Information Retrieval 2, 165–176 (2000)

The TREC-5 Confusion Track: Comparing Retrieval Methods for Scanned Text

PAUL B. KANTOR*
Department of Library and Information Science, Rutgers University, 4 Huntington St. New Brunswick, NJ 08901, USA

ELLEN M. VOORHEES
National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899, USA

Received April 1, 1999; Revised December 27, 1999; Accepted December 27, 1999

Abstract. A known-item search is a particular information retrieval task in which the system is asked to find a single target document in a large document set. The TREC-5 confusion track used a set of 49 known-item tasks to study the impact of data corruption on retrieval system performance. Two corrupted versions of a 55,600 document corpus whose true content was known were created by applying OCR techniques to page images. The first version of the corpus used the page images as scanned, resulting in an estimated character error rate of approximately 5%. The second version used page images that had been down-sampled, resulting in an estimated character error rate of approximately 20%. The true text and each of the corrupted versions were then searched using the same set of 49 questions. In general, retrieval methods that attempted a probabilistic reconstruction of the original clean text fared better than methods that simply accepted corrupted versions of the query text.

Keywords: optical character recognition (OCR), text retrieval, evaluation, TREC

1. Introduction

The Text REtrieval Conference (TREC) is a workshop series designed to encourage research on text retrieval for realistic applications by providing large test collections, uniform scoring procedures, and a forum for organizations interested in comparing results (Harman 1995). Started in 1992, the conference is co-sponsored by the National Institute of Standards and Technology (NIST) and the Defense Advanced Research Projects Agency (DARPA). For each TREC, NIST provides a test set of documents and questions. Participants run their retrieval systems on the data, and return to NIST a list of the top-ranked retrieved documents. NIST pools the individual results, judges the retrieved documents for correctness, and evaluates the results. The annual TREC cycle ends with a closed workshop at which participants share their experiences.

The first TREC conferences involved just two main tasks distinguished by the presence (*routing*) or absence (*adhoc*) of previously judged "training" documents. Additional subtasks known as "tracks" were introduced into TREC in TREC-4 (1995). While the main TREC tasks provide an entry point for new participants and provide a baseline measure of

*Supported in part by the NSF under Grant IIS 98-120086.

retrieval performance, the tracks invigorate TREC by focusing research on new areas or particularly difficult aspects of text retrieval.

One of the first tracks to be introduced was the Confusion Track, which is a track that investigates how retrieval performance is affected by noisy, or “confused”, text. This track was motivated by the growing interest in providing electronic access to legacy documents by scanning originals and capturing the output of the optical character recognition (OCR) process. Unfortunately, OCR processing has substantial error rates for poor quality originals. This track was designed to promote research on best methods for providing access to such documents.

This paper presents the findings of the TREC-5 Confusion Track. The next section defines the specific task, the data set, and the evaluation methodology used. Section 3 describes the different approaches taken by the participants in the track and presents the individual results. The final section presents a new method for interpreting and presenting the results of this type of multi-part task. Using a generalization of the idea of an operating characteristic, we represent the performance of each system in terms of the accumulated retrieval of target documents, plotted against the total effort required to attack all problems simultaneously.

2. Experimental method

The confusion track studied a particular type of retrieval problem called *known-item searching* (see Table 1). A known-item search simulates a user seeking a particular, partially-remembered, document in the collection. In contrast to a more standard retrieval search where the goal is to retrieve and rank the entire set of documents that pertain to a given subject of interest, the goal in a known-item search is to retrieve just that one particular document. Known-item searching is well-suited to the problem of retrieving noisy data because it stresses those parts of a retrieval system that are most affected by noise. OCR techniques generally misinterpret low-frequency words such as proper nouns and technical terms more often than common words. Yet low-frequency words are high content-bearing words, and are precisely the words likely to be used to locate a specific document.

