Evaluation Over Thousands of Queries

Ben Carterette*, Virgil Pavlu†, Evangelos Kanoulas†, Javed A. Aslam†, and James Allan*

*Center for Intelligent Information Retrieval
University of Massachusetts Amherst
140 Governors Drive
Amherst, MA 01003

†College of Computer and Information Science
Northeastern University
360 Huntington Ave, #202 WVH
Boston, MA 02115

ABSTRACT

Information retrieval evaluation has typically been performed over several dozen queries, each judged to near-completeness. There has been a great deal of recent work on evaluation over much smaller judgment sets: how to select the best set of documents to judge and how to estimate evaluation measures when few judgments are available. In light of this, it should be possible to evaluate over many more queries without much more total judging effort. The Million Query Track at TREC 2007 used two document selection algorithms to acquire relevance judgments for more than 1,800 queries. We present results of the track, along with deeper analysis: investigating tradeoffs between the number of queries and number of judgments shows that, up to a point, evaluation over more queries with fewer judgments is more cost-effective and as reliable as fewer queries with more judgments. Total assessor effort can be reduced by 95% with no appreciable increase in evaluation errors.

Categories and Subject Descriptors: H.3 Information Storage and Retrieval; H.3.4 Systems and Software: Performance Evaluation

General Terms: Experimentation, Measurement, Algorithms

Keywords: information retrieval, evaluation, test collections, million query track

1. INTRODUCTION

Over the past 40 years, Information Retrieval research has progressed against a background of ever-increasing corpus size. From the 1,400 abstracts in the Cranfield collection, the first portable test collection, to the 3,200 abstracts of the Communications of the ACM (CACM), to the 348,000 Medline abstracts (OHSUMED), to the first TREC collections of millions of documents, to the web—billions of HTML and other documents—IR research has had to address larger and more diverse corpora.

As corpora grow, the assessor effort needed to construct test collections grows in tandem. Sparck Jones and van Rijsbergen introduced the *pooling method* [17] to deal with the problem of acquiring judgments. Rather than judge every document to every query, the documents ranked by actual retrieval systems could be pooled and judged, thus focusing judging effort on those documents least likely to be nonrelevant. The pooling method has been successful for decades. However, recent work suggests that the growth in the size of corpora is outpacing even the ability of pooling to find and judge enough documents [9]. Rather than trying to keep up simply by judging more documents, there has been interest in focusing judging effort even better and make smarter inferences when few judgments are available.

With fewer judgments available, estimates of evaluation measures will have higher variance. One way to cope with this is to evaluate over more queries. Web search engines, for instance, typically judge very shallow pools for thousands of queries. For the recall-based measures that are ubiquitous in retrieval research, such a shallow pool is not enough.

In this work we describe an evaluation over a corpus of 25 million documents and 10,000 queries, the Million Query Track that ran for the first time at the Text REtrieval Conference (TREC) in 2007 [1]. Using two recent method for selecting documents and evaluating over small collections, we achieve results very similar to an evaluation using 149 queries judged with more depth, with 62% of the assessor effort but 11 times as many queries. But this only establishes an upper bound. An evaluation with high rank correlation can be achieved with 3% of the effort over only 1.34 times as many queries, and using analysis of variance and stability studies, we show that the amount of effort needed to establish that differences between systems are not simply due to random variation in scores is at most 5% of the effort over 1.14 times as many queries.

We begin in Section 2 by describing the methods used to select documents and evaluate for the Million Query Track. In Section 3 we describe the setup of the experiment and the relevance judgments collected. Section 4 presents initial results, comparing the two methods to the baseline set of 149 queries. In Section 5, we approach the question of the number of queries and judgments for each query that is needed to evaluate with minimum effort, and also explore the reusability of such a small test collection.

2. INCOMPLETE TEST COLLECTIONS

The two methods we used differ by the aspect of evaluation that they attack. The Minimal Test Collection (MTC) algorithm is designed to induce rankings of systems by iden-

SIGIR'08, July 20–24, 2008, Singapore
Copyright 2008 ACM 978-1-60558-164-4/08/07 ...$5.00.

tifying differences between them [11], without regard to the values of measures. StatAP is a sampling method designed to produce unbiased, minimum-variance estimates of average precision [2]. Both methods are designed to evaluate systems by average precision (AP), which is the official evaluation measure of TREC *ad hoc* and *ad hoc*-like tracks. AP is the average of the precision values at ranks at which relevant documents were retrieved. Letting x_i represent the relevance of the document at rank i, precision at rank k is $prec@k = \frac{1}{k}\sum_{i=1}^{k} x_i$. Average precision is then

$$AP = \frac{1}{R}\sum_{i=1}^{n} x_i prec@i = \frac{1}{R}\sum_{i=1}^{n}\sum_{j=1}^{i} \tfrac{1}{i} x_i x_j$$

where n is the number of documents and R is the number of relevant documents. Average precision is a common and well-understood measure in IR research.

2.1 Minimal Test Collections (MTC)

MTC is a greedy on-line algorithm for selecting documents to be judged. Given a particular evaluation measure and any extant relevance judgments, it weighs documents by how informative they are likely to be in determining whether there is a difference in the measure between two systems. The highest-weight document is presented to an assessor for judging; the judgment is used to update document weights.

Define the difference in AP as $\Delta AP = AP_1 - AP_2$. Let x_i be the relevance of document i, where the index i is arbitrary, unrelated to the rank of the document. We can then express ΔAP in closed form as

$$\Delta AP = \frac{1}{\sum_{i=1}^{n} x_i}\sum_{i=1}^{n}\sum_{j=i}^{n} c_{ij} x_i x_j$$

where c_{ij} is a constant depending on the ranks of documents i and j in the two systems:

$$c_{ij} = \tfrac{1}{\max\{rank_1(i), rank_1(j)\}} - \tfrac{1}{\max\{rank_2(i), rank_2(j)\}}.$$

If $\Delta AP > 0$ over our judgments (assuming unjudged documents to be nonrelevant), then making the worst case assumptions that every unjudged document that will decrease ΔAP if relevant will be judged relevant, and that every unjudged document that would increase ΔAP if relevant will be judged nonrelevant, we can determine whether there is *any* set of judgments that will result in the sign of ΔAP changing. If not, we have proved the difference.

