Building a Filtering Test Collection for TREC 2002

Ian Soboroff
National Institute of
Standards and Technology
Gaithersburg, MD
ian.soboroff@nist.gov

Stephen Robertson
Microsoft Research
Cambridge, UK
ser@microsoft.com

ABSTRACT

Test collections for the filtering track in TREC have typically used either past sets of relevance judgments, or categorized collections such as Reuters Corpus Volume 1 or OHSUMED, because filtering systems need relevance judgments during the experiment for training and adaptation. For TREC 2002, we constructed an entirely new set of search topics for the Reuters Corpus for measuring filtering systems. Our method for building the topics involved multiple iterations of feedback from assessors, and fusion of results from multiple search systems using different search algorithms. We also developed a second set of "inexpensive" topics based on categories in the document collection. We found that the initial judgments made for the experiment were sufficient; subsequent pooled judging changed system rankings very little. We also found that systems performed very differently on the category topics than on the assessor-built topics.

Categories & Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – Information Filtering, Relevance Feedback; H.3.m [Information Storage and Retrieval]: Miscellaneous – Test Collections

General Terms: Experimentation, Measurement

1. INTRODUCTION

Filtering is a special kind of retrieval task where someone with a long-term information need is monitoring a stream of documents, and the system selects documents from the stream by learning a profile of the users' interests. A number of experiments have been conducted in the TREC conferences on various aspects of this process, including routing, batch filtering, and adaptive filtering. A routing system learns a static profile from training documents, and ranks all documents in the test set according to the profile. Filtering systems examine the test set one document at a time and must make a decision at each document whether to show it to the user or not. If an adaptive system decides to show a document, it receives any available relevance information for that document, which it can use to update its profile or decision thresholds.

The filtering task presents unique challenges in building test collections, because filtering systems require relevance judgments during a run for training and adaptation, whereas in TREC ad hoc collections the relevance of documents is not determined until after the experiment. Where do these relevance judgments come from? Typically, older search topics (such as from a previous year's ad hoc task) with their corresponding documents and relevance assessments were used for training, and a new document collection was provided for testing. However, obtaining new test data, as well as finding collections appropriate to the task, has been a challenge for the track since the beginning.

1.1 TREC 1-3: Routing

In TREC-1, topics 1-50 were developed for routing and 51-100 for ad hoc. The first two TREC CDs comprised the collections. For routing, a small number of judgments were made on CD 1 for topics 1-50 and released as training data. Because of the incompleteness of these training judgments, the TREC-1 results should be seen as preliminary. Participants constructed routing profiles from this data and tested their systems on the documents on the second TREC CD. In TREC-2 and 3, routing was able to make use of the past year's ad hoc topics, with their judgments serving as training data. It was intended that routing would use new collections each year for test data, and new documents were accordingly promised for TREC-3, but in the end were not delivered. As a result, CD 3 was reused as the test set. Despite these topics already being judged for CD 3, NIST pooled the results and made additional judgments from the TREC-3 routing runs [2].

1.2 TREC 4-6: Early Filtering

After TREC-3, a strong argument was made that a more realistic filtering task should be developed in addition to routing. David Lewis designed the structure of the filtering track for TREC 4-6, which was cast as a binary classification task rather than a ranking task. This necessitated several methodological departures from the standard TREC evaluations, namely the use of set-based measures and statistical sampling for pooling [8].

Because new test data was hard to come by, the choice of topics was driven by what data could be had. For TREC-4, topics were selected from past years and comprised two subsets. Half of the topics were selected on the basis of hav-

Copyright 2002 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
SIGIR, July 28–August 1, 2003, Toronto, Canada.
Copyright 2003 ACM 1-58113-646-3/03/0007 ...$5.00.

Year	Tasks	Training	Test	Topics	Judged?
TREC-1	R	CD 1	CD 2	1-50 (pilot)	yes
TREC-2	R	CD 1,2	CD 3	51-100 (T1 ad hoc)	yes
TREC-3	R	CD 1,2	CD 3	101-150 (T2 ad hoc)	yes
TREC-4	R,FI	CD 1,2,3	FR 94, Ziff (CD 3), "net trash"	50 old	yes
TREC-5	R,FI	AP from CD 1-3	FBIS	49 old	yes
TREC-6	R,FI,A	FBIS CD 5	FBIS CD 6	38 old, 9 new	yes
TREC-7	R,B,A	AP 88 (R,B)	AP 88 (A), 89-90 (R,B,A)	1-50	yes
TREC-8	R,B,A	FT 92	FT 93,94	351-400 (T7 ad hoc)	yes
TREC-9	R,B,A	OHSUMED (1987)	OHSUMED (remainder)	63 OHSU topics 500 and 4904 MeSH labels	no
TREC 2001	R,B,A	RCV1 (Aug 96)	RCV1 (remainder)	84 Reuters categories	no
TREC 2002	R,B,A	RCV1 (Aug-Sep 96)	RCV1 (remainder)	50 new topics 50 category intersections	yes

