighting scheme. The terms used for a given topic were determined by extracting all non-function content words and then searching for all co-occurrences of these terms (within varying lexical proximities) to determine suitable multi-word terms. The weights for these phrasal terms were computed as a function of their occurrence frequencies and their expected frequencies.

In the `anu5aut2` run, all multi-word terms that were derived were used; in the `anu5aut1` run there was a threshold value used to select a subset of the multi-word terms.

All non-function single words from the topic also received a weight proportional to the *idf* for the word.

Results: Table 1 compares the performances of TREC-4 and TREC-5 automatically generated queries averaged across all topics. There was no significant difference between the two TREC-5 runs on average precision ($t(49) = 0.059$) but `anu5aut2` performed 9% better on average topic-by-topic percentage recall ($t(49) = 2.589$, $p < 0.05$).

Discussion and Conclusions: Comparison of the two TREC-5 runs suggests that attempting to select a subset of the multi-word terms may be counter-productive. The similarity of TREC-5 to TREC-4 results was a source of initial disappointment until it was realised that other groups found the task significantly harder than its equivalent in TREC-4. This observation is corroborated by the big improvement in the number of "best" scores achieved (see table 1). Although early precision worsened relative to TREC-4, average precision improved slightly due to improved precision in middle to later stages (0.4 recall level onwards). The fact that average precision results were similar to those of leading groups prior to query expansion indicates that the current method will provide a promising basis for application of expansion techniques.

3 Effectiveness of Distance-Based Relevance Scoring

The query set employed in ANU's TREC-4 Manual Adhoc submission in TREC-4, (which will be referred to here as Q_{T4}^{ANU}) scored document relevance using only the lexical distance between instances of concepts in concept intersections (*Z-mode*). Clarke, Cormack and Burkowski [3] independently developed a very similar system of distance-based scoring and used it in the University of Waterloo TREC-4 submission. Buckley, Singal and Mitra [2] also used distance-based measures in their Individual Term Locality run `CrnlAL`.

Subsequent to TREC-4, Hawking and Thistlewaite [9] proposed an extended formal model of *spans*. Much of this model has now been implemented in PADRE and was used as the scoring method in the Manual Adhoc, DB Merging, Confusion and VLC runs reported below.

Although distance-based queries performed well in the Manual Adhoc and DB Merging categories of TREC-4, there remained a gap between their performance and that of the best conventionally-ranked queries. The goal of the present experiment was to determine whether the gap represented a fundamental limitation of distance-based scoring or whether the problem lay in the quality of the queries. A partial answer was sought by attempting to create a set of high-standard distance-based queries and to compare its performance with that of the best TREC-4 systems. The new query set was constructed by selecting the best-performing of three independently generated queries for each topic. Each query was constructed without reference to retrieved documents.

The three query sets used comprised Q_{T4}^{ANU}, the official Waterloo TREC-4 queries (Q_{T4}^{UW}), and a new set of span-based TREC-4 queries (Q_{T4}^{T5prac}) generated to practice a new manual approach to query generation. The author of the latter queries was the same person who generated the Q_{T4}^{ANU} set but it was hoped that a period of two months without exposure to the topics or documents would permit the second set to be relatively independent of the first. These query sets achieved average precision results of 0.2383 (Q_{T4}^{ANU}), 0.2994 (Q_{T4}^{UW}) and 0.2898 (Q_{T4}^{T5prac}), compared with 0.3436 for the overall best official TREC-4 run (`CnQst2`, submitted by Excalibur Technologies Inc).

The sixteen queries from the Q_{T4}^{UW} set which performed more than 0.100 better on average precision than the corresponding ones from Q_{T4}^{ANU} were translated into PADRE format. It was verified that each translated query achieved similar average precision to the Waterloo origi-

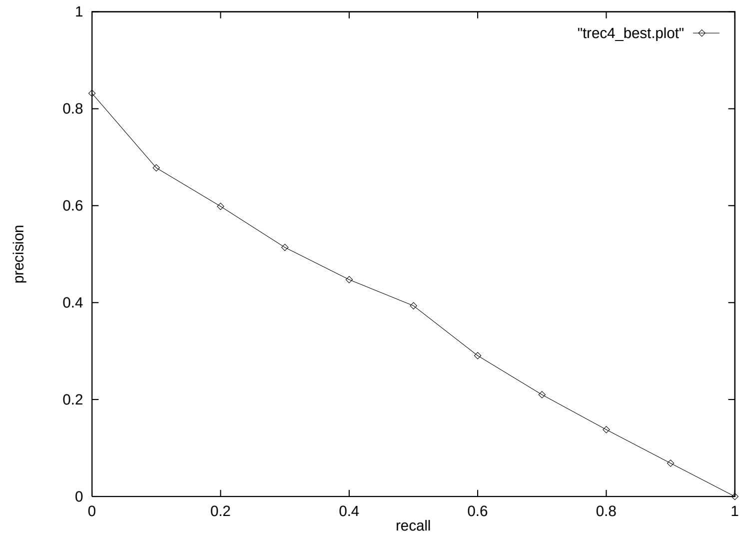

Figure 1: Precision-recall curve for the best available PADRE queries for the TREC-4 task. They were scored entirely according to span distances. The distance decay function employed was the custom function shown in figure 3. The maximum score achievable by a partial span was reduced by a factor of 10 for every term missing. A proximity limit of 1000 characters applied to spans.

nal. The translated queries then replaced the corresponding inferior versions in Q_{T4}^{ANU} to form $Q_{T4}^{ANU/UW}$. This query set achieved an average precision of 0.3208.