2.1. The data

Table 1 provides examples of the known-item questions used in the track. The questions were created by five NIST staff members (“authors”) who developed ten questions each.[1]

Table 1. Example known-item questions from the TREC-5 confusion track.

- Use of solar power by the Florida energy office.
- Excessive mark up of zero coupon treasury bonds.
- I am looking for a document about the dismissal of a lawsuit involving Adventist Health Systems.
- I am looking for theft data on the Chevrolet Corsica.
- efforts to establish cooperative breeding programs for the yellow crowned amazon parrot.
- morphological similarities between different populations of saltwater crocodiles.

The questions were designed to target one document from the *1994 Federal Register*, the corpus used in the track. Different authors used different techniques to construct their questions (for example, using an index of the collection to find unique words or starting with “interesting” documents and adding conditions to the question to ensure uniqueness), but the authors did *not* specifically pick documents that they thought would be difficult for the OCR process. The authors made every effort to ensure that only one document was an answer to the question, and there have been no reported problems with this assessment.

As mentioned, the corpus for the track was the set of documents contained in the 1994 edition of the *Federal Register*. The United States Government Printing Office (GPO) prints the *Federal Register* as a record of the transactions of the government. One issue is published each business day and contains notices to Federal agencies and organizations, executive orders and proclamations, proposed rules and regulations, etc. The *Federal Register* was selected for these experiments because it is a large collection for which both hardcopy and electronic versions are readily available. The corpus contains 395MB of text divided into approximately 55,600 documents.

Participants in the track were given three different text versions of the *Federal Register* corpus. The first version was derived from the typesetting files provided by the GPO; this was regarded as the ground truth text version of the collection and was used to design the questions. The second version was the output obtained by scanning the hardcopy version of the corpus. The estimated character error rate of this version of the corpus is 5%. The final version of the corpus was obtained by downsampling the original page images produced above and scanning the new images. The estimated character error rate of the downsampled images is 20%.[2]

2.2. Evaluation

The task in the track was to locate the target story for each question in each of the three versions of the corpus. Participants submitted to NIST a ranking of the top 1000 candidates per question per corpus. They were free to use any retrieval approach they desired, provided no information from one version of the corpus was used to search another version of the corpus. For example, this restriction prohibited groups from expanding queries using the clean collection and running those expanded queries on a degraded version of the corpus. The design of the track permits study of how individual retrieval approaches deal with OCR character error rates, and also comparison of the effectiveness of different retrieval approaches when using noisy data.

The most basic evaluation of how well a retrieval system performs a known-item search is to look at the rank at which the target document is retrieved. Summary measures are also needed to facilitate comparison. Several different evaluation measures are given for each system in the confusion results in the TREC-5 proceedings (Voorhees and Harman, 1997). In this paper we report only “mean-reciprocal-rank”. The mean-reciprocal-rank is the mean of the reciprocal of the rank at which the known item was found, averaged over all the queries, and using 0 as the reciprocal for queries that did not retrieve the known document.

As a measure, the mean of the reciprocals of the ranks has several advantages compared to the mean of the ranks themselves. The mean reciprocal is bounded between 0 and 1 (with 1 representing perfect retrieval), so its value is interpretable without knowing how many documents were ranked. Furthermore, the mean rank is greatly influenced by target documents that are retrieved at large ranks, which does not reflect the importance of those documents in practice. In contrast, the mean reciprocal still penalizes runs that do not retrieve a known item, but minimizes the difference between, say, retrieving a known item at rank 750 and retrieving it at rank 900. An additional benefit is that, since there is only one target document, the mean reciprocal is mathematically equivalent to the summary measure most commonly used to report ranked retrieval results, mean average precision, and also equal to precision at 100% recall. This gives researchers familiar with ranked retrieval evaluation a feel for the known-item search results.

