

Inferring Document Relevance via Average Precision

Javed A. Aslam*, Emine Yilmaz
College of Computer and Information Science
Northeastern University
360 Huntington Ave, #202 WVH
Boston, MA 02115
{jaa,emine}@ccs.neu.edu

ABSTRACT

We consider the problem of evaluating retrieval systems using a limited number of relevance judgments. Recent work has demonstrated that one can accurately estimate average precision via a judged pool corresponding to a relatively small random sample of documents. In this work, we demonstrate that given values or estimates of average precision, one can accurately infer the relevances of unjudged documents. Combined, we thus show how one can efficiently and accurately infer a large judged pool from a relatively small number of judged documents, thus permitting accurate and efficient retrieval evaluation on a large scale.

Categories and Subject Descriptors

H3.4 [Information Storage and Retrieval]: Systems and Software – *Performance evaluation*

General Terms

Theory, Measurement, Experimentation

Keywords

Relevance Judgments, Average Precision

1. INTRODUCTION

We consider the problem of efficiently evaluating the performance of retrieval systems on a large scale. The large scale evaluation of retrieval systems requires significant human effort due to the relevance judgments needed. In TREC-style evaluations of retrieval systems, for each topic, a *pool* of the union of the top 100 documents retrieved by each system is formed, and these documents are assessed to determine their relevance for the associated topic. This method is referred to as *depth 100 pooling*, and the relevance judgments formed are stored in a file called the *qrel*.

The costs for such an assessment effort can be quite high; in TREC 8, for example, 86,830 relevance judgments were used to assess the quality of the retrieved lists corresponding to 129 system runs in response to 50 topics.

In recent work, we have shown that sampling techniques can be used to efficiently estimate standard measures of re-

trieval performance and that these random samples generalize well to previously unseen runs [1]. Our results demonstrate that accurate estimates of average precision, the number of documents relevant to a topic (R), and often other measure of retrieval performance can be obtained using small subsets of the complete judgment set.

One disadvantage of the aforementioned technique is that it cannot be used to assess the performance of systems in any standard way: in order to estimate the measures, a special procedure requiring access to information on the sampling process is needed, so standard tools such as `trec_eval` or other software implementations which calculate average precision and other performance measures cannot be used.

In this work, we demonstrate that given a set of ranked lists of documents submitted in response to a given topic, together with the average precisions associated with these lists and R , the number of documents relevant to the topic, one can accurately infer which underlying documents are relevant and which are not. When combined with the aforementioned techniques for accurately *estimating* average precision from a small sample of documents, one can effectively infer a large judged pool from a relatively small number of relevance assessments, thus permitting simple, standard, accurate, and efficient retrieval evaluation on a large scale.

2. METHODOLOGY

The methodology for inferring relevance assessments is conceptually simple: given the ranked lists of documents submitted in response to a given topic together with the average precisions associated with these lists and R , the number of documents relevant to the topic, find the binary relevance judgments associated with the underlying documents which minimize the “difference” between the given average precisions and those incurred by the inferred relevance assessments. This is a *constrained integer optimization* problem. *Constraints:* (1) The total number of relevant documents is R . (2) Any document contained in multiple lists must have the same relevance assessment. *Integrality:* The inferred assessments must be binary. *Optimization:* The average precisions incurred must be “close” to the given average precisions. Such a characterization of the problem gives rise to a number of issues.

First, this constrained integer optimization problem is intractable, for much the same reason that integer programming is intractable. To alleviate this problem, we relax the condition that the inferred relevance assessments must be binary. We instead allow the inferred relevance assessments to correspond to *probabilities* of relevance, and we deduce an

*We gratefully acknowledge the support provided by NSF grant CCF-0418390.

	docs judged	prec	recall	F_1	
	40	2.3%	0.5562	0.3833	0.4171
	95	5.5%	0.5919	0.5495	0.5332
	144	8.3%	0.6243	0.6004	0.5880
	260	15%	0.7068	0.6887	0.6906
	379	22%	0.7720	0.7361	0.7465
	494	28%	0.8101	0.7694	0.7835

Table 1: Precision, recall and F_1 values of the qrels inferred using 40, 95, 144, 260, 379 and 494 judgments for TREC 8.