This offers a guide for selecting documents to judge. To prove that $AP_1 > AP_2$, we pick documents that would benefit AP_2 if relevant but are in fact likely to be nonrelevant, or documents that would benefit AP_1 if relevant and are likely to be relevant. To be fair to both systems, we simply alternate trying to prove $AP_1 > AP_2$ and $AP_2 > AP_1$. Since AP is quadratic, each judgment influences our knowledge of the benefit of future judgments: knowing that document 1 is nonrelevant, for instance, would tell us that $\frac{1}{2}x_1x_2 + \frac{1}{3}x_1x_3 + \cdots$ is 0.

Expected MAP. In practice the number of judgments it takes to prove a difference in AP is quite large, but the marginal value of a judgment drops rapidly. At some point it becomes highly probable that we know the sign of the difference despite not having yet proved it. Let X_i be a Bernoulli random trial representing the relevance of document i, and $p_i = p(X_i = 1)$ the probability that document i is relevant. We can then estimate the expected value of ΔAP as

$$E[\Delta AP] = \frac{1}{\sum p_i}\sum_{i=1}^{n}\left(c_{ii}p_i + \sum_{j>i} c_{ij}p_ip_j\right).$$

The variance has a closed form as well [11].

It is straightforward to adapt this to the evaluation of a single system over multiple topics. Replace the rank constant c_{ij} with its single-system component to get $E[AP]$, then sum over topics to get $\mathcal{E}MAP$. In this work we present rankings of systems by $\mathcal{E}MAP$ and use ΔMAP in the context of comparing pairs of systems.

ΔMAP, and therefore $\mathcal{E}MAP$, converge to an approximately normal distribution over possible assignments of relevance, and thus can be understood by their expectation and variance. To determine the probability that ΔAP is less than zero, we simply look up the value in a normal distribution table. We refer to this probability as "confidence".

Calculating $\mathcal{E}MAP$ and confidence requires some estimate of the probability of relevance of each document. Carterette [10] described a method for using known relevance judgments and the performance of the systems to estimate the relevance of the unjudged documents they ranked.

2.2 Statistical Average Precision (statAP)

In statistical terms, average precision can be thought of as the mean of a population: the elements of the population are the relevant documents in the document collection and the population value of each element is the precision at this document for the list being evaluated. This principle is the base for many recently proposed evaluation techniques [20, 5, 2, 1]. Essentially there are two ways to vary implementations: (1) by choosing a certain sampling strategy and (2) by choosing a specific estimator of the population mean. For example, infAP [20] uses uniform sampling and the common mean estimator. For TREC 2007's MQ track, samples taken were of very small size (max. 40), which makes infAP unsuitable for the evaluation task, because small size samples taken uniformly contain very few (if any) relevant documents. We chose to use *statAP* [2, 1], which focuses on estimating the AP accurately with small size samples; the estimates can of course be used for any purpose, including for ranking the systems by performance. Following the sample-and-evaluate principle, *statAP* consists of the following choices.

Stratified Sampling, as developed by Stevens [8, 18], is very straightforward for our application. Briefly, it consists of bucketing the documents ordered by a chosen prior distribution and then sampling in two stages: first sample buckets with replacement according to cumulative weight; then sample documents inside each bucket without replacement according to hits registered at the previous stage. For the prior, we used the natural prior induced by the AP measure [5, 4, 3], but many other priors may work just as well. Stratified sampling has a number of desirable features, including practicality, proportionality to size ("pps"), easy computability and ability to include outside judgments [2, 1].

Generalized ratio estimator. Given a sample S of judged documents along with inclusion probabilities, in order to estimate average precision, *statAP* adapts the generalized ratio estimator for unequal probability designs [19]:

$$\mu = \frac{\sum_{k\in S} w_k/\pi_k}{\sum_{k\in S} 1/\pi_k}$$

where w_k is the value sampled and π_k is the inclusion probability for item k. For our problem, the population values are precisions at relevant ranks so for a given query and a particular system determined by ranking r(.), we have

$$statAP \;=\; \frac{1}{\widehat{R}}\sum_{d\in S}\frac{x_d \cdot \widehat{prec@r(d)}}{\pi_d}$$

where

$$\widehat{R} = \sum_{d\in S}\frac{x_d}{\pi_d}\;; \qquad \widehat{prec@k} = \frac{1}{k}\sum_{d\in S, r(d)\leq k}\frac{x_d}{\pi_d}$$

are estimates the total number of relevant documents and precision at rank k, respectively, both using the Horwitz-Thompson unbiased estimator [19]. To estimate mean average precision, we average the $statAP$ estimates across the judged queries set Q, obtaining $statMAP$ ("statistical mean average precision"), which we report in the experimental section of paper.

$$statMAP = \frac{1}{|Q|}\sum_{q\in Q} statAP_q$$

For the rest of the paper, $statAP$ refers to the estimated average precision for a query, and also to the sampling and estimation method; $statMAP$ refers to the estimated mean average precision of a run for a particular set of judged queries.

Confidence intervals. We can compute the inclusion probability for each document (π_d) and also for pairs of documents (π_{df}); therefore we can calculate an estimate of variance, $\widehat{var}(statAP)$, from the sample, using the ratio estimator variance formula found in [19], pp. 78 (see [2, 1] for details). Assuming the set of queries Q is chosen randomly and independently,

$$\widehat{var}(statMAP) = \frac{1}{|Q|^2}\sum_{q\in Q}\widehat{var}(statAP_q)$$

Assuming normally distributed $statMAP$ values, the 95% confident interval is given by $\pm 2std$ or $\pm 2\sqrt{\widehat{var}(statMAP)}$.