3. Overview of retrieval schemes

The guidelines for each TREC track specify the number of runs a participant may submit to NIST. For the TREC-5 Confusion Track, participants were permitted to submit one or two run sets, where each set consisted of a run on the three different versions of the corpus (though not all participants ran complete sets). A run on the true version of the corpus was considered a baseline run, and results from the degraded versions of the corpus were compared to the corresponding baseline. Different sets from the same participant used different methods to compensate for the corruption and/or different retrieval methods. Table 2 lists the different runs sets submitted to the track.

Five broadly different retrieval methods were applied in the confusion track. They form a progression in terms of the detail with which they attempt to discern the correct text underlying the corrupted version. A summary of each these methods is presented here. For specific details about a particular approach, see the appropriate paper in the TREC-5 proceedings.

Rutgers SCILS APLab: Ng et al. (1997) based their retrieval on the number of approximate matches of query words that could be found in each document. An approximate match was defined in terms of “double-dot-5-grams”. A double-dot-5-gram is a 5-gram where one position is a don’t-care character that can match 0, 1, or 2 other characters. For each

Table 2. Mean-reciprocal-rank scores for runs submitted to the TREC-5 confusion track.

Run Name	Submitter	Baseline	5%	20%
rutcf	Rutgers	–	.2041	.1174
anu5con	ANU	.3635	.2992	–
gmu961	GMU	.3856	.3135	.2221
gmu962	GMU	.2039	.1900	.1524
CLCON	CLARIT	.7293	.4024	.2138
CLCONF	CLARIT	.7293	.2297	.1898
ETHFR94N	ETH	.7353	.5737	.3218
ETHFR94P	ETH	.7353	.3720	.4978

query, stop words were removed (using the SMART stopword list) and the complete set of 5-grams contained in the remainder of the query was generated. The 5-grams could not cross word boundaries, so, for example, the query “red balloon” would generate the set {red, ballo, alloo, lloon}. Each 5-gram was then converted into a set of double-dot-5-grams by substituting each character in turn with the don’t-care character. (Thus, “ballo” produces {?allo, b?llo, ba?lo, bal?o, and ball?} as its set of double-dot-5-grams where ‘?’ is used to denote the don’t-care character.) A score was produced for each line in the corrupted text by counting the number of double-dot-5-gram matches the line contained. The final score for a text was the average of its line scores. Note that this scheme gives higher weight to uncorrupted texts since the uncorrupted version of a query word matches each 5-gram the word generates for each of the 5-gram’s don’t-care positions.

Australian National University (ANU): The ANU group (Hawking et al. 1997) “corrupted” their queries and then used their standard span-scoring-based retrieval method to retrieve documents from the degraded collection. Span-scoring is a retrieval technique in which the similarity of a document and a query is based on the proximity of the concepts of the query in the document. For the TREC-5 confusion track, queries were manually generated using an average of about 3 minutes per question to generate a query. The manually generated queries were run as is against the uncorrupted documents to form the baseline ANU run. These queries were then corrupted using a set of characteristic scanning errors discovered from a small sample of the clean and 5% degraded texts. All combinations of a characteristic error being present or absent for a particular query term were put into the corrupted version of the queries. That is, each query was expanded by the addition of corrupted terms likely to appear in the text. The corrupted queries were then run against the corrupted text to produce the final retrieval results.

George Mason University (GMU): The GMU group (Grossman et al. 1997) submitted two run sets. Each set used overlapping 4-grams, including term boundaries, as terms. Stop words were eliminated before the formation of the 4-grams, and the 150 most common 4-grams were also eliminated. Retrieval was based on *tf.idf* weights and the cosine similarity measure. The ‘gmu961’ run set was the result of this basic processing, while the ‘gmu962’ run set also incorporated automatic feedback. For the automatic feedback, the 4-grams in the top 10 documents of the basic run were sorted by $n_i * idf_i$, where n_i is the number of times the ith 4-gram occurred in the ten documents and idf_i is the inverse document frequency of the ith 4-gram. The top 20 4-grams were added to the query, each with a weight equal to 0.4 times its natural *tf.idf* weight. The expanded query was used with the cosine measure to produce the final ranked list of documents.