Table 1: TREC routing and filtering tasks, their collections and search topics. For tasks, R=routing, FI=filtering (as defined in TREC-4), A=adaptive filtering, B=batch filtering. The "Judged" column indicates if new judgments were made for that topic set and collection of test documents.

ing relevant documents in the existing FR collection, and a new set of Federal Register documents (FR94) was assembled. The other half of the topics were chosen so as to make a "computers" subcollection; training data came from the Ziff collections on disks 1-2, and the test data was Ziff from disk 3 and an assortment of USENET articles and issues of the IR Digest and Virtual Worlds mailing lists. The topics were interspersed together so that systems did not necessarily know what kind of topic they were processing [20]. For TREC-5 and 6, new documents became available from the Foreign Broadcast Information Service (FBIS). In TREC-5, all 50 topics[1] were chosen by relevant document occurrence in AP, assuming that FBIS documents would resemble AP. This assumption turned out not to be valid; several topics had few or no relevant documents found in the test set, while others had a very large number [9]. For TREC-6, more FBIS documents were acquired and as such a better training/test match could be made. There were 47 topics used: 38 TREC-5 filtering topics with at least six relevant FBIS documents, five additional old topics (62, 128, 148, 180, and 282), and four brand new topics (numbered 10001-10004). For the nine new topics, incomplete training judgments were made by assessing the top 100 FBIS documents retrieved by PRISE.

In TREC-6 adaptive filtering began as a "pilot" task although only one group (UMass) attempted it. An adaptive filtering run began with only the topic statement, ran over the entire FBIS collection as a test, and could adapt based on an available judgment for a document retrieved by the system. Participants were allowed to use any other non-topic-specific training data they wanted, such as IDFs from TREC collections outside of FBIS, or thesauri. Because of the incomplete judgments for the nine new topics, the task only used the 38 TREC-5 topics [4].

1.3 Modern Filtering

In TREC-7, routing was folded into the filtering track, while the definition of "filtering" was refined into two tasks: batch filtering and adaptive filtering. Topics 1-50 and the AP collections on CDs 1-3 were used as data. These topics had the limited relevance judgments for 1988 used as training for routing in TREC-1, better judgments for 1989 (the TREC-1 routing test data), and no judgments at all for 1990. Thus, routing and batch filtering used 1988-9 for training and 1990 for test data; adaptive filtering started with the (very long) topic statements and had to filter the whole AP collection [5]. It is likely that the quality of the TREC-1 relevance judgments on AP88 and 89, combined with the very large time gap between when the TREC-1 judgments on AP88-9 and TREC-7 judgments on AP90 were made, had a large effect on the TREC-7 filtering results. To overcome this, TREC-8 used the TREC-7 ad hoc topics and document collection. Although no new judgments were initially planned, NIST in the end agreed to do limited pooling [6]. For TREC-9, no assessment resources were available for filtering, so the track used the OHSUMED collection [3], which consists of nearly 350,000 documents labeled with MeSH categories, as well as 101 search topics with relevance judgments on a three-point scale. Adaptive filtering systems were given the topic statements along with two "definitely relevant" training documents.

Two other "topic" sets were used: 4904 MeSH headings having four or more definitely relevant documents in 1987 and at least one document in the final year, and a subset of 500 MeSH headings sampled from this larger set. Adaptive systems were given the heading name itself and its scope note (about the length of a TREC description field), along with four relevant training documents. Apart from dramatically increasing the scale of TREC filtering experiments, these topic sets were the first use of categories as filtering user needs [12].

In TREC 2001, the filtering task decided to use the new Reuters Corpus Volume 1 (RCV1) as a document collection.[2] RCV1 contains about 800,000 Reuters news articles dating from August 1996 to 1997. Each article is classified by hand into topic, country, and/or industry categories [10][15]. At the time, there were no search topics for RCV1, and so for the TREC 2001 filtering tasks the track coordinators selected 84 topic categories to serve as filtering topics. These particular topics were among those containing

[1] One topic was inadvertently dropped from the routing and filtering evaluation in TREC-5, so there are 49 usable topics from this collection. Twelve of the topics were also used in TREC-4.

[2] `http://about.reuters.com/researchandstandards/corpus/index.asp/`

the fewest relevant documents, but even so many categories had several thousand [13].

2. DEVELOPING FILTERING TOPICS

We can see that the TREC filtering tasks have used a wide variety of document collections, search topics, and approaches for gathering training and test judgments. These collections have not been driven by the task but rather by the (limited) availability of document collections and resources for making new relevance judgments. In some years this resulted in a poor match between training and test collections. Furthermore when TREC topics were used, training data consisted of small sets of judgments made on the results of a single search system, or past judgments made for those topics at least one year earlier. There was always a time gap of at least a year when new relevance judgments were made for a test set. Chronological consistency was best maintained when using categories as topics, but categories were less desirable than true search topics.

Consequently it was decided for TREC 2002 to have the NIST assessors develop a new set of topics specifically for the filtering track. The goals of this process were to build a topic set for RCV1 of comparable quality to a TREC ad hoc collection, but making as many of the judgments during topic development as possible, both to provide adaptive systems with full data and to avoid problems of assessment "drift" due to time lag.