3. EXPERIMENT

As in other TREC tracks, sites participating in the Million Query Track were provided a set of queries to run through their retrieval engines, producing ranked lists of up to 1,000 documents from a given corpus for each query. The submitted runs were used as input to the MTC and statAP algorithms for selection of documents to be judged.

Corpus. The corpus was the GOV2 collection, a crawl of the .gov domain in early 2004 [13]. It includes 25 million documents in 426 gigabytes. The documents are a mix of plain text, HTML, and other formats converted to text.

Queries. A total of 10,000 queries were sampled from the logs of a large internet search engine. They were sampled from a set of queries that had at least one click within the .gov domain, so they are believed to contain at least one relevant document in the corpus. Queries were generally 1-5 words long and were not accompanied by any hints about the intent of the user that originally entered them. The title queries of TREC topics 701–850 were seeded in this set [14].

Retrieval runs. Ten sites submitted a total of 24 retrieval runs. The runs used a variety of methods: tf-idf, language modeling, dependence modeling, model combination; some used query expansion, in one case expanding using an external corpus. Some attempted to leverage the semi-structured nature of HTML by using anchor text, links, and metadata as part of the document representation.

Assessors. Judgments were made by three groups: NIST assessors, sites that submitted runs, and undergraduate work-study students. Upon logging in for the first time, assessors were required to go through a brief training phase to acquaint them with the web-based interface. After at least five training judgments, they entered the full assessment interface. They were presented with a list of 10 randomly-chosen queries from the sample. They selected one query from that list. They were asked to develop the query into a full topic by entering an information need and a narrative describing what types of information a document would have to present in order to be considered relevant and what information would not be considered relevant.

Each query was served by one of three methods (unknown to the assessors): MTC, statMAP, or an alternation of MTC and statMAP. For MTC, documents weights were updated after each judgment; this resulted in no noticeable delay to the assessor. StatMAP samples were selected in advance of any judging. The alternations proceeded as though MTC and statMAP were running in parallel; neither was allowed knowledge of the judgments to documents served by the other. If one served a document that had already been judged from the other, it was given the same judgment so that the assessor would not see the document again.

Documents were displayed with query terms highlighted and images included to the extent possible. Assessors could update their topic definitions as they viewed documents, a concession to the fact that the meaning of a query could be difficult to establish without looking at documents. Judgments were made on a tertiary scale: nonrelevant, relevant, or highly relevant. Assessors were not given instructions about the difference between relevant and highly relevance.

Assessors were required to judge at least 40 documents for each topic. After 40 judgments they were given the option of closing the topic and choosing a new query.

3.1 Judgments

There were three separate judging phases. The first, by NIST assessors and participating sites, was the longest. It resulted in 69,730 judged documents for 1,692 queries, with 10.62 relevant per topic on average and 25.7% relevant overall. This set comprises three subsets: 429 queries that were served by MTC, 443 served by statAP, and 801 that alternated between methods. Details of these three are shown in Table 1 as "1MQ-MTC", "1MQ-statAP", and "1MQ-alt".

Due to the late discovery of an implementation error, a second judging phase began in October with the undergraduate assessors and statAP judging only. This resulted in an additional 3,974 judgments for 93 queries, of which 21.22% were relevant (8.29 per topic on average). These were folded into the 1MQ-statAP set.

The TREC queries in this set had already been judged with some depth. These queries and judgments, details of which are shown in Table 1 as "TB", were used as a "gold standard" to compare the results of evaluations by MTC and statAP. It should be noted that these queries are not sampled from the same source as the other 10,000 and may not be representative of that space. They are, however, nearer to "truth" than any other set of queries we have.

There were two additional short judging phases with the

set	topics	judgments	rel/topic	% rel
TB	149	135,352	180.65	19.89%
1MQ-MTC	429	17,523	11.08	27.12%
1MQ-statAP	536	21,887	10.42	25.47%
1MQ-alt	801	33,077	10.32	24.99%
depth10	25	2,357	14.16	15.00%

Table 1: Judgment sets.

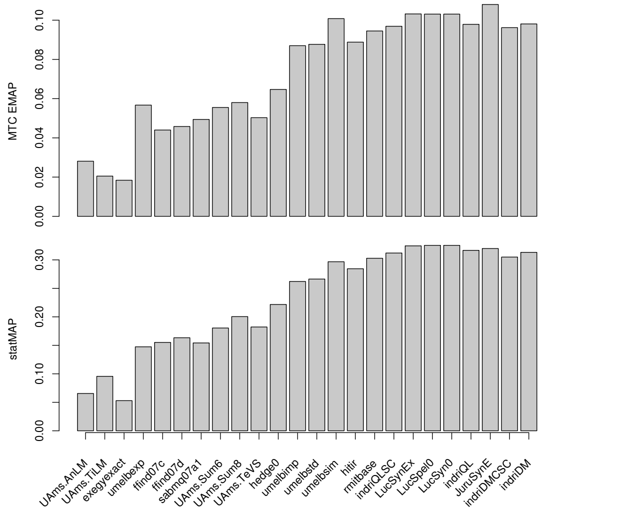

Figure 1: $\mathcal{E}MAP$ and statMAP evaluation results sorted by evaluation over 149 Terabyte topics.

goal of reinforcing the gold standard. For the first, a pool of depth 10 was judged for queries in the sample of 10,000; this is described in Table 1 as "depth10". One striking feature of depth10 is that assessors found many fewer relevant documents than in previous judging phases. Second, since some of our runs turned out to be poorly represented in the TB set, we made 533 additional judgments on top-ranked documents from sparsely-judged systems.