CLARITECH Corporation (CLARIT): In this approach (Tong et al. 1997), stochastic methods were applied to the documents to correct corrupted words on a sentence-by-sentence basis. Federal Register data from 1988 and 1989 was used to estimate word frequencies, and word-word transition (bi-gram) probabilities. Correction was applied only to words that did not have an exact match in the lexicon (call these *c-words*). For each such c-word, up to 200 candidate corrections were ranked by their probability of matching the corrupt word. The top 10 candidates for each c-word were retained for sentence processing, which used the Viterbi algorithm to get the most likely word sequence for the sentence. Thus, CLARIT produced a single sentence-based, maximum-likelihood assignment for every c-word. The CLARIT group notes that their method for correcting text assumes that word boundaries are reliable—an assumption that is frequently violated in the 20% word error rate collection.

Retrieval was accomplished by using standard CLARIT indexing and search procedures on the corrected text. The natural language queries were submitted to the CLARIT indexer, which parsed them into single terms and phrases. For the baseline run (i.e., on the uncorrupted text) and the 'CLCON' runs, these queries were used as is, without expansion. For the 'CLCONF' runs, the original queries were expanded using automatic feedback and the final ranking was produced by the expanded queries.

Swiss Federal Institute of Technology (ETH): The ETH group (Ballerini et al. 1997) made use of several devices. Document scores were computed using both a pivot method (which controls for the effect of overly long documents) and term contributions. The term contributions included $f(\phi, t)$, the observed frequency of a term ϕ in text (document or query) t; $d(\phi)$, which measures the prevalence of term ϕ in the collection; and f^*, a corrected estimated frequency for terms in document texts. The estimated correct frequencies are determined in three steps. First, the document is divided into overlapping slots, any of which might contain the term. A slot is used for further computation if it contains at least a fraction P of the term's characters. The number of slots such that the edit distance between the slot contents and the term is less than 20% of the length of the term is determined. The probability that a term appears in a slot is set to a nominal value for each matching slot, and then summed over all the slots in the document. This sum is then multiplied by a constant which makes the estimate of term frequency more accurate, as determined by regression analysis applied to a set of 100 documents used in clear, 5% and 20% corrupted forms. The terms used are (Porter) stems, including the preceding white space. In sum, the ETH method gives a document credit for all of the terms which have a "sufficiently close match" to the noisy text as it is presented. Since all terms are words or initial substrings of words, the method could have trouble with corruption of the word separation characters.

Because of the computational cost, this detailed calculation was carried out only for a set of 2000 documents for each query. The 2000 documents were the top 2000 as retrieved using *tf.idf* weighted *n*-grams as terms and pivot document length normalization. 4-grams were used for the 5% corrupted data and 3-grams were used for the 20% corrupted data. The retrieval runs produced by taking the top 1000 documents from this basic processing were submitted as the 'ETHFR94N' run set. The 'ETHFR94P' run set used the detailed processing described above.

Table 2 lists the mean-reciprocal-rank scores obtained for each of the submitted runs. The means were computed over the 49 questions used in the track. All runs but one show a noticeable drop in effectiveness as the noise in the documents increases. (The lone exception, the ETHFR94P 5% run, had a mistake in the weighting function used, so that a relatively large number of terms were given negative weights). These results thus demonstrate that even a moderate character error rate of 5% can depress retrieval capability.

The different approaches used by the participants vary in their treatment of both the query and of the corrupted texts. They appear to form a progression in the following sense:

- Rutgers expanded the terms appearing in the query by a 5-gram sliding window with each character replaced with any set of 0, 1 or 2 characters. The 5-grams did not cross word boundaries. Retrieval ranking was based on the average number of hits per line of text. This probably discriminates much too strongly against long documents. Performance was poor.
- ANU expanded queries based on corruption errors found likely in a study of a sample of corrupted text. Thus additional terms (which might in principle be words in a lexicon) were added to the query.
- GMU resolved both query and documents into overlapping 4-grams, judged to be more resistant to corruption, and required an exact match. Special stop-lists of 4-grams were constructed. Queries were expanded by a method based on preliminary retrieval from the corpus, resulting in the addition of new 4-grams.