Creating TREC topics is an expensive process even for ad hoc search tasks, and so the track also decided to experiment with intersections of Reuters categories as a cheaper way to build realistic search topics. Filtering track participants were given 100 search topics, of which 50 were composed by human searchers and 50 by intersecting two Reuters topic categories, and were asked to run their systems on all the topics together. Our hope was that system performance on the intersection topics would predict performance on the assessor topics, but this turned out not to be the case.

2.1 Assessor Topics

For TREC ad hoc collections, assessors develop topics by exploring the collection using a retrieval system and making minimal relevance assessments to try to determine how easy or hard the topic will turn out to be. Final relevance judgments are made by pooling participants' search results after they have submitted their TREC runs. For the filtering track, we have to provide the relevance judgments to the participants up front. Thus, the problem we faced was how to determine the relevant documents for the topics, without exhaustively searching the collection, or releasing the topics to participants. Furthermore, we didn't want to use more assessor time than would typically be used to do relevance assessing for an ad hoc task. In the end we did make additional relevance judgments from pooled runs in order to verify that the test-set judgments were reasonably complete.

To allow the assessors to do more exhaustive searching, we augmented the topic development process with multiple iterations of relevance feedback. After their initial searches were complete and the topic definition established, we asked the assessors to judge the top 100 documents as retrieved by PRISE for their final query for the topic. These judgments were used as relevance feedback, and on the next day the assessors received a new set of 100 documents to judge. The feedback cycles continued until no more relevant documents were found, or for a maximum of five days. Due to some glitches in the system, some topics were judged for more than five days.

An important concern was the quality and diversity of the documents being judged. The quality of a pool depends on the number and variety of systems and searchers contributing to it. For example, manual ad hoc runs often contribute a disproportionately large fraction of relevant documents found by no other run [21]. One reason pooling works in TREC to create test collections usable outside the conference itself is that many different systems are contributing to the pools. It was clear to us that if we only used PRISE for searching, we would very likely miss many documents that other systems would retrieve. To avoid this, we fed the feedback results to four search systems using seven different retrieval strategies:

PRISE PRISE is NIST's search system and is used to develop topics for many TREC tracks. This is actually an internal development version and has not been released publicly. It is a traditional IR system and supports many retrieval and feedback models. For this task, we used BM25 weights for terms, Robertson-Sparck Jones reweighting for feedback terms, and selection of the top ten reweighted terms for use in feedback retrieval. Feedback was based on the topic statement and all relevant documents found in previous iterations.

SMART Cornell's SMART system, with some minor modifications not used here. We used ltc.ntc weighting, and Rocchio feedback with default settings. Feedback input was all relevant documents found so far, and irrelevant documents from the most recent iteration.

YARI YARI is a language modeling system written by Victor Lavrenko of the University of Massachusetts. The specific language modeling approach is described in [7]. In our setup, YARI built its model using all prior positive feedback. We used uniform weighting for documents and linear smoothing.

BOW Andrew McCallum's Bag Of Words toolkit is designed for experimenting with text classification algorithms [11]. We used the Naive-Bayes and SVM algorithms from BOW in a multiclass classification setup where each topic was a separate class. For SVM, we used two different input sets, one with just the feedback data for each topic, and one including a sample of around 3000 documents as an "unlabeled" class for transduction. We also used BOW's k-nearest-neighbors algorithm, but discarded it after two iterations for speed reasons; thus, not all topics have documents from the kNN classifier. The classifiers were given all topics running in the current feedback iteration. Documents were ranked by their classification score for each topic being run that day.

We chose systems that represented a variety of approaches that might be used in the TREC filtering track, came with source code, and could be scripted easily into our feedback process.

Each system was configured to return the top 100 documents for each topic based on the latest feedback as described above. These result sets were then merged using the

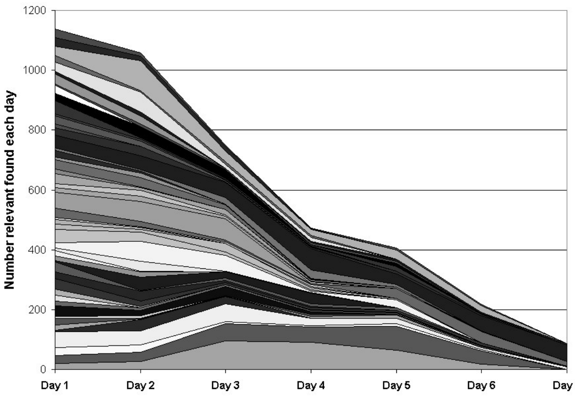

Figure 1: New relevant documents found on each day a topic was evaluated.

CombMNZ fusion algorithm [1], and the top-ranked 100 documents were chosen as the pool to be judged the next day. Unlike TREC pools, these pools were judged in descending order of CombMNZ score.