4. RESULTS

The 24 runs were evaluated over the TB set using `trec_eval` and over the 1MQ set using $\mathcal{E}MAP$ and statMAP. If TB is representative, we should see that $\mathcal{E}MAP$ and statMAP agree with each other as well as TB about the relative ordering of systems. Our expectation is that statMAP will present better estimates of MAP while $\mathcal{E}MAP$ is more likely to present a correct ranking of systems.

The left side of Table 2 shows the MAP for our 24 systems over the 149 Terabyte queries, ranked from lowest to highest. The average number of unjudged documents in the top 100 retrieved is also shown. Since some of these systems did not contribute to the Terabyte judgments, they ranked quite a few unjudged documents.

The right side shows $\mathcal{E}MAP$ and statMAP over the queries judged for our experiment, in order of increasing MAP over Terabyte queries. It also shows the number of unjudged documents in the top 100. $\mathcal{E}MAP$ and statMAP are evaluated over somewhat different sets of queries; statMAP excludes queries judged by MTC and queries for which no relevant documents were found, while $\mathcal{E}MAP$ includes all queries, with those that have no relevant documents having some probability that a relevant document may yet be found.

	149 Terabyte		1MQ		
run name	unjudg	MAP	unjudg	$\mathcal{E}MAP$	statMAP
UAms.AnLM	64.72	$0.0278^{\ddagger}$	90.75	0.0281	0.0650
UAms.TiLM	61.43	$0.0392^{\ddagger}$	89.40	0.0205	0.0938
exegyexact	8.81	$0.0752^{\ddagger}$	13.67	0.0184	0.0517
umelbexp	61.17	0.1251	91.85	$0.0567^{*\dagger}$	$0.1436^{\dagger}$
ffind07c	22.91	$0.1272^{\ddagger}$	77.94	0.0440	0.1531
ffind07d	24.07	0.1360	82.11	0.0458	0.1612
sabmq07a1	21.69	0.1376	86.51	0.0494	0.1519
UAms.Sum6	32.74	$0.1398^{\ddagger}$	81.37	0.0555	0.1816
UAms.Sum8	24.40	0.1621	79.92	0.0580	0.1995
UAms.TeVS	21.11	0.1654	81.35	0.0503	0.1805
hedge0	16.90	$0.1708^{\ddagger}$	80.44	0.0647	0.2175
umelbimp	15.40	0.2499	80.83	0.0870	0.2568
umelbstd	11.48	$0.2532^{\ddagger}$	82.17	0.0877	0.2583
umelbsim	10.38	$0.2641^{\ddagger}$	80.17	$0.1008^{*\dagger}$	$0.2891^{\dagger}$
hitir	9.06	0.2873	80.25	0.0888	0.2768
rmitbase	8.32	0.2936	79.28	0.0945	0.2950
indriQLSC	7.34	0.2939	79.18	0.0969	0.3040
LucSynEx	13.02	0.2939	78.23	0.1032^{*}	0.3184^{*}
LucSpel0	13.08	0.2940	78.27	0.1031	0.3194^{*}
LucSyn0	13.08	0.2940	78.27	0.1031	0.3194^{*}
indriQL	7.12	$0.2960^{\ddagger}$	78.80	0.0979^{*}	0.3086
JuruSynE	8.86	0.3135	78.36	0.1080	0.3117
indriDMCSC	9.79	0.3197	80.36	0.0962^{*}	0.2981^{*}
indriDM	8.67	0.3238	79.51	0.0981^{*}	0.3060^{*}

Table 2: Performance on 149 Terabyte topics, 1692 partially-judged topics per $\mathcal{E}MAP$, and 1084 partially-judged queries per statMAP, along with the number of unjudged documents in the top 100 for both sets.

Overall, the rankings by $\mathcal{E}MAP$ and statMAP are fairly similar, and both are similar to the "gold standard". Figure 1 shows a graphical representation of the two rankings compared to the ranking by Terabyte systems. Figure 2 shows how statMAP, $\mathcal{E}MAP$, and MAP over TB queries correlate. All three methods have identified the same three clusters of systems, separated in Table 2 by horizontal lines; within those clusters there is some variation in the rankings between methods. For statMAP estimates (Figure 2, left plot), besides the ranking correlation, we note the accuracy in terms of absolute difference with the TB MAP values by the line corresponding to the main diagonal.

Some of the bigger differences between the methods are noted in Table 2 by a * indicating that the run moved four or more ranks from its position in the TB ranking, or a † indicating a difference of four or more ranks between $\mathcal{E}MAP$ and statMAP. Both methods presented about the same number of such disagreements, though not on the same systems. The biggest disagreements between $\mathcal{E}MAP$ and statMAP were on umelbexp and umelbsim, both of which $\mathcal{E}MAP$ ranked five places higher than statMAP. Each method settled on a different "winner": indriDM for the TB queries, JuruSynE for $\mathcal{E}MAP$, and LucSpel0 and LucSyn0 tied by statMAP. However, these systems are all quite close in performance by all three methods.

We also evaluated statistical significance over the TB queries by a one-sided paired t-test at $\alpha = 0.05$. A run denoted by a ‡ has a MAP significantly less than the next run in the ranking. (Considering the number of unjudged documents, some of these results should be taken with a grain of salt.) Significance is not transitive, so a significant difference between two adjacent runs does not always imply a significant difference between other runs. Both $\mathcal{E}MAP$ and statMAP swapped some significant pairs, though they agreed with each other for nearly all such swaps.