Finally, by converting the $Q_{T4}^{ANU/UW}$ queries to PADRE's new span formulation (which scores partial spans appropriately) and merging the best of these queries with the best of Q_{T4}^{T5prac}, the set Q_{T4}^{best} was formed.

Results: The Q_{T4}^{best} queries retrieved 69.7% of all relevant documents, averaged 6.3 relevant documents in the first 10 and achieved an average precision of **0.3634**. As shown in the topic-by-topic results for this query set (appendix A), average precision results are better than the median of all runs for 44 of the 49 topics.

Discussion and Conclusions: These results compare very favourably with corresponding figures of 71.0%, 5.7 and 0.3436 for the best official TREC-4 run (`CnQst2`). As it is not yet certain whether distance-based queries are suitable for all topics, it would be useful to further study the queries for the five below-median topics with a view to determining whether any of them appear to be intractable to span scoring. Although the process of selecting the best of different sets of queries violates TREC-4 rules, it is justifiable to conclude that use of distance-based scoring alone does not limit performance to a level below that of state-of-the-art, conventionally scored systems. However, methods for reliable, low cost generation of high-standard distance-based queries are clearly needed.

4 Manual (Non-Interactive) Query Generation

Manual AdHoc, Official Runs `anu5man4` and `anu5man6`

The following goals motivated the work in the Manual AdHoc category:

```
topic 251
weight 0
anyof "exportation |exporting |offshore "
anyof "factories |factory |industries |industry |manufactur"
loadmatchset USall
loadmatchset countries
anyof "employ|job |jobs |unemploy"
span key 5 1000 2000 2 2
top 1000
```

Figure 2: The Q_0 query for the topic relating to the exportation of industry. `USall` and `countries` are pre-computed matchsets resulting from searches for very large numbers of words, phrases and abbreviations indicating the USA and other countries respectively. The span command scores the relevance of documents containing spans across all 5 matchsets, of which the first two are mandatory. Documents containing instances of all five matchsets within a proximity of 2,000 characters receive a relevance score contribution depending upon the length of the span. Partial spans, for example those missing an indication of a foreign country, will attract a lower score contribution than would a complete span of the same length.

1. To confirm or otherwise the precision-recall effectiveness of distance-based measures alone,
2. To reduce the amount of human time spent in writing queries,
3. To reduce reliance on subject experts during query construction, and
4. To explore automatic query construction aids for span queries.

4.1 Query Structure

Manual Adhoc queries were constructed using concept intersections. Queries consisted of 1 - 3 spans. Spans contained 1 - 5 branches. The enhanced span model allows scoring of partial spans and also allows singleton spans, which differ from conventional terms only in that *idf* plays no part whatever. In certain topics, such as `self induced hypnosis` and `treatment of schizophrenia` from TREC-4, the use of singleton spans is critical to good ranking.

4.2 Overview of Manual Query Generation Process

In order to study the effectiveness of various query enhancement techniques, an initial set of queries was subjected to successive refinements and each resulting set was run against the TREC-5 data. The initial set of queries was drafted without using any information from the collection. Refinements used term-term implication and partial-match frequency information derived from the collection. Finally, the refined queries for some topics were augmented with manually generated regular expressions for numbers, dates and currency amounts.

A utility program `qpreen` was developed and used to check queries for errors, inconsistencies and inefficiencies. No retrieved documents were examined manually at any stage in manual query construction.

4.3 Construction of Initial (Draft) Queries

Queries were again generated by the first author. Last year, considerable time was invested in constructing the queries and considerable help was obtained from subject experts. In some cases this resulted in unnecessarily elaborate queries constructed around terms which are important in the subject as a whole but which appear too infrequently in the collection to be useful.

This year, in line with the goals stated above, no experts were consulted and only about half the amount of time was allowed for the manual input. (See table 2.)

The set of initial queries thus constructed was called Q_0. It was used in [unofficial] run `anu5man1`. An example query is shown in figure 2.

> **Results:** The results achieved by `anu5man1` were considerably worse than the corresponding `padreZ` run from TREC-4. Average precision dropped from 0.2383 to 0.1973. Early precision and overall recall also declined markedly.
>
> **Discussion and Conclusions:** The decline in performance relative to the same category in TREC-4 is partly due to the increased task difficulty. No doubt there is also a relative decline due to reduced query input time and lack of subject-expert advice. In this context it is difficult to estimate whether any benefit was conferred by the increased sophistication of span scoring.