The pools were judged by the assessor who originally composed the topic. On each day, an assessor would judge pools for three to five topics, and at the end of the day the assessors' judgments were collected and fed back to the above systems. If no new relevant documents were found on that day for a given topic, that topic was "retired" and no more judgments were made for it. Also, if a topic had been judged for five days, we halted that topic. After retiring a topic from feedback, we gave a new topic to that assessor to judge until all topics were judged or we ran out of time.

Figure 1 illustrates the number of relevant documents discovered in each feedback iteration for the fifty topics used in TREC 2002. "Day 1" is the final PRISE search before the feedback process was started. Several different patterns are evident. Some topics displayed the expected behavior, with most relevant documents found in the first couple of iterations followed by a dropoff. However, others only "blossomed" after one or two iterations, and others kept turning up new relevant documents even after a week of searching.

If we consider the full set of documents judged for a topic as an aggregate pool, the average number of documents judged was 433 with 82 relevant (19%). In comparison, the average TREC-8 ad hoc pool contained 1736 documents, of which 94 (5%) were judged relevant [21]. So by using relevance feedback to construct multiple pools in sequence, we were able to find comparable numbers of relevant documents in an overall smaller pool. This does not suggest that the assessors were more lenient judges of these smaller pools, since they are experienced TREC assessors and the standards of relevance were the same as for TREC ad hoc relevance, and also the general distribution of relevant documents across topics was similar to an ad hoc collection of similar size.

Table 2 illustrates how each system contributed to the pool (considering all the pools for each topic together). The first column shows the percentage of the pool contributed by each system; for example, 33.6% of the documents in the pool were contributed by PRISE. The numbers add up to more than 100% because of overlap among the systems; on average, 55.6% of the documents in each pool were contributed by more than one system. This is a much higher degree of overlap than is seen in TREC pools [21]. The second column gives the percentage of unique pool documents contributed by each system. We can see that PRISE and SMART contributed the most unique documents to the pools. Even though BOW-SVM had the largest average contribution, the BOW systems probably had fewer unique documents in the pool because of similarity to each other.

	% Judged		% Relevant	
System	Total	Unique	Total	Unique
PRISE	33.6%	9.2%	40.2%	4.2%
SMART	40.5%	12.2%	40.7%	3.0%
YARI	22.3%	4.2%	41.2%	13.6%
BOW-NB	28.0%	5.0%	32.4%	0.9%
BOW-SVM	42.6%	6.8%	43.7%	2.6%
BOW-SVM-trans	19.2%	5.8%	10.0%	0.0%
BOW-kNN	2.4%	1.1%	0.4%	0.0%

Table 2: Contributions of each system to the sets of judged and relevant documents.

The third and fourth columns show the percentage of relevant documents contributed by each system, and also unique relevant documents. YARI's percentage of unique relevant documents is actually due to a bug in the YARI runs: a coding error caused each document to be included twice at rank n and $n + 1$. Thus, YARI's overall contributions to the pool are understated because their runs were effectively only examined to depth 50. However, each of those documents received an undue CombMNZ score because YARI "recommended" them each twice.

These numbers hide a lot of variation in the contributions to each topic and on each day. Not only did the systems vary quite a bit in how they contributed to each topic, but also during each feedback cycle some systems made more of a contribution to the pool than others. Figure 2 illustrates relevant contributions by each system to the pools over the multiple feedback cycles. While there is a general decreasing trend as time progresses, different systems found new groups of relevant documents on different topics.

Ninety-eight topics in all were judged through this process, and we selected fifty to use for TREC 2002. These topics had between 226 and 678 judgments with between 12 and 351 relevant documents each. The selection process was guided by the training requirements for adaptive filtering. Furthermore, since some topics are "bursty" in nature and some are more periodic, we wanted topics with a variety of patterns of relevant documents across the collection. During the feedback process, we tracked topics to see where the judged documents are occurring within the collection. In the end, we were able to select a training period cutoff date in the collection (9/30/96) that included at least three positive training examples per topic.

The topic creation process took a total of four weeks. If there had been no bugs and we had also better identified topics to "retire", we probably could have saved a week. This is roughly equivalent to the total amount of time needed to develop and judge a set of TREC ad hoc topics. Note however that TREC topic development and assessment are typically separated by 4-6 months, during which time assessors' notion of relevance can change.

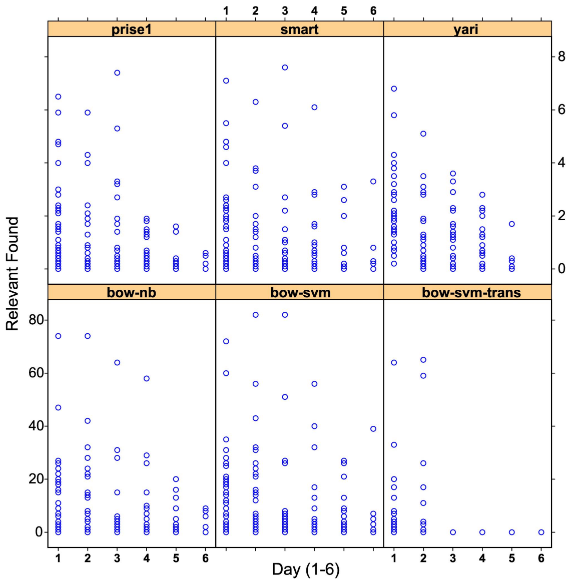

Figure 2: Relevant documents contributed to the pools on each day, per system. BOW-kNN has been omitted.