These three methods represent expansion of the query, in an effort to include or match corrupted forms that either might (Rutgers), or could (ANU), or sometimes do (GMU) arise under the corruption observed. Results for the first of these methods were relatively weak; the second method was not applied to the most severely corrupted data, and the third exhibited somewhat surprising performance detailed below.

The remaining two methods sought to “expand” or “clarify” the corrupted texts.

- CLARIT used statistical methods to replace each non-word by a word which makes the entire resulting sentence most likely in some well-defined sense. Each non-word is replaced by exactly one word.
- ETH, in effect, replaced each “slot” (which might be occupied by a word in the corrupted text) by a vector of candidate words, each of which is permitted to contribute to the computed similarity to the question. This is, in principle, a wider expansion of the corrupted text, since the second ranked candidate can enter the computation in this method, but not in the CLARIT method.

4. Comparative performance using operating characteristics

We will use a Generalized Retrieval Operating Characteristic (Kantor 1997) to obtain a more detailed view of comparative performance. In contrast to precision and recall measures, this approach contrasts the benefits achieved by using a system with the cumulated human effort required to achieve that benefit. This permits users with different value schemes to, in principle, arrive at different preferences among systems, based on the value of the target documents and the cost of scanning retrieved documents.

Let $r(i)$ be the rank of the sought item for question i. If we consider the cumulated value delivered by a set of ranked lists to be proportional (with constant v) to the number (G) of target documents found (out of a total of S target documents), and cost as proportional (with constant c) to the number of documents which must be examined before reaching them, the corresponding measure of value has three terms. The first term is the value of all documents found; the second term is the cost of finding those documents; and the third term is the cost of not finding the remaining documents.

$$\begin{aligned} V &= vG - c \sum_{i\ found} r(i) - 1000(S-G)c \\ &= G(v + 1000c) - c \sum_{i\ found} r(i) - 1000S \end{aligned}$$

Hence systems would be ranked according to:

$$G(v + 1000c) - c \sum_{i\ found} r(i)$$

This is the same as ranking them according to:

$$G\left(\frac{v}{c} + 1000\right) - \sum_{i\ found} r(i)$$

In other words, the relative importance of finding a document at all, compared to the importance of placing found documents high in the list, depends in an unavoidable way on the cost assigned to examining documents. Thus there is no single measure which covers all reasonable opinions about this parameter.

On the other hand, one may extend the idea of a Retrieval Operating Characteristic to compare systems. The extension is to imagine that all retrieved lists are perused in parallel, and to ask how many of the documents have been found, when r documents have been examined on each list. This performance curve may be calculated directly from the reported lists by computing the cumulated number of documents examined up to each "hit". In effect, we imagine that for a set of S questions, there are S analysts who work in parallel. Each examines the next document in the list for her problem. As soon as she finds the desired document, she stops working on this task. If she reaches the end of the list she stops working. The process continues until all the analysts have stopped working. We plot the number of target documents that have been discovered against the number of documents that have been examined in all rounds prior to its discovery.

If the curve for one scheme lies everywhere above the curve for another then, at least for this set of target documents, it delivers greater value, whatever the value (v) assigned to good and cost (c) assigned to bad documents. If neither curve is always above the other, then we cannot definitely state that one scheme is to be preferred to another. As with other measures currently used in information retrieval evaluation, the statistical significance (confidence levels, confidence intervals, etc.) this type of comparison is not known, particularly for the case of multiple comparisons.