After construction of the initial query for each topic a number of partial queries were extracted, to be used in term implication runs explained below. The number of partial queries used for a topic ranged from one to five.

4.4 Query Augmentation

Robertson [10] argues that the best methods for selecting terms for query expansion and for weighting selected terms are not necessarily the same. In distance-based queries, as formulated here, this difficulty is avoided because individual terms are not weighted. However, a new complication is introduced.

When augmenting distance-based queries using concept intersections, it is necessary not only to find new terms but also to identify which concept they augment. New terms found to be strongly associated with documents scoring highly against a query may be good candidates for addition. However, if a particular new term is added to the definition of the wrong concept, then certain documents will score artificially highly. For example, if the term `BFI` (a well-known recycling company) were added to the `tire` concept rather than the `recycling` concept in a 3-way intersection addressing `the economic impact of recycling tires` then some totally `tire`-free documents might achieve undeservedly high scores.

In term association, terms are sought which tend to co-occur with a query term (which might be a stem, a word, a phrase or a complex sub-query). All documents matching the query term and all those containing candidates for association are considered. The strength of association between a candidate term t_c and the query term t_q can be expressed as:

$$A_{cq} = \frac{|D_c \cap D_q|}{|D_c \cup D_q|}$$

where D_c and D_q are the sets of documents containing t_c and t_q respectively.

In the experiments reported here, implication strengths rather than association strengths were used. These are directional.

$$I_{cq} = \frac{|D_c \cap D_q|}{|D_c|}$$

$$I_{qc} = \frac{|D_c \cap D_q|}{|D_q|}$$

I_{cq} measures the extent to which the presence of t_c in a document implies that t_q will also be present. $I_{cq} = 1.0$ iff every document containing t_c also contains t_q, but there may be documents containing t_q which do not contain t_c.

The implication strength machinery was also used to achieve the effect of relevance feedback. The same formulae for implication strength were used, except that D_q became the set of top-ranked documents relative to a query q rather than the complete set of matching documents.

In searching for useful additional terms with which to augment the manual queries two distinct processes were used. These are best described by illustration.

Imagine a draft query consisting of a three-way intersection of concepts such as `economic impact`, `recycling` and `tires`. The relevance feedback process finds the top n (say 10) documents against the partial query and looks for terms which occur in a high proportion of them. The second process uses term association to try to find terms which might be associated with individual concepts, with individual concepts or with sub-groupings of concepts. In the term association process, implication strengths are computed for all collection terms passing low-cut and high-cut frequency filters against a series of partial queries such as: `profit*`, `recycl*`, `tires`, `recycl*` *near* `tires`, `profit*` *near* `recycl*`.

4.5 First Refinement of Draft Queries. $Q_0 \rightarrow Q_{RF1}$

The set of queries augmented using the results of term implication results over the partial queries ONLY is called Q_{RF1}. It was used in [unofficial] run `anu5man2`.

> **Results:** Run `anu5man2` performed significantly better than `anu5man1` on topic-by-topic percentage recall ($t(49) = 2.448$, $p < 0.05$, observed difference +11%). There was no significant difference in average precision ($t(49) = 1.585$, observed difference +10%) or precision @20 ($t(49) = 2.172$, observed difference +7%).

4.6 Second Refinement of Draft Queries. $Q_{RF1} \rightarrow Q_{RF2}$

Full queries from Q_{RF1} were used in a further term implication run resulting in a set of queries called Q_{RF2} which were used in runs `anu5man3` and `anu5man4`.

Only the following information was used in refining Q_{RF1}:

- Terms implying or implied by the top 10 ranked documents selected by the partial or draft query,
- Terms implying or implied by all documents selected by the partial or draft query,
- The count of documents selected by partial or draft queries.

Table 2: Human time consumed in manual query construction. The regular expressions incorporated in Q_{RF2RX} tended to be repeated in multiple topics and the time quoted is total time spent divided by the number of queries (12) which were augmented with regular expressions. The time to devise (and test) complex one-off regular expressions would be greater than the quoted amount. The quoted amount would have been much less if all 50 queries had required recognition of numbers, dates etc.

Activity	Time per query (min.)
Initial composition	15
Checking	5
$Q_0 \rightarrow Q_{RF1}$	10
$Q_{RF1} \rightarrow Q_{RF2}$	5
$Q_{RF2} \rightarrow Q_{RF2RX}$	3

Results: Apparent differences between means for runs `anu5man4` and `anu5man2` were small (4% in average precision, -2% in precision @20 and 1% in recall) and not statistically significant ($t(49) = 1.334, 0.649, 0.428$ respectively).

Compared to the baseline `anu5man1`, `anu5man4` was significantly better on topic-by-topic percentage recall ($t(49) = 2.264$, $p < 0.05$, observed difference 12%). Apparent improvements of 5% in precision @20 and 15% were not statistically significant. ($t(49) = 1.117$ and 1.973 respectively.)