2.2 Intersection topics

Because RCV1 documents are already labeled with category codes by Reuters, we wished to see if some use could be made of these. Based on a suggestion at TREC 2001, we decided to explore using intersections of categories as topics. A category intersection is a pair of categories where the relevant documents are those that belong to both categories. Category intersections have the same or fewer relevant documents than either parent category. Furthermore, the intersection of two categories might be considered as a specialized interest in each of those categories, and thus more similar to TREC topics than categories alone.

The 126 Reuters topic categories are divided into five groups and have a wide range of scope. Several of these groups have hierarchy, and the "Government/Social" group has general categories as well as a hierarchy which overlaps somewhat with the general categories. The Reuters industry categories are an even richer hierarchy, but have coding problems which would have complicated usage of the corpus [10]. We did not make use of the region codes.

Of the full set of topic categories, 99 are represented in the corpus. Most documents are labeled with more than one topic category. Of all possible pairs of topic categories, we found 3435 represented in the collection (that is, for a pair there exists at least one document labeled with the two categories and possibly others as well). 1514 of those pairs had three or more documents in the portion of the collection designated for training; this was our starting set of candidate topic intersections. We also briefly looked at category triples but did not use them.

We selected 50 category pairs that seemed (from the category names) to be meaningful as search topics, and to have an overall number of relevant documents (documents labeled with both categories) within the range of the assessor-built topics. The topics were selected after the assessors had completed their topics but before the topics were used in TREC, so the assessor-built topics actually have more relevant documents because of additional assessments made from participants runs, described below.

Since the assessor-built topics also have documents labeled as irrelevant, we created irrelevant sets for the intersections by selecting a random sample of documents belonging to one of the two categories but not their intersection. This ensured that the irrelevant documents were not arbitrary but represented near-misses as might be selected for a pool and marked irrelevant by an assessor.

TREC-style topic statements for the intersection topics were created mechanically from a set of category descriptions obtained from Reuters. The descriptions were extended phrases such as "stories relating to deaths of famous persons" for the category *GOBIT: Obituaries.* The 'title' and 'description' sections were made from the category names, and the 'narrative' was pasted together from the Reuters descriptions. The descriptions were cleaned up minimally, by hand, for grammatical consistency. For example, for topic R200, "Management, Obituaries", the narrative reads, "Relevant documents discuss all management issues and stories relating to deaths of famous persons." Clearly, these descriptions are not as good as manually-created ones, but filtering systems rely more on training documents than the topic description, and these allow systems to process the intersection topics in the same way they do regular ones.

3. TREC 2002 EXPERIENCES

As discussed above, the Filtering track in TREC 2002 had three main tasks: adaptive filtering, batch filtering and routing. In addition, two measures of performance were used for filtering (utility and FBeta), which meant that there were in effect five distinct tasks.

3.1 Tasks

The TREC 2002 model of adaptive filtering task follows the general pattern discussed. We assume the user arrives with a small number of known positive examples (relevant documents). For each topic, the last three relevant documents in the training set were made available to the participants for this purpose; no other relevance judgments from the training set could be used. However, statistics such as term frequencies could be taken from the full training set. Subsequently, once a document is retrieved, the relevance assessment (when one exists) is immediately made available to the system. Judgments for unretrieved documents are never revealed to the system. Once the system makes a decision about whether or not to retrieve a document, that decision is final. No back-tracking or temporary caching of documents is allowed.

Again as discussed, in batch filtering, all the training set documents and all relevance judgments on that set are available in advance. Once the system is trained, the test set is processed in its entirety. For each topic, the system returns a single retrieved set. For routing, the training data is the same as for batch filtering, but in this case systems return a ranked list of the top 1000 retrieved documents from the test set.

3.2 Measures

FBeta is a variant on the F1 measure commonly used in text categorization, and originally proposed by van Rijsbergen [16]. The constant β is set to 0.5, corresponding to an emphasis on precision. The measure is averaged over topics. The utility measure is a linear utility, with a credit of 2 units for a relevant document retrieved and a debit of 1 unit for a non-relevant retrieved. This measure is scaled before being averaged over topics; the form of normalization used means that a system which retrieves nothing gets a certain positive score, which we treat as a baseline performance level (indicated in the figures below). Full details are given in [14]. These two measures are used for the adaptive and batch filtering tasks; each submitted run was declared to be optimized for one of these measures. For the routing task, mean average precision was used.

3.3 Results

For reasons which will become apparent, these results are separated into Assessor and Intersection topics.