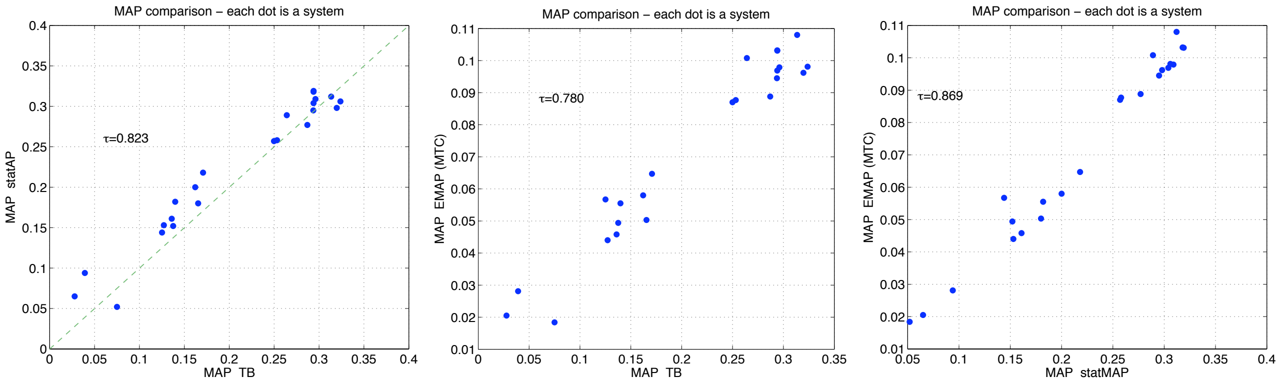

Figure 2: From left, evaluation over Terabyte queries versus statMAP evaluation, evaluation over Terabyte queries versus $\mathcal{E}MAP$ evaluation, and statMAP evaluation versus $\mathcal{E}MAP$ evaluation.

An obvious concern about the gold standard is the correlation between the number of unjudged documents and MAP: the tau correlation is $-.517$, or $-.608$ when exegyexact (which often retrieved only one document) is excluded. This correlation persists for the number unjudged in the top 10. To ensure that we were not inadvertently ranking systems by the number of judged documents, we selected some of the top-retrieved documents in sparsely-judged systems for additional judgments. A total of 533 additional judgments only discovered 7 new relevant documents for the UAms systems, 4 new relevant documents for the ffind systems, but 58 for umelbexp. The new relevant judgments caused umelbexp to move up one rank. This suggests that while the ranking is fair for most systems, it is likely underestimating umelbexp's performance.

It is interesting that the three evaluations disagree as much as they do in light of work such as Zobel's [21]. There are at least three possible reasons for the disagreement: (1) the gold standard queries represent a different space than the rest; (2) the gold standard queries are incompletely judged; and (3) the assessors did not pick queries truly randomly. The fact that $\mathcal{E}MAP$ and statMAP agree with each other more than either agrees with the gold standard suggests to us that the gold standard is most useful as a loose guide to the relative differences between systems, but does not meaningfully reflect "truth" over the larger query sample. But the possibility of biased sampling affects the validity of the other two sets as well: as described above, assessors were allowed to choose from 10 different queries, and it is possible they chose queries that they could decide on clear intents for rather than queries that were unclear. It is difficult to determine how random query selection was. We might hypothesize that, due to order effects, if selection was entirely random we would expect to see the top most query selected most, followed by the second-ranked query, followed by the third, and so on, roughly conforming to a log-normal distribution. This in fact is *not* what happened; instead, assessors chose the top-ranked query slightly more often than the others (13.9% of all clicks), but the rest were roughly equal (slightly under 10%). But this would only disprove random selection if we could guarantee that presentation bias holds in this situation. Nevertheless, it does lend weight to the idea that query selection was not random.

In an attempt to resolve some of these questions, we evaluated systems over the depth10 set using `trec_eval`. Evaluation results over this set do not correlate well to any previous set, with rank correlations in the $0.6 - 0.7$ range. In particular, the Luc* systems drop from the top tier to the second tier. This is a very interesting result that bears closer investigation, since if queries are a random sample it disagrees with the notion that two samples of queries can be used to evaluate systems reliably. Our current hypothesis (supported by conversations with the assessors) is that assessors selected these queries less randomly than other sets, so that they are not a representative sample. Note in Table 1 that the frequency of relevant documents in this set is significantly lower than any other set.

5. ANALYSIS

In this section, we describe a set of analyses performed on the data collected as described above. Our analyses are of two forms: (1) *efficiency studies*, aimed at determining how quickly one can arrive at accurate evaluation results and (2) *reusability studies*, aimed at determining how reusable our evaluation paradigms are in assessing future systems.

5.1 Efficiency Studies

The end goal of evaluation is assessing retrieval systems by their overall performance. According to the empirical methodology most commonly employed in IR, retrieval systems are run over a given *set of topics* producing a ranked list of document. The performance of each system per topic is expressed in terms of average precision of the output list of documents while the overall quality of a system is captured by averaging its AP values over all topics into its mean average precision. Systems are ranked by their MAP scores.

Hypothetically, if a second set of topics was available, the systems could be run over this new set of topics and new MAP scores (and consequently a new ranking of the systems) would be produced. Naturally, two questions arise: (1) how do MAP scores or a ranking of systems over different set of topics compare to each other, and (2) how many topics are needed to guarantee that the MAP scores or a ranking of systems reflect their actual performance?

We describe two efficiency studies, the first based on analysis of variance (ANOVA) and generalizability theory, and the second based on an empirical study of the stability of rankings induced by subsets of queries.

5.1.1 ANOVA and Generalizability Theory

Given different sets of topics one could decompose the amount of variability that occurs in MAP scores (as measured by variance) across all sets of topics and all systems into three components: (a) variance due to actual performance differences among systems—*system variance*, (b) variance due to the relative difficulty of a particular set of topics—*topics variance*, and (c) variance due to the fact that different systems consider different set of topics hard (or easy)—*system-topics interaction variance*.

Ideally, one would like the total variance in MAP scores to be due to the actual performance differences between systems as opposed to the other two sources of variance. In such a case, having the systems run over different sets of topics would result into each system obtaining identical MAP scores over all sets of topics, and thus MAP scores over a single set of topics would be 100% reliable in evaluating the quality of the systems. Note that among the three variance components, only the variances due to the *systems* and *system-topics interactions* affect the *ranking* of systems—it is these two components that can alter the relative differences among MAP scores, while the topic variance will affect all systems equally, reflecting the overall difficulty of the set of topics.