Compared to manual runs submitted by other groups, `anu5man4` achieved best or equal best results on seven topics in recall and on three in average precision. However, the total number of relevant documents for many of these topics was quite small (in two cases, as few as one). Run `anu5man4` performed better than or equal to the median on 29 topics for recall and 30 topics for average precision.

Discussion and Conclusions: On the basis of training results using TREC-4 data and judgments, results in the 0.28 - 0.32 average precision range were expected. Actual performance fell far short of this. It may be that expectations based on training with TREC-4 data were unrealistically high due to an overtraining effect. However, it is clear that the TREC-5 task was harder than that of TREC-4.

Overall, run `anu5man4` is understood to have achieved very close to the best results for non-interactive manual submissions. Nearly all participants who achieved better results used queries modified after examination of retrieved documents or contexts. Compared to all manual runs submitted by other groups, `anu5man4` achieved best or equal best results on seven topics in recall and on three in average precision. However, the number of *best* recall results may not be very meaningful as the total number of relevant documents for many of these topics was quite small (in two cases, as few as one). Run `anu5man4` performed better than or equal to the overall median on 29 topics for recall and 30 topics for average precision.

4.7 Numbers, Dates, Percentages and Currency Sums. $Q_{RF2} \rightarrow Q_{RF2RX}$

Twelve topics (numbers 257, 260, 264, 268, 272, 277, 285, 286, 291, 293, 298, and 300) were selected as those most likely to benefit from the recognition of numbers, dates, percentages, and currency amounts. Three of these (264, 293, and 298) requested the imposition of a date filter such as “after 1900”.

Table 3: Separation values (equivalent number of words) accorded to intervening document features. A sample of documents was read to make a list of SGML tags which indicate no greater semantic or structural barrier than a sentence break. Twelve such marker pairs were identified. They are the so-called *lightweight* tags. All other SGML tags were assumed to be *heavyweight*.

Feature	Distance (equiv. no. words)
Sentence break	4
Uncertain paragraph break	6
Certain paragraph break	8
Lightweight SGML tag	4
Heavyweight SGML tag	200

It was decided to modify the Q_{RF2} queries for these topics by including regular expressions to achieve the desired effect. This produced a new query set Q_{RF2RX}. The 12 modified queries were then run using PADRE's FTS (Full Text Scanning) method as regular expressions are not supported in SD method.

It was realised before running that the presence of numeric quantities in SGML fields (such as DOCIDs, DATEs, DOCNOs etc) accompanying nearly every document would cause a large number of spurious matches on dates and numbers (but probably not percentages or currency amounts). Two alternative solutions to this problem were identified:

1. Use DTDs for the documents to prevent the regexp search code from looking in these confusing places, and

2. Adapt the span scoring code to significantly increase the span length when an SGML tag was encountered.

Of these, the second method was chosen because it was consistent with a desire to investigate measurement of span length taking into account intervening text features such as sentence breaks, paragraph breaks and SGML tags of various types. (See Hawking and Thistlewaite, 1996[9], p. 13.)

The Q_{RF2} queries were run again using a non-linear scoring algorithm whose salient properties are summarised in table 3 to serve as a baseline for judging the benefit of using regular expressions. Unfortunately, the non-linear span-length algorithm has only been implemented in FTS method and consequently there was a very considerable increase in runtimes. Indeed, processing the regexp version of query 277 took more time than all 50 queries in `anu5man1`!

> **Results:** The introduction of non-linear scoring (coupled with the change from super dictionary method to full text scanning) caused an unexpected decline in performance. Runs `anu5man4` and `anu5man5` used identical queries but relative to the former, `anu5man5` was significantly worse on average precision ($t(49) = 3.131, p < 0.05$, observed difference -18%) and on topic-by-topic percentage recall ($t(49) = 4.401, p < 0.05$, observed difference -16%). An apparent difference of -7% on precision @20 was not statistically significant.
>
> Official run `anu5man6` was evaluated using `anu5man5` as a baseline in order to judge the effectiveness of the regular expressions. Considering only the twelve affected topics, the use of regular expressions made a statistically significant difference to

Table 4: Summary of Manual AdHoc runs.

Run-id	Queries	Method	Scoring	Prox. Lim.	Runtime	Recall	@20	Ave. Prec.
anu5man1	Q_0	SD	1/sqrt	2000	23:35	0.4642	0.3650	0.1973
anu5man2	Q_{RF1}	SD	1/sqrt	2000	28:18	0.5163	0.3910	0.2170
anu5man3	Q_{RF2}	SD	1/sqrt	2000	30:00	0.5203	0.3830	0.2161
anu5man4	Q_{RF2}	SD	custom	1000	32:22	0.5230	0.3840	0.2261
anu5man5	Q_{RF2}	FTS	custom/nls	1000	2:23:34	0.4213	0.3560	0.1849
anu5man6	Q_{RF2RX}	FTS	custom/nls	1000	4:02:59	0.4354	0.3580	0.1889

topic-by-topic percentage recall ($t(11) = 2.554, p < 0.05$, observed difference +13%). Apparent improvements of 2% in precision @20 and 11% in average precision were not significant ($t(11) = 0.261$ and 1.443 respectively).