Assessor topics. The graph on the left of Figure 3 shows the utility results for the assessor topics and all submissions to the adaptive filtering track. (Note that some of the runs were not optimized for this measure.) The systems are ranked by the mean value across topics of the scaled utility measure, T11SU. The horizontal line inside a run's box is the median topic score, the box shows interquartile distance, the whiskers extend to the furthest topic within 1.5 times the interquartile distance, and the circles are outliers. The horizontal line across the whole graph shows the performance that would obtain if a system were to retrieve nothing. Unlike in some earlier TREC filtering experiments, a substantial number of systems performed well over this level. In fact for quite a number of systems, 75% of the topics were over this level.

There is a certain amount of bunching among the best-performing systems – a characteristic which is generally taken at TREC to mean some degree of maturity among the competing systems for this task. However, there are clearly several systems with scope for improvement. The performance measurement appears to be doing a reasonable job of distinguishing between systems.

Intersection topics. The graph on the right of Figure 3 shows the equivalent results from the intersection topics. The story told by this graph is very different. First, the absolute levels of performance are terrible – no system did better on average than our hypothetical baseline which retrieves nothing. Indeed, the best upper quartile is exactly on the baseline – no system succeeded in getting even 25% of the intersection topics above this level.

One might be tempted to think that the intersection topics are simply much harder, but nevertheless represent a realistic task. However, the magnitude of the difference makes this explanation difficult to maintain. In fact it seems that the only solution open to the systems was to shut down most topics as soon as possible, to cut their losses.

Some ongoing analysis of individual topics is suggesting some possible reasons for this discrepancy. But we are forced to the conclusion, at least for the present, that the intersection topics do not constitute a useful set of topics for filtering experiments. Again, these impressions are not confined to the adaptive filtering utility results – the results for batch filtering and routing are equally bad, despite the additional training material available.

	T11U	T11F
Adaptive	0.969	0.936
Batch	0.996	0.983
Routing	0.912 (MAP)	

Table 3: Correlation of the official TREC results to a system ranking measured using the first-round relevance judgments only.

New judgments. Although more than 21,000 relevance judgments were made during topic creation and released with the topics, we were concerned that participants would still find more relevant documents. In order to make sure systems were measured fairly, NIST pooled participants' runs and judged any previously unjudged documents in the pool. Pooling was done as follows. Each participating group (who may have submitted up to four adaptive, two batch, and two routing runs) was allotted a fixed budget of documents to be pooled from their runs. If the group had any routing runs, we added unjudged documents from the top 100 ranks to the pool. If the group also had filtering runs, at most half the budget was expended on routing documents. We then merged all batch and adaptive filtering runs from that group and took a random sample of documents from the combined runs to fill out the pool budget. In all, another 42,000 documents were judged during this second round of assessment.

Figure 4 shows the numbers of relevant documents found for each topic in the first and second rounds of judging. Note that overall the topics have between 9 and 599 relevant documents apiece, much fewer than the TREC 2001 categories and closer to TREC ad hoc scale. For most topics only a few new relevant documents were found in the second round (median = 8.5), but seven topics had more than fifty new. Four of these topics had more than twenty new relevant documents found in their last feedback iteration during the creation phase. Although our pooling process is radically different, these findings agree with Harman's analysis of the TREC-3 relevance judgments [2], as well as those of Zobel [22] that the "largest" topics (those with the most relevant documents) tend to yield even more relevant documents upon further searching. We have seen that such topics tend to have a greater number of relevant documents found in the last round of judging. In retrospect it probably would have been a good idea to discard these topics.

Another important factor is that five topics were judged by a different assessor in the second round than the one who had created it. Although as a general rule assessors always judges their own topics, due to time constraints we were forced to move these topics to different assessors. In these cases, the assessor was shown all of the relevant documents found in the first round as orientation to the topic. Four of these "moved" topics were also topics with more than fifty new relevant found, suggesting that these topics were not judged as well as the others.

Despite all the additional judgments and newfound relevant documents, the performance of the systems participating in the TREC 2002 filtering track was largely unchanged when measured with the full set of relevance judgments. Kendall's tau correlations (shown in Table 3) between the official TREC scores and measurements made using the first round of judgments only show that the rankings are virtu-