The overall confusion track results roughly parallel the order of generality of the nets cast by the methods as described in Section 3. Figure 1 uses the rank of the target document as the abscissa. In figure 2 we show the results using the economically more meaningful measure $w =$ [cumulated total number of items examined]. We show only the performance achieved by each team on the 20% degraded materials. Conveniently, almost every teams' better effort dominated its weaker effort, in the sense described above. (The exception is

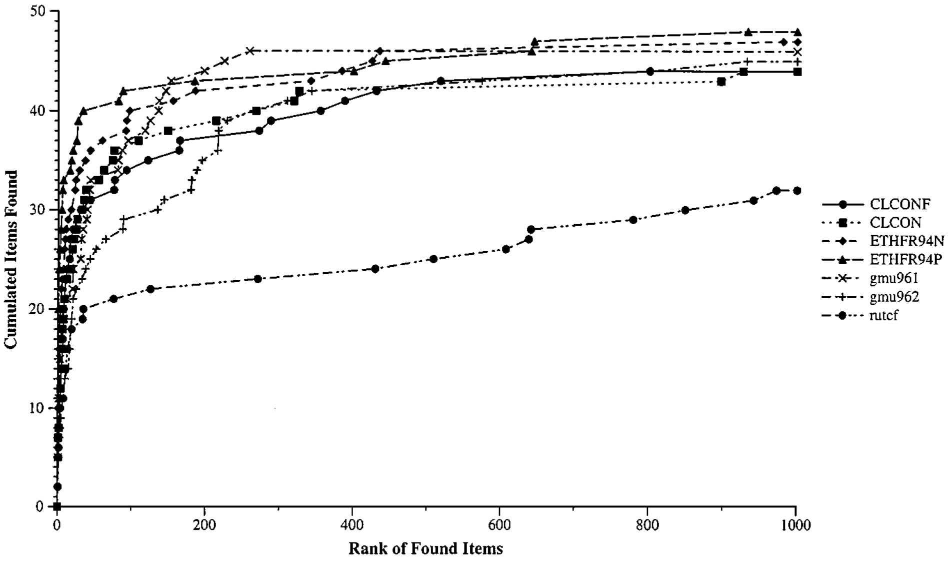

Figure 1. Cumulated target documents as a function of the rank in the lists of documents examined. Data are for 20% corrupted text.

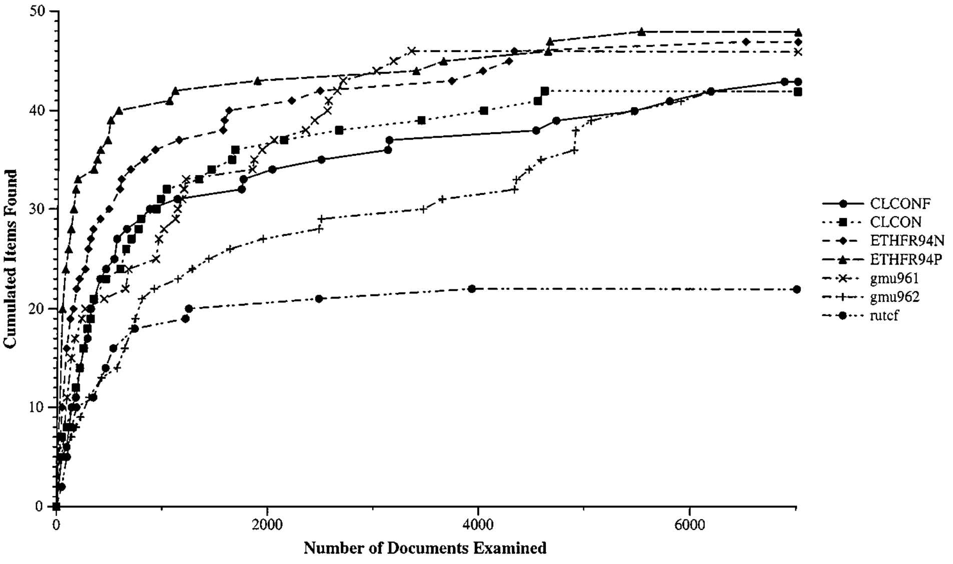

Figure 2. Cumulated target documents as a function of the total number of documents examined. The systems of figure 1 are shown here. The number of documents examined is a non-linear function of the rank at which items are found. This causes a change in the appearance of the curves. Note that the crossing of *gmu961* and *ETHFR94P* covers a much shorter range when presented in this way than in figure 1. Data are for 20% corrupted text.