On a topic-by-topic basis, recall apparently improved on nine topics and deteriorated on only one. Average precision improved on seven and deteriorated on five. Precision @20 improved on six and deteriorated on four.

Discussion and Conclusions: The cause of the decline in performance from run `anu5man4` to run `anu5man5` is of some concern. Further work is needed to ascertain whether the problem lies with the assignment of separation values, or to an unexpectedly significant difference between SD and FTS methods or to the presence of a bug. Despite this outstanding question, it does appear that regular expressions have contributed something of an improvement, albeit at very great computational cost.

4.8 Relevance Scoring

Some training runs and some unofficial TREC-5 runs (see table 4) investigated the effect of different parameters in the distance-based relevance scoring model.

A number of different functions have been used to estimate the declining weight of relevance evidence as span length increases. Inverse square-root and a custom function (figure 3) were used in runs reported here.

A cut-off proximity limit beyond which the probability of relevance is forced to zero is also applied for efficiency. Its value has also been found to have an effect on precision-recall. If set too high or too low recall and average precision are adversely affected. As shown in table 4, two different values were used in TREC-5 runs.

5 Simulated Server Selection and Result Merging

DB Merging Track, Official Runs `anu5mrg0, 1 and 7`

Full details of experimentation in the context of the database merging task are given elsewhere [7]. The merging task was interpreted as a simulation of a network server selection problem. In this model, each of the 98 sub-sub-collections was served by a distinct network server, simulated by a processing node on ANU's Fujitsu AP1000 (leaving 30 unused nodes). The Q_0 query set was used to avoid using collection information in query generation.

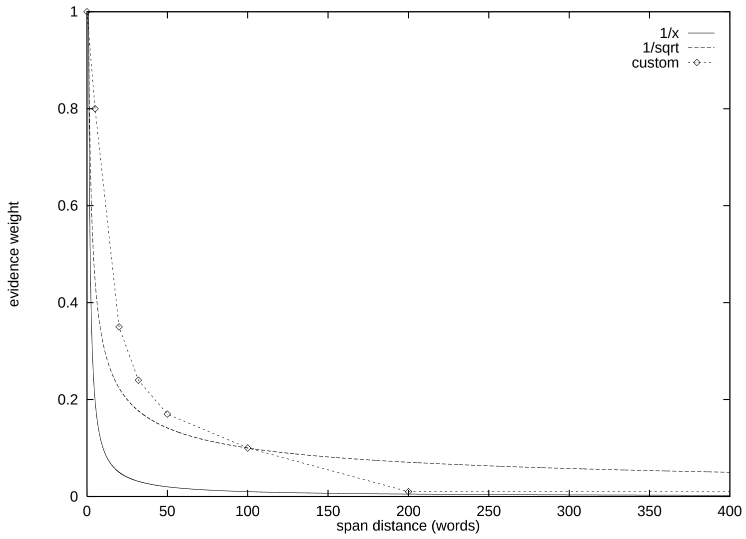

Figure 3: Some possible relationships between strength of evidence represented by a span and its length.

Table 5: Summary of DB Merging runs. All runs used manually-generated query set Q_0. Percentages in columns 4-6 are relative to the baseline run. Columns 4 and 5 refer to the `trec_eval` measures using the official relevance judgments. Column 6 records the percentage of documents retrieved by the baseline which were also retrieved by the run in question. Figures in parentheses have been scaled-up by $\frac{32}{30}$ as a first-order compensation for the difference in number of sub-sub-collections used.

Run-id	Synopsis of Method	No. Servers	% rel_ret	% ave_prec	% ret
`anu5mrg0`	baseline	98	100	100	100
`anu5mrg1`	Topic Similarity	32	63	73	53
`anu5mrg7`	Lightweight Probes	30	59(63)	61(65)	47(50)

As in TREC-4, distance-based relevance measures only were used. Consequently, the problem of merging rankings from different sub-collections is a non-issue as the distance measures are independent of collection statistics. Results of runs `anu5mrg0` and `anu5man1` are close to identical, as they should be, [8] despite the fact that the former used the database merging division of the collection.

Three runs constitute ANU's submissions in the DB Merging track. These are summarised in table 5.

The first method (`anu5mrg1`) used TREC-4 topics and queries processed over the entire collection of servers as historical data and sent TREC-5 queries to servers which had proven useful in processing TREC-4 topics which were manually judged to lie in similar subject areas. Similarity judgments are given in appendix A.

The second method (`anu5mrg7`) used no historical information but instead sent light-weight (two-term) probe queries to all servers and used a small packet of frequency information returned in response to select a subset of servers to process the full query.

The goal was to retrieve as high a fraction of relevant documents as possible using only a small subset (about one third) of the available servers.

Results: No significant difference was observed between the methods. Roughly

Table 6: Unofficial Confusion Results. The highest ranked document has rank $R = 0$. Items not found are assigned $R = 1000$.