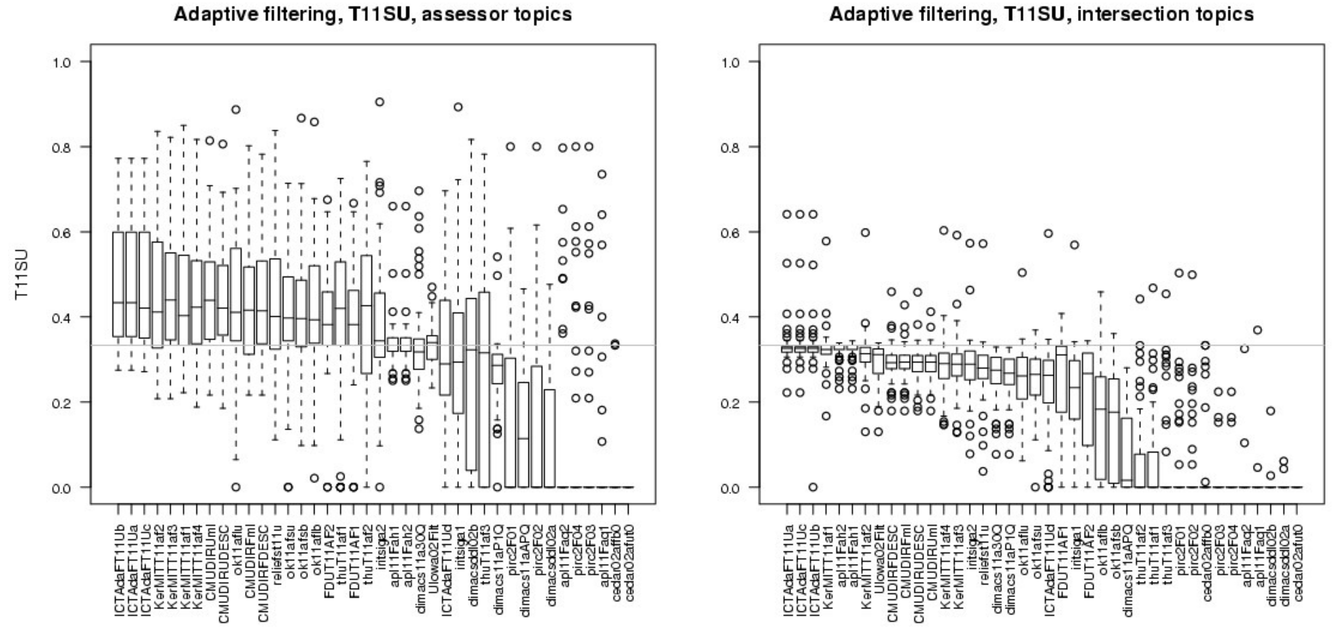

Figure 3: Adaptive filtering – Utility

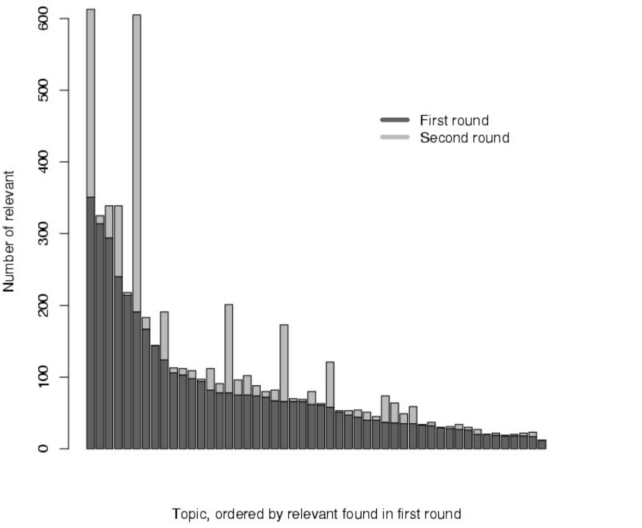

Figure 4: Relevant documents found in the first and second rounds of judging.

Run	T11U (utility)			T11F (Fbeta)		
	TREC	Rerun	diff	TREC	Rerun	diff
1a	0.4350	0.4397	0.0047	0.4214	0.4192	-0.0022
1b	0.4088	0.4198	0.0110	0.4033	0.4118	0.0085
1c	0.4057	0.4199	0.0142	0.3959	0.4137	0.0178
1d	0.4056	0.4093	0.0037	0.3939	0.4035	0.0096
2a	0.4751	0.4788	0.0037	0.4272	0.4215	-0.0057
2b	0.4753	0.4784	0.0031	0.4278	0.4205	-0.0073
2c	0.4706	0.4747	0.0041	0.4225	0.4183	-0.0042

Table 4: Two adaptive systems performed slightly differently when using the final set of relevance judgments for adaptation during their run.

ally identical. This means that the evaluation did not penalize systems because of relevant retrieved documents which were not judged. For adaptive systems, the story is a bit more complicated, since we don't know if the systems would have adapted differently because there were more judgments available. We asked participants to do adaptive runs using the final relevance judgments as input, but otherwise keeping their systems identical to what was submitted to TREC. Two groups were able to provide a total of seven runs. The results, shown in Table 4, are not conclusive since they only come from two groups, but seem to indicate that adaptive systems would not have performed very differently on the main measures if they had been given the additional relevance judgments. The true effect depends on how systems adapt when they retrieve an unjudged document as opposed to a judged one. A more detailed look at the results suggests that systems achieved noticeably higher recall in the new runs, but with a balancing loss of precision.

4. CONCLUSIONS

We believe that the 50 new assessor topics, together with the relevance judgments on the Reuters RCV1 corpus, constitute a good and valuable addition to the resource represented by the collective TREC collections.

By and large, the method of generating relevance judgments by successive feedback iterations on four different systems has proved valid and useful. The resulting judgments are likely to be sufficient for both feedback and evaluation purposes. The discovery of additional relevant documents in the second round does not appear to invalidate this conclusion; however, if we were running the experiment again, we might be inclined to reject topics which are continuing to generate significant numbers in the final feedback iteration.

Until we better understand the problems of the intersection topic set, this method of construction cannot be recommended.