In practice, as already described, retrieval systems are run over a single given set of topics. The decomposition of the total MAP variance into the aforementioned components in this case can be realized by using tools provided by Generalizability Theory (GT) [6, 7].

We ran two separate GT studies; one over the MAP scores estimated by the MTC method given the set of 429 topics exclusively selected by MTC and one over the MAP scores estimated by the statAP method over the set of 459 topics exclusively selected by statAP (both methods utilized 40 relevance judgments per topic). For both studies we reported (a) the ratio of the variance due to system and the total variance and (b) the ratio of the variance due to system and the variance components that affect the relative MAP scores (i.e. the ranking of systems), both as a function of the number of topics in the topics set. The results of the two studies are illustrated in Figure 3. The solid lines correspond to the ratio of the variance due to system and the total variance and expresses how fast (in terms of number of topics) we reach stable MAP values over different sets of topics of the same size. As the figure shows, the statAP method eliminates all variance components (other than the system) faster than the MTC method, reaching a ratio of 0.95 with a set of 152 topics, while MTC reaches the same ratio with 170 topics. The dashed lines correspond to the ratio of the variance due to system and the variance due to effects that can alter the relative MAP scores (rankings) of the systems. The figure shows that the MTC method produces a stable ranking of systems over different sets of topics faster (in terms of number of topics) than the statAP method reaching a ratio of variance 0.95 with a set of 40 topics, while statAP reaches the same ratio with 85 topics.

These results further support the claims that the statAP method, by design, aims to estimate the actual MAP scores of the systems, while the MTC method, by design, aims to infer the proper ranking of systems.

5.1.2 Ranking stability

Figure 4 shows the τ correlation between both $\mathcal{E}MAP$ and statMAP rankings over 1000 queries with 40 judgments each and rankings by both measures over fewer queries and/or fewer judgments. 1000 queries were selected to make the x-axes equal; the two methods cannot use the same queries. These figures assume that the goal of reducing the assessor effort is to reach a ranking that is close to the one that would have been produced by the same method over all available judgments and these 1000 queries.

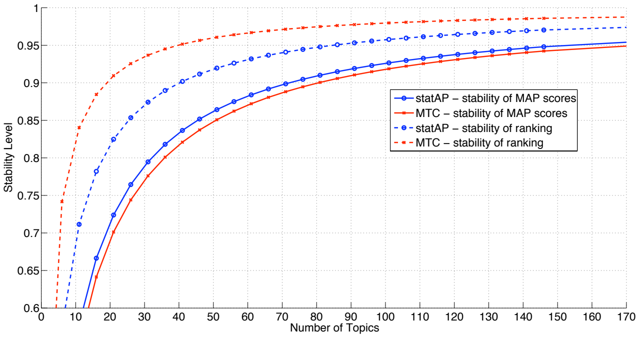

Figure 3: Stability levels of the MAP scores and the ranking of systems for statAP and MTC as a function of the number of topics.

In all cases the lines rise quickly as queries are added, then flatten. Each of the lines seems to asymptote to a point below $\tau = 1$; without certain judgments it may be impossible to reach the same level. This plot therefore depends to some extent on which documents were judged.

5.1.3 Cost analysis

Empirically, stability depends on the particular documents judged and how those judgments are used to make inferences and estimations. We can study stability empirically by selecting an *operating point*, then simulating evaluation runs to reach that point. The minimum cost required to reach some operating point given some parameter such as the number of judgments per query is a measure of stability.

For MTC, we will use as the operating point Kendall's τ rank correlation. If we want to ensure a τ of at least 0.9 to the ranking over all queries and all judgments, what is the minimum judging effort we need to expend?

Since MTC picks documents in an order, it is possible to simulate increasing numbers of judgments from 1 up to 40. To find the optimal cost point, we simulated increasing judgments, then increasing queries as in Figure 3 until we first reach a Kendall's τ of 0.9 between the ranking over the smaller set of queries and judgments and the full set.

As the figure shows, with 5 judgments per query MTC does not quite reach a 0.9 τ correlation with 1000 topics—it finishes at 0.872. With 10 judgments, τ reaches 0.9 with 900 topics (though 0.9 is well within the standard error with as little as 600 topics). With 20 judgments per topic, it only requires 250 topics to reach 0.9. Making an additional 20 judgments per topic does not provide much gain, as τ reaches 0.9 with only 50 fewer topics than with 20 judgments.

Assessor effort has two primary components: the amount of time spent making developing the topic and the amount of time spent making judgments. It took assessors a median time of about 14 minutes to judge 40 documents, and about five minutes to develop the topic[1]. Given that, the total effort needed to do 10 judgments for each of 600 topics is at

[1] We could not measure topic development time precisely, so this is a rough estimate based on mean time between viewing a new list of 10 queries and saving a topic description.