Run-id	Queries	Collection	No. found	Ranked 0	Ranked < 20	Ave. rank
`anu5con0`	Q_{con0}	Truth	47	14	35	84
`anu5con3`	Q_{con0}	Degrade5	44	9	23	197
`anu5con1`	Q_{con1}	Degrade5	47	12	30	109

speaking, each method retrieved nearly two thirds of the relevant documents retrieved by the baseline, while accessing only about one third of the servers. Note that, in retrieving two thirds of the baseline relevant documents, it was only necessary to retrieve half of all the documents retrieved by the baseline run.

Discussion and Conclusions: Both server selection methods appear to be capable of biasing server selection toward servers which supply documents which achieve non-zero scores and of even more strongly biasing towards those which supply actually relevant documents. The good performance of the lightweight probe method is encouraging as historical query-processing data required by other methods is not always available.

6 Information Retrieval Over OCR-Corrupted Data

Confusion Track, Official Runs `anu5con0` and `anu5con1`

A set of queries Q_{con0} were manually devised using an average of only 3 minutes of human time per topic. They were again based entirely on span scoring. These queries were processed against the "truth" version (`anu5con0`) and the "degrade5" version of the data (`anu5con1`).

Full detail of the the method employed is documented in [5]. In summary, characteristic scanning errors were identified in a small sample of the text by comparing truth and degrade5 versions. All combinations of presence/absence of these characteristic errors in each term were then applied to the Q_{con0} queries by a preprocessor, resulting in significantly longer queries Q_{con1}.

Results: Using official expected run-lengths, ANU's baseline run was 65% worse than the median and the degrade5 run was 48% worse. Relative to the best in each category, the corresponding figures are 1,684% worse and 589% worse. Expected run-lengths were 65% of the worst on the baseline and 22% of the worst on degrade5.

Discussion and Conclusions: It is tempting to conclude that the 3-minute ANU queries were very poor but that the method for compensating for OCR errors was relatively effective. It might be considerably more so if a more systematic approach were taken to identifying the characteristic errors.

7 Experiments with a Larger Collection

Very Large Collection Track, Official Runs `anu5vlc2` and `anu5vlc3`

The collection used in the VLC pre-track comprised all four TREC CDs, a total in excess of 4 gigabytes. This factor of 2 increase over the TREC mainstream task is potentially significant as it goes beyond the amount of data which can be addressed using 32 bits.

In the hope of boosting early precision, VLC runs used the inverse square root of distance as the decay function rather than the custom function and imposed a much more severe penalty on scores derived from partial spans. (In the VLC runs the shortest partial span involving $k-1$ terms would score just less than the longest admissible span involving k terms. In the manual run, a partial span involving $k-1$ terms would score 0.1 of the score for a k term span of the same length.)

The baseline run was carried out over the data as organised for the DB Merging run, a super-dictionary linking 5 distributed index files, and only 98 processing nodes were used. The VLC run proper used 9 distributed indexes, the five from DB Merging and two for each of CD1 and CD3.

Table 7: Summary of VLC measures for the ANU submissions.

Measure	`anu5vlc2` Baseline	`anu5vlc3` VLC	VLC/Baseline
Precision@20	0.3920	0.5020	1.28
Query processing time	44.8	68.5	1.53
Data struct. bld. time	2177	4721	2.169
Disk space	6.29 gB	10.29 gB	1.63
Memory			

Results: Table 7 shows the VLC measures taken from the two ANU runs in the trial VLC track. Both runs used the Q_{RF2} queries. The baseline run for the VLC task is thus comparable to the **`anu5man4`** manual adhoc run except for a different model of span scoring. It is not clear how to report memory use on the parallel machine. Since the current operating system enforces a single-user mode of operation, one could say that memory use for both methods is 2 gigabytes, even though that includes space for kernels, PADRE executables, message buffers and unused freespace, all replicated 128 times.

Using NIST assesments, the VLC baseline run scored 0.3920 on precision@20 compared with 0.3840 for **`anu5man4`**.

Discussion and Conclusions: Early precision was significantly better on the 4 gigabyte collection than on the 2 gigabyte collection. ($t(49) = 3.243, p < 0.05$, observed difference +28%.) A similar phenomenon was observed by other participants in the task. Unfortunately, because the two sets of results were judged by different assessors, difference between judges cannot be ruled out as an explanation. However, it may be that a small class of relevant documents contains so many obvious relevance features that its members will appear ahead of [nearly] all irrelevant documents in any reasonable ranking scheme. Documents outside this obviously relevant class may not be so reliably ranked. If this supposition is correct, doubling the size of the collection, will probably double the size of the obviously relevant class, leading to higher early precision. At present, this is mere conjecture and needs to be tested.

Data structure building time for the 4 gigabyte case is a little over double the comparable figure for the 2 gigabyte collection. Given the PADRE super dictionary architecture, it would be expected that index and dictionary building time would rise linearly with the amount of data but that building the super dictionary would require approximately three times as much I/O. Since super dictionary building for this size of collection takes only a fraction of the index building time, the observed ratio is quite within the range of expectation.