5. REFERENCES

[1] Edward A. Fox and Joseph A. Shaw. Combination of Multiple Searches. In D. K. Harman, editor, *Proc. of the Second Text REtrieval Conference (TREC-2)*, NIST SP 500-215, pp. 243–252. National Institute of Standards and Technology, Gaithersburg, MD, Nov 1993.

[2] Donna Harman. Overview of the Third Text REtrieval Conference (TREC-3). In Donna K. Harman, editor, *Proc. of the Third Text REtrieval Conference (TREC-3)*, NIST SP 500-225, pp. 1–20. Gaithersburg, MD, Nov 1994.

[3] William Hersh, Chris Buckley, T. J. Leone, and David Hickam. OHSUMED: an Interactive Retrieval Evaluation and new Large Test Collection for Research. In *Proc. of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 192–201. Dublin, Ireland, Jul 1994.

[4] David A. Hull. The TREC-6 Filtering Track: Description and Analysis. In E. M. Voorhees and D. K. Harman, editors, *Proc. of the Sixth Text REtrieval Conference (TREC-6)*, NIST SP 500-240. National Institute of Standards and Technology, Gaithersburg, MD, Nov 1998.

[5] David A. Hull. The TREC-7 Filtering Track: Description and Analysis. In Voorhees and Harman [18], pp. 33–56.

[6] David A. Hull and Stephen Robertson. The TREC-8 Filtering Track Final Report. In Voorhees and Harman [19], pp. 35–56.

[7] Victor Lavrenko and W. Bruce Croft. Relevance-Based Language Models. In W. Bruce Croft, David J. Harper, Donald H. Kraft, and Justin Zobel, editors, *Proc. of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2001)*, pp. 120–127. ACM Press, New Orleans, LA, Sep 2001.

[8] David D. Lewis. The TREC-4 Filtering Track. In Donna K. Harman, editor, *Proc. of the Fourth Text REtrieval Conference (TREC-4)*, NIST SP 500-236, pp. 165–180. Gaithersburg, MD, Nov 1995.

[9] David D. Lewis. The TREC-5 Filtering Track. In Voorhees and Harman [17], pp. 75–96.

[10] David D. Lewis, Yiming Yang, Tony Rose, and Fan Li. RCV1: A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research*, 2002. To appear.

[11] Andrew K. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering, 1996. `http://www.cs.cmu.edu/~mccallum/bow`.

[12] Stephen Robertson and David A. Hull. The TREC-9 Filtering Track Final Report. In E. M. Voorhees and D. K. Harman, editors, *Proc. of the Ninth Text REtrieval Conference (TREC-9)*, NIST SP 500-249, pp. 29–40. National Institute of Standards and Technology, Gaithersburg, MD, Nov 2000.

[13] Stephen Robertson and Ian Soboroff. The TREC 2001 Filtering Track Final Report. In E. M. Voorhees and D. K. Harman, editors, *Proc. of the Tenth Text REtrieval Conference (TREC 2001)*, NIST SP 500-250, pp. 26–37. Gaithersburg, MD, Nov 2001.

[14] Stephen Robertson and Ian Soboroff. The TREC 2002 Filtering Track Final Report. In E. M. Voorhees and D. K. Harman, editors, *Proc. of the Eleventh Text REtrieval Conference (TREC 2002)*, NIST SP 500-xxx. Gaithersburg, MD, Nov 2002. To appear.

[15] T. G. Rose, M. Stevenson, and M. Whitehead. The Reuters Corpus Volume 1 - from Yesterday's News to Tomorrow's Language Resources. In *In Proc. of the Third International Conference on Language Resources and Evaluation*. Las Palmas de Gran Canaria, May 2002.

[16] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

[17] E. M. Voorhees and D. K. Harman, editors. *Proc. of the Fifth Text REtrieval Conference (TREC-5)*, NIST SP 500-238. Gaithersburg, MD, Nov 1996.

[18] E. M. Voorhees and D. K. Harman, editors. *Proc. of the Seventh Text REtrieval Conference (TREC-7)*, NIST SP 500-242. National Institute of Standards and Technology, Gaithersburg, MD, Nov 1998.

[19] E. M. Voorhees and D. K. Harman, editors. *Proc. of the Eighth Text REtrieval Conference (TREC-8)*, NIST SP 500-246. National Institute of Standards and Technology, Gaithersburg, MD, Nov 1999.

[20] Ellen M. Voorhees and Donna Harman. Overview of the Fifth Text REtrieval Conference (TREC-5). In Voorhees and Harman [17], pp. 1–28.

[21] Ellen M. Voorhees and Donna Harman. Overview of the Eighth Text REtrieval Conference (TREC-8). In Voorhees and Harman [19], pp. 1–24.

[22] Justin Zobel. How Reliable are the Results of Large-Scale Retrieval Experiments? In W. Bruce Croft, Alistair Moffat, C. J. van Rijsbergen, Ross Wilkinson, and Justin Zobel, editors, *Proc. of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '98)*, pp. 307–314. ACM Press, Melbourne, Australia, Aug 1998.