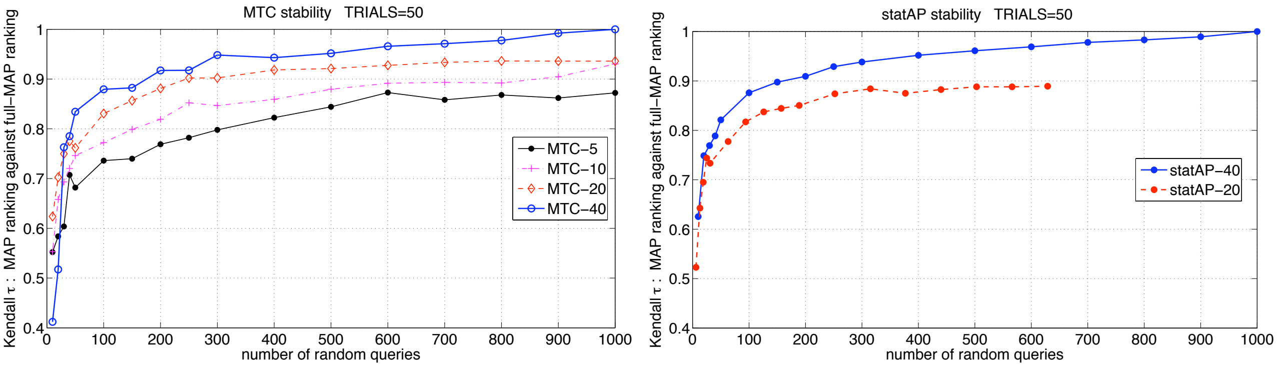

Figure 4: Stability of MTC and statAP

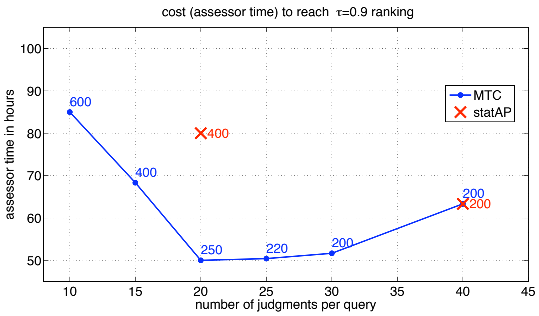

Figure 5: Total assessor cost required to reach a stable ranking. The number of queries required to reach $\tau = 0.9$ is indicated on the plot for both MTC (blue) and statAP (red).

least $\frac{1}{60}(5\cdot600+\frac{14}{40}\cdot10\cdot600) = 85$ hours (assuming 0.9 can be reached with 600 topics); with 20 judgments and 250 topics it is less than half that at 44 hours. With 40 judgments the time rises to 63 hours.

Figure 5 shows the total effort needed to reach a τ of 0.9 in hours as a function of the number of judgments. At each increasing judgment level, we found the minimum number of queries needed for a τ of 0.9, then calculated total cost by the formula above. Note that there is a precipitous drop followed by a gradual increase, with the minimum point at about 20 judgments and 250 topics. This is the least amount of effort that must be done to ensure that the ranking will not change significantly with more judgments or topics.

For *statAP*, a Kendall's τ of 0.9 can be obtained with slightly less than 200 queries, 40 judgments per query, for a total of 8000 judgments (Figure 4, right plot); a slightly lower τ can be obtained using about 400 queries for 20 judgments per query for the same total of 8000 judgments. These values correspond to 63 and 80 assessor hours respectively, shown in Figure 5.

5.2 Reusability

Reusability in the sense it is traditionally understood is impossible—we cannot predict what new systems will do, and as corpora keep getting bigger it will get harder to create test collections that work as well for new systems as they do for old. Instead, evaluation should report a confidence based on the missing judgments. Reusability should be understood in terms of how well the confidence holds up [10].

The two methods have different notions of confidence. As described in Section 2.1, MTC calculates confidence as the probability that the sign of ΔMAP is negative (or positive). Confidence in ΔMAP are over relevance judgments only; they ask what the probability is that there is a difference between two systems on a given set of topics. StatMAP calculates a confidence interval for the value of AP for each query, then a confidence interval for the value of MAP over the sample of queries.

run	$\mathcal{E}MAP$	conf	statMAP $\pm 2std$
exegyexact	0.0184	0.959	0.0517 ±0.0014
UAmsT07MAnLM	0.0205	1.000	0.0650 ±0.0019
UAmsT07MTiLM	0.0281	1.000	0.0938 ±0.0021
ffind07c	0.0440	1.000	0.1531 ±0.0022
ffind07d	0.0458	0.971	0.1612 ±0.0024
sabmq07a1	0.0494	0.703	0.1519 ±0.0022
UAmsT07MTeVS	0.0503	0.999	0.1805 ±0.0028
UAmsT07MSum6	0.0555	0.663	0.1816 ±0.0028
umelbexp	0.0567	0.688	0.1436 ±0.0037
UAmsT07MSum8	0.0580	0.999	0.1995 ±0.0028
hedge0	0.0647	1.000	0.2175 ±0.0029
umelbimp	0.0870	0.608	0.2568 ±0.0034
umelbstd	0.0877	0.690	0.2583 ±0.0033
hitir2007mq	0.0888	1.000	0.2768 ±0.0032
rmitbase	0.0945	0.842	0.2950 ±0.0031
indriDMCSC	0.0962	0.655	0.2981 ±0.0035
indriQLSC	0.0969	0.993	0.3040 ±0.0033
indriQL	0.0979	0.555	0.3086 ±0.0033
indriDM	0.0981	0.870	0.3060 ±0.0035
umelbsim	0.1008	0.808	0.2891 ±0.0038
LucSyn0	0.1031	0.583	0.3194 ±0.0036
LucSpel0	0.1031	0.681	0.3194 ±0.0036
LucSynEx	0.1032	0.996	0.3184 ±0.0035
JuruSynE	0.1080	NA	0.3117 ±0.0033

Table 3: Confidence estimates for $\mathcal{E}MAP$ and statMAP. For $\mathcal{E}MAP$, the confidence is the probability that the system has a lower MAP than the next system. For statMAP, they are confidence intervals.

Table 3 shows confidence estimates for the two methods. The left side is $\mathcal{E}MAP$; since to display all the information MTC provides would require a 24×24 table, we have limited the table to only the confidence between adjacent pairs in the ranking by $\mathcal{E}MAP$. The right side shows statMAP with confidence intervals calculated as in Section 2.2. These are described in more detail below.

5.2.1 MTC analysis

Confidence can be interpreted as the probability that two systems will swap in the ranking given more relevance judgments. If confidence is high, systems are unlikely to swap; the results of the evaluation can be trusted. If confidence is low, more judgments should be acquired. In that case, MTC can take any existing judgments and produce a list of additional judgments that should be made.