Query processing time increased by 53% when moving to the larger collection. An 80% increase might have expected on the basis of the increase from 5 to 9 in the number of superdictionary components. However, the fact that the CPU load of processing the additional four SD components is spread over 128 nodes rather than just 98 is a compensating factor. There are also some fixed costs, such as resetting at the beginning of a topic and returning rankings at the end, which are independent of the amount of data.

Disk space requirements are rather large because the HiDIOS filesystem [11] is organised for time efficiency rather than for minimising space. Significant imbalance between nodes in the DB Merging layout and the fact that 30 nodes have no data at all in this layout mean that parallel files include large amounts of unused space. In fact, the raw text for CD2 and CD4 occupies a total of 3.56 gigabytes in this layout! The fact that space requirements do not rise in proportion to the amount of data is indicative of the fact that the CD1/CD3 data is much better balanced on the machine.

8 Overall Summary and Conclusions

Taking into account the increased difficulty of the TREC-5 task and the fact that no query augmentation was employed, ANU/ACSys techniques for automatic query generation performed well and should constitute a sound base for eventual top-rank performance when coupled with a good query expansion system.

The span scoring model is capable of top-flight precision-recall results, subject to the use of good queries. Methods for refining span queries using term-term implications and pseudo relevance feedback conferred benefit but not as much as had been hoped. Taking into account the difficulty of the task and the use of interactive query development by most other groups, the ANU/ACSys manual adhoc queries performed quite well despite reduced development times and avoidance of subject-expert consultation. Clearly, the non-interactive manual approach to query formulation is unrealistic and an interactive framework for span-query development is an obvious direction for future PADRE development.

Further work is needed to confirm the validity of the techniques used to overcome OCR degradation. ANU/ACSys results in this category were hampered by lack of time to develop good queries, to systematically observe characteristic errors and to work with the heavily degraded dataset.

The DB Merging track provided a springboard for an extensive series of experiments which will be reported elsewhere. The use of the Fujitsu AP1000 to simulate the operation of distributed information servers is an interesting application of large-scale parallelism in IR.

The benefits of parallelism would be expected to show themselves most strongly in the VLC track. Good scalability was demonstrated by the ANU/ACSys VLC submissions. However, the two-fold increase in data size in the TREC-5 pre-track was too small to fully test the potential benefits. Nonetheless, experience gained in the pre-track has suggested ways to ensure better performance and scalability over the planned 20 gigabyte corpus. The observed increase in average precision as data size increased needs confirmation and, if necessary, explanation.

Acknowledgements

The cooperation of our colleagues at the University of Waterloo in making available their TREC-4 queries is gratefully acknowledged.

References

[1] Peter Bailey and David Hawking. A parallel architecture for query processing over a terabyte of text. Technical Report TR-CS-96-04, Department of Computer Science, Australian National University, `http://cs.anu.edu.au/techreports/1996/`, 1996.

[2] Chris Buckley, Amit Singhal, Mandar Mitra, and Gerard Salton. New retrieval approaches using SMART: TREC 4. In Harman [4], pages 25–48.

[3] C. L. A. Clarke, G. V. Cormack, and F. J. Burkowski. Shortest substring ranking multitext experiments for TREC-4. In Harman [4].

[4] D. K. Harman, editor. *Proc. Fourth Text Retrieval Conference (TREC-4)*, Gaithersburg, MD, November 1995. U.S. National Institute of Standards and Technology.

[5] David Hawking. Document retrieval in OCR-scanned text. In *Proc. Sixth Parallel Computing Workshop*, Kawasaki, Japan, November 1996. paper P2-F.

[6] David Hawking and Peter Bailey. *PADRE v. 2.4 User Manual.* `http://cap.anu.edu.au/cap/projects/text_retrieval/`.

[7] David Hawking and Paul Thistlewaite. Methods for information server selection and result merging. In preparation.

[8] David Hawking and Paul Thistlewaite. Proximity operators - so near and yet so far. In Harman [4], pages 131–143.

[9] David Hawking and Paul Thistlewaite. Relevance weighting using distance between term occurrences. Technical Report TR-CS-96-08, Department of Computer Science, Australian National University, `http://cs.anu.edu.au/techreports/1996/index.html`, 1996.

[10] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46(4):359–364, 1990.

[11] Andrew Tridgell and David Walsh. The HiDIOS filesystem. In *Proc. Fourth International Parallel Computing Workshop*, pages 53–63, London, September 1995. Imperial College/Fujitsu.