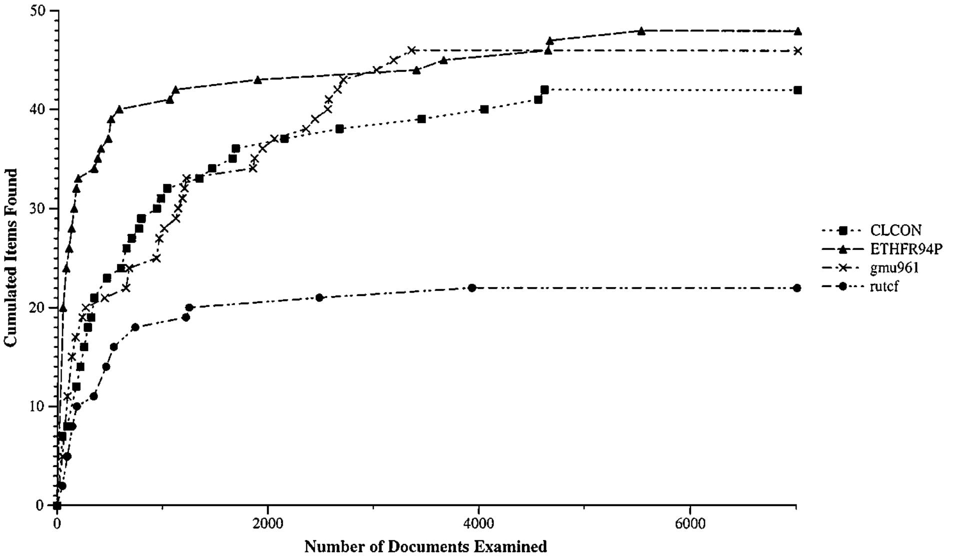

Figure 3. Cumulated target documents as a function of the total number of documents examined. Only the dominant system from each group is shown. Data are for 20% corrupted text.

ETH, for which the poorer scheme does eventually surpass the better scheme, measured against the work involved.) In figure 3 we show only the results for each team's better effort. ANU is not represented because this group did not submit a report for the 20% degraded material. Rutgers submitted only one report, as shown here.

Study of figure 3 reveals that, first of all, the simple pattern matching scheme (rutcf) does most poorly. About 20 target items are found in the first 1200 or so examined, and after that progress is minimal. Initially runs CLCON and gmu961, with radically different philosophies, perform about equally. But at somewhere around the 2000th item examined, the *n*-gram based method continues its slow climb, while the single term assignment method begins to level off.

The ETHFR94P method, which permits multiple interpretations of a slot in the document, climbs early to a striking advantage, and nearly dominates the other methods. However, there is a small region, corresponding to the recovery of some 4 or 5 documents near the end of the run, where the gmu961 method briefly pulls above the multiple interpretation method. The difference in this region is probably not statistically significant, but it does eliminate the possibility of a clean dominance ordering being reported from this particular trial.

Note that in preparing these figures, we have followed the TREC philosophy that one good document is as good as another, and have not considered which specific target documents were turning up at each particular point on the curve. There is no barrier in principle to doing such an analysis, which might prove interesting. In particular, if some schemes do well on some targets, and others do well on others, and *it is possible to tell them apart prior* *to retrieval*, specific assignment of schemes could produce better performance. Even if it is not possible to tell them apart, some variant of data fusion might produce results superior to those achieved by single schemes.