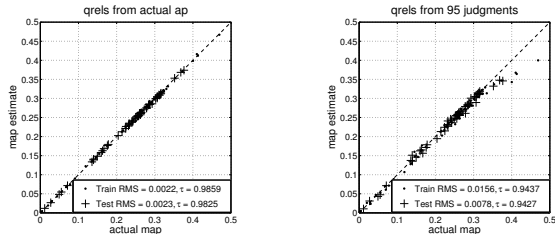


Figure 1: MAP estimates for training and test systems vs. actual MAP values obtained via inferred qrels from (left) actual ap values and (right) from estimates of ap obtained using 95 judgments for TREC 8.

expected value for average precision from these probabilistic relevance assessments as follows [2]

$$E[AP] = \frac{1}{R} \sum_{i=1}^Z \left(\frac{p_i}{i} \left(1 + \sum_{j=1}^{i-1} p_j \right) \right)$$

where p_i is the probability of relevance associated with the document at rank i in the list of length Z .

Second, we ensure that the inferred relevance judgments incur average precision values “close” to those given by minimizing the sum squared error between the actual and inferred expected average precision values. Thus, our optimization criterion is $\min \sum_i (E[AP_i] - ap_i)^2$ where ap_i is the given average precision associated with list i . The problem as formulated above can be solved using any number of constrained optimization routines, available, for instance, in MatLab.

Finally, we convert these probabilistic relevance assessments to binary relevance assessments by assigning a relevance score of 1 with probability p and a score of 0 with probability $1-p$, in the same spirit that linear programming with randomized rounding is used to solve integer programs.

When the given average precision (and R) values are actual estimates derived from sampling, we can also make use of the judged documents in the sample. Once the inferred qrels are obtained as above, we assign the known relevance assessments to the documents in the sample.

3. EXPERIMENTAL RESULTS

Table 1 shows the quality of the inferred qrels formed by using the estimates¹ obtained using $k = 40, 95, 144, 260, 379,$ and 494 judgments on average per topic for TREC 8.

¹Details on how these estimates were obtained may be found in our companion paper [1].

The complete judgment set contains 1737 judgments on average per topic; hence, these judgments correspond to 2.3, 5.5, 8.3, 15, 22, and 28% of the complete judgment set, respectively. Each row corresponds to an estimate obtained using a different sample of the given size, and the columns report the average of the precision, recall, and F_1 values over all queries when the inferred qrel file is treated as a labeled set and compared to the actual qrel file. Note that with estimates corresponding to 494 judgments on average per topic, an average qrel precision and recall of 81% and 77% is achieved. Even with estimates obtained from as few as 40 judgments on average per topic, the inferred qrels achieve non-trivial precisions and recalls.

Another way of evaluating the quality of these inferred qrels is to evaluate systems using these judgments and compare these assessments with the results obtained using the actual qrel; in particular, we are interested in how well these inferred qrels generalize to evaluating *unseen* runs. We separate the runs submitted to TREC 8 into *training* and *testing* sets: The 70 runs which contributed to the original TREC depth 100 pool (and thus the actual qrel) form the *training set*, while the 59 runs which did not contribute to the pool (or qrel) form the *testing set*. For each topic, we infer relevance judgments using the described method with estimates of average precision for the training runs obtained through k judgments on average per topic. We then evaluate the mean average precisions of the training and testing runs using these inferred judgments and compare these values with the actual mean average precisions of these runs.

Figure 1 shows the results of this experiment using the actual average precision values (and R) for TREC 8 as well as those results obtained from estimates using 95 judgments on average per topic. The y -axis is the mean average precision value of a run computed from the inferred qrels formed by the method, while the x -axis is the actual mean average precision. The plot also reports the root mean squared (RMS) error and the Kendall τ values in comparing the training and testing MAP values with actual MAP values. Note that estimated MAP values for both training and testing systems are very close to the actual MAP values both in terms of value and ranking purposes, demonstrating that the inferred qrels are accurate and generalize well.

In conclusion, we propose a method that can infer relevance judgments from estimates of average precision and R , and we demonstrate that these inferred relevance judgments can accurately assess the performance of retrieval systems, even when the inferred relevance judgments were effectively obtained from very few real relevance judgments.

4. REFERENCES

- [1] J. A. Aslam, V. Pavlu, and E. Yilmaz. A statistical method for system evaluation using incomplete judgments. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, August 2006. To appear.
- [2] J. A. Aslam, E. Yilmaz, and V. Pavlu. The maximum entropy method for analyzing retrieval measures. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 27–34. ACM Press, August 2005.