Appendix A: Topic-by-Topic Performance of Best Distance-Based Queries

topic	ret	rel_ret/rel	%best	%median	@10	ave_prec	%best	%median
202	211	158/283	62%	87%	10	0.4952	68%	162%
203	102	24/33	83%	126%	4	0.2707	82%	347%
204	466	113/397	38%	55%	4	0.1215	34%	93%
205	1000	198/310	99%	1320%	10	0.3929	79%	14552%
206	1000	25/47	83%	312%	2	0.0597	46%	728%
207	1000	69/74	93%	100%	7	0.5805	90%	112%
208	1000	31/54	67%	111%	0	0.0514	21%	101%
209	1000	74/87	90%	101%	5	0.2711	77%	130%
210	1000	54/57	95%	108%	8	0.6084	79%	113%
211	1000	231/323	86%	186%	7	0.3781	66%	325%
212	1000	139/153	99%	145%	6	0.4514	80%	258%
213	57	16/21	80%	114%	6	0.4522	127%	203%
214	26	3/5	60%	60%	2	0.3462	51%	70%
215	1000	172/183	97%	108%	10	0.6154	98%	125%
216	512	34/36	97%	117%	7	0.5309	82%	129%
217	498	43/57	75%	154%	10	0.4620	76%	228%
218	12	7/46	17%	22%	5	0.0930	27%	89%
219	1000	96/133	99%	128%	3	0.1677	77%	172%
220	49	16/24	67%	73%	9	0.5226	80%	115%
221	1000	131/181	87%	116%	9	0.3112	64%	130%
222	74	44/74	64%	80%	10	0.5312	120%	211%
223	1000	147/363	57%	109%	6	0.1673	45%	163%
224	696	127/149	85%	134%	7	0.4229	88%	163%
225	1000	215/216	100%	119%	7	0.6985	96%	126%
226	1000	137/145	108%	326%	7	0.3888	94%	523%
227	1000	240/347	100%	189%	10	0.5097	107%	389%
228	1000	14/66	30%	56%	0	0.0066	6%	26%
229	424	20/21	95%	111%	7	0.5731	99%	154%
230	1000	82/85	98%	164%	10	0.6554	82%	495%
231	204	15/23	65%	68%	3	0.1631	70%	201%
232	186	5/9	71%	125%	1	0.1230	39%	1108%
233	280	110/121	101%	234%	9	0.6157	97%	1251%
234	57	26/28	96%	113%	9	0.7029	90%	166%
235	306	160/197	85%	150%	10	0.7171	89%	317%
236	1000	30/43	83%	333%	3	0.1080	104%	3484%
237	1000	180/215	97%	141%	8	0.4872	84%	203%
238	1000	220/270	86%	167%	5	0.3513	65%	316%
239	1000	69/123	77%	147%	3	0.1224	58%	334%
240	1000	173/276	88%	216%	8	0.2667	87%	516%
241	412	21/62	38%	131%	3	0.0798	25%	706%
242	396	33/38	89%	114%	9	0.5197	94%	374%
243	451	19/69	31%	50%	2	0.0350	15%	57%
244	1000	359/431	89%	120%	10	0.6201	94%	160%
245	1000	25/43	78%	147%	5	0.2021	93%	355%
246	1000	177/286	98%	174%	9	0.3132	101%	208%
247	76	28/36	80%	122%	5	0.4073	76%	119%
248	1000	113/122	109%	365%	7	0.4241	76%	1140%
249	1000	32/53	67%	86%	6	0.2063	73%	140%
250	1000	73/86	91%	149%	4	0.2040	61%	273%
49	33495	4528/6501	83%	137%	6.3	0.3634	78%	189%

Appendix B: Topic Similarities Between TREC-5 and TREC-4

These similarities were used in run `anu5mrg1` in the Database Merging track.

TREC-5 topic	TREC-4 topics in related area
251	203, 218, 219, 242, 244, 246
252	205, 209, 212, 240
253	205, 226, 232, 239
254	210, 216, 224, 229, 239
255	228, 243, 249
256	205, 206, 215, 226, 238
257	210, 215, 224, 239, 250
258	212, 223, 235, 240, 248
259	206, 222, 250
260	205, 216, 217, 239, 242
261	202
262	213, 214, 229, 232
263	213, 216, 231, 243
264	227, 236, 240, 247
265	211, 221, 235, 242, 250
266	220, 228, 248
267	225, 238
268	246
269	212, 219, 244
270	210, 216, 231, 243
271	204, 208, 230, 237, 249
272	210, 224, 229, 239
273	213, 217, 225
274	230
275	210, 216, 231, 243
276	205, 206, 215, 226, 238
277	202, 227, 236, 240, 246
278	217, 224, 213
279	213, 217, 240, 249
280	228, 236, 240, 249
281	213, 216, 224
282	221, 236, 250
283	219, 244, 246
284	221, 222, 235, 240, 250
285	202, 246
286	203, 218, 242
287	222, 235, 250
288	214, 216, 224, 232, 239
289	210, 216, 224, 226, 241
290	211, 219, 230, 237
291	209, 226, 231, 247
292	209, 245
293	207, 227, 246
294	208, 220, 238, 245
295	206, 207, 227, 235
296	206, 207, 231, 232, 241
297	206, 207, 216, 235, 245
298	221, 250
299	219, 226, 227, 246
300	206, 207, 227, 235