Since we do not have enough systems or judgments to be able to do standard leave-one-out reusability experiments, we instead investigated the ability of the confidence estimate to predict what would happen after more judgments. After the first 20 judgments by MTC over all topics, we calculated confidence between all pairs. We then completed the judgments. Pairs that had high confidence after the first 20 judgments should not have swapped. For MTC we used the same 1000 topics used for the stability experiment above.

The τ correlation between the 20-judgment ranking and the 40-judgment ranking is 0.928, so not many pairs swapped. Of those that did, half had a confidence of less than 0.6. The greatest confidence of any pair that swapped was 0.875; though it was not particularly likely to swap, it was not unimaginable. There were 243 pairs with confidences of greater than 0.95 with 20 judgments, and none of them swapped after the next 20.

5.2.2 statMAP confidence interval

Per query, the estimated interval length varies between 0 and ±2.6; for $statMAP$, assuming query independence, we obtain numbers varying from ±.0014 and ±.0038 for the 24 systems (Table 3). In most cases, if two confidence intervals (centered at the $statMAP$ value) overlap, it is a strong indication that the true MAP values are very close; when they do not, it is a strong indication that the MAP values are significantly different. Empirical tests using previous TREC data show that our estimator slightly underestimates the variance, accounting for about 90% of it, so in practice slightly larger confidence intervals should be used.

For independent queries, the standard deviation of $statMAP$ decreases linearly with the number of queries; with more than 1100 queries, the confidence interval length is very small and that truly reflects the confidence that the $statMAP$ value is very close to the mean of the estimator. However, while $\widehat{prec@k}$ and $\widehat{R}$ are unbiased estimators, the ratio estimator $statMAP$ it is not guaranteed to be unbiased and so the mean can be slightly different that the true AP value; therefore the overall confidence, especially for a large number of queries should not be derived solely from the estimated confidence intervals.

6. CONCLUSION

The Million Query Track and subsequent analyses show that we can evaluate retrieval systems with greatly reduced effort, beyond what was done for the track, and down to a few hundred queries with several dozen judgments for each one. Even when this is too few judgments to reliably distinguish between systems, we can still identify the systems that we have the least confidence in and focus on acquiring more judgments for them, thus ensuring future reusability.

The results of our study confirm those found by Sanderson & Zobel [16], Jensen [15], and Carterette & Smucker [12], all of which argued that evaluation over more queries with fewer or noisier judgments is preferable to evaluation over fewer queries with more judgments. There are tradeoffs, of course: failure analysis may be more difficult when judgments are scarce, and there may be limited data for training new algorithms. Exploring and quantifying these tradeoffs are clear directions for future work.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by the Defense Advanced Research Projects Agency (DARPA) under contract number HR0011-06-C-0023, in part by NSF grants IIS-0534482 and IIS-0533625, and in part by Microsoft Live Labs. Any opinions, findings, and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsors.

7. REFERENCES

[1] J. Allan, B. Carterette, J. A. Aslam, V. Pavlu, B. Dachev, and E. Kanoulas. Overview of the TREC 2007 Million Query Track. In *Proceedings of TREC*, 2007.

[2] J. A. Aslam and V. Pavlu. A practical sampling strategy for efficient retrieval evaluation, technical report.

[3] J. A. Aslam and V. Pavlu. Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions. In *Proceedings of ECIR*, pages 198–209. 2007.

[4] J. A. Aslam, V. Pavlu, and E. Yilmaz. Measure-based metasearch. In *Proceedings of SIGIR*, pages 571–572, 2005.

[5] J. A. Aslam, V. Pavlu, and E. Yilmaz. A statistical method for system evaluation using incomplete judgments. In *Proceedings of SIGIR*, pages 541–548, 2006.

[6] D. Bodoff and P. Li. Test theory for assessing ir test collection. In *Proceedings of SIGIR*, pages 367–374, 2007.

[7] R. L. Brennan. *Generalizability Theory.* Springer-Verlag, New York, 2001.

[8] K. R. W. Brewer and M. Hanif. *Sampling With Unequal Probabilities.* Springer, New York, 1983.

[9] C. Buckley, D. Dimmick, I. Soboroff, and E. Voorhees. Bias and the limits of pooling. In *Proceedings of SIGIR*, pages 619–620, 2006.

[10] B. Carterette. Robust test collections for retrieval evaluation. In *Proceedings of SIGIR*, pages 55–62, 2007.

[11] B. Carterette, J. Allan, and R. K. Sitaraman. Minimal test collections for retrieval evaluation. In *Proceedings of SIGIR*, pages 268–275, 2006.

[12] B. Carterette and M. Smucker. Hypothesis testing with incomplete relevance judgments. In *Proceedings of CIKM*, pages 643–652, 2007.

[13] C. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2004 Terabyte Track. In *Proceedings of TREC*, 2004.

[14] C. L. A. Clarke, F. Scholer, and I. Soboroff. The TREC 2005 terabyte track. In *Proceedings of TREC*, 2005.

[15] E. C. Jensen. *Repeatable Evaluation of Information Retrieval Effectiveness in Dynamic Environments.* PhD thesis, Illinois Institute of Technology, 2006.

[16] M. Sanderson and J. Zobel. Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proceedings of SIGIR*, pages 162–169, 2005.

[17] K. Sparck Jones and C. J. van Rijsbergen. Information retrieval test collections. *Journal of Documentation*, 32(1):59–75, 1976.

[18] W. L. Stevens. Sampling without replacement with probability proportional to size. *Journal of the Royal Statistical Society. Series B (Methodological), Vol. 20, No. 2. (1958), pp. 393-397.*

[19] S. K. Thompson. *Sampling.* Wiley Series in Probability and Mathematical Statistics, 1992.

[20] E. Yilmaz and J. A. Aslam. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of CIKM*, pages 102–111, 2006.

[21] J. Zobel. How reliable are the results of large-scale retrieval experiments? In *Proceedings of SIGIR*, pages 307–314, 1998.