5. Summary

Taken together, the teams reported one or more counter-intuitive results at the conference. For example, the ETH initial screening system using 3-grams with a length based weight scheme was substantially more effective than the GMU scheme using 4-grams and cosine weighting. It is not known how much the choice of matching function contributes to this difference, and how much is due to the length of the *n*-grams. Both the CLARIT team and the GMU team found that their (different) methods of query expansion did not help at all, and in fact made performance worse.

Based on these issues, and the problems noted in the workshop papers themselves, there is still a great deal to be understood about the interaction of the diverse approaches used by the participants. The issue will remain an important one, with practical application ranging from management of corporate legacy records and technical reports to automated declassification of government documents. In this connection it would be very valuable to determine whether the “easily found” documents were characterized by relatively high frequency of the terms used in the queries. If so, it suggests that IR on confused texts will not provide the high levels of recall that are needed in certain situations involving either security or litigation. On the other hand, if the several schemes differ in the questions on which they performed well, then it is possible that some combination or fusion of several schemes will provide sufficiently high levels of performance for practical applications.

Acknowledgements

We acknowledge the enormous effort that all the teams put into tackling the Confusion Track problem, and the additional work done at NIST to formulate and refine the questions and score the results.

Notes

1. Only 49 questions were actually used in the track since question 29 had to be dropped from the evaluation. Some input files were mistakenly truncated when producing the degraded versions of the text, so all three collections were restricted to the intersection of the three sets. One of the omitted documents was the target item for question 29.
2. In January 1999, NIST issued page images and true text versions of part of the 1994 *Federal Register* as Standard Reference Database 25. While the same edition of the *Federal Register* was used in the confusion track reported here and for Database 25, the segmentation into documents is completely different. Standard Reference Database 25 is *not* simply a subset of the corpus used in the track.

References

Ballerini JP, Büchel M, Domenig R, Knaus D, Mateev B, Mittendorf E, Schäuble P, Sheridan P and Wechsler M (1997) SPIDER retrieval system at TREC-5. In: Voorhees E and Harman D (Eds.), Proceedings of the Fifth Text REtrieval Conference (TREC-5) NIST Special Publication 500-238, pp. 217–228.

Grossman DA, Lundquist C, Reichart J, Holmes D, Chowdhury A and Frieder O (1997) Using relevance feedback within the relational model for TREC-5. In: Voorhees E and Harman D (Eds.), Proceedings of the Fifth Text REtrieval Conference (TREC-5) NIST Special Publication 500-238, pp. 405–414.

Harman D (1995) The second Text REtrieval Conference (TREC-2) (special issue) *Information Processing and Management*, 31(3).

Hawking D, Thistlewaite P and Bailey P (1997) ANU/ACSys TREC-5 experiments. In: Voorhees E and Harman D (Eds.), Proceedings of the Fifth Text REtrieval Conference (TREC-5) NIST Special Publication 500-238, pp. 359–375.

Kantor PB (1997) Non-linear utility functions in information retrieval. Tech. Rep. APLab Technical Report, SCILS, Rutgers University. URL = http://scils.rutgers.edu/∼kantor/PAPERS/utility.ps

Ng KB, Loewenstern D, Basu C, Hirsh H and Kantor PB (1997) Data fusion of machine-learning methods for the TREC5 routing task (and other work). In: Voorhees E and Harman D (Eds.), Proceedings of the Fifth Text REtrieval Conference (TREC-5) NIST Special Publication 500-238, pp. 477–487.

Tong X, Zhai C, Milić-Frayling N and Evans DA (1997) OCR correction and query expansion for retrieval on OCR data—CLARIT TREC-5 confusion track report. In: Voorhees E and Harman D (Eds.), Proceedings of the Fifth Text REtrieval Conference (TREC-5) NIST Special Publication 500-238, pp. 341–345.

Voorhees E and Harman D (Eds.) (1997) Proceedings of the Fifth Text REtrieval Conference (TREC-5). NIST Special Publication 500-238.