Do Batch and User Evaluations Give the Same Results? An Analysis from the TREC-8 Interactive Track

William Hersh, Andrew, Turpin, Susan Price, Dale Kraemer,
Benjamin Chan, Lynetta Sacherek, Daniel Olson
Division of Medical Informatics & Outcomes Research
Oregon Health Sciences University
Portland, OR, USA

An unanswered question in information retrieval research is whether improvements in system performance demonstrated by batch evaluations confer the same benefit for real users. We used the TREC-8 Interactive Track to investigate this question. After identifying a weighting scheme that gave maximum improvement over the baseline, we used it with real users searching on an instance recall task. Our results showed no improvement; although there was overall average improvement comparable to the batch results, it was not statistically significant and due to the effect of just one out of the six queries. Further analysis with more queries is necessary to resolve this question.

Introduction

A great deal of information retrieval (IR) evaluation research dating back to the Cranfield studies [1] and continuing through the Text Retrieval Conference (TREC) [2] is based on entering fixed query statements from a test collection into an IR system in batch mode with measurement of recall and precision of the output. It is assumed that this is an effective and realistic approach to determining the system's performance [3]. Some have argued against this view, maintaining that the real world of searching is more complex than can be captured with such studies. They point out that relevance is not a fixed notion [4], interaction is the key element of successful retrieval system use [5], and relevance-based measures do not capture user performance in some domains such as medicine [6].

The TREC Interactive Track is designed to assess real-user searching in the system-oriented TREC evaluation milieu. For the past three years (TREC-6, TREC-7, and TREC-8), the track has employed an "instance recall" task, where users are asked to identify instances of a topic [7]. Instance recall is defined as the fraction of total instances (as determined by the NIST assessor) for the topic that are covered by the documents saved by the user. Also measured is instance precision, which is the fraction of saved documents that contain one or more instances. Figure 1 shows two example queries from this year's track. The track allows a variety of hypotheses about IR systems to be evaluated with real users.

The goal of our effort for this year's Interactive Track was to assess whether IR approaches achieving better performance in the batch environment could translate that effectiveness to real users. This was done by first transforming queries, documents, and relevance judgements from the TREC-6 and TREC-7 interactive tracks into a test collection that could identify highly effective batch performance compared to a baseline. In particular, we focused on the newer weighting schemes that have shown to be effective with TREC data over the standard TF*IDF baseline. The most effective approach was chosen to serve as the "experimental" system while the standard TF*IDF served as the "control" system. Since we compared weighting schemes - a back-end functionality - the user interface for both systems was identical. We also evaluated two different searcher populations - librarians (mostly non-medical) and graduate students.

```
Number:
  414i
Title:
  Cuba, sugar, imports
Description:
  What countries import Cuban sugar?
Instances:
  In the time alloted, please find as many DIFFERENT countries of
  the sort described above as you can. Please save at least one
  document for EACH such DIFFERENT country.
  If one document discusses several such countries, then you need
  not save other documents that repeat those, since your goal
  is to identify as many DIFFERENT countries of the sort described
  above as possible.

Number:
  428i
Title:
  declining birth rates
Description:
  What countries other than the US and China have or have had
  a declining birth rate?
Instances:
  In the time alloted, please find as many DIFFERENT countries of
  the sort described above as you can. Please save at least one
  document for EACH such DIFFERENT country.
  If one document discusses several such countries, then you need
  not save other documents that repeat those, since your goal
  is to identify as many DIFFERENT countries of the sort described
  above as possible.
```

Figure 1 - Sample queries from the TREC interactive track.

Experiment 1 - Finding an effective weighting scheme for experimental system

The goal for the first experiment was to find the most effective batch-mode weighting scheme for interactive track data that would subsequently be used in interactive experiments. All of our batch and user experiments used the MG retrieval system [8]. MG allows queries to be entered in either Boolean or ranked mode. If ranking is chosen, the ranking scheme can be varied according to the Q-expression notation introduced by Zobel and Moffat [9].

A Q-expression consists of eight letters written in three groups, each group separated by hyphens. For example, BB-ACB-BCA, is a valid Q-expression. The two triples describe how terms should contribute to the weight of a document and the weight of a query respectively. The first two letters define how a single term contributes to the document/query weight. The final letter of each triple describes the document/query length normalization scheme. The second character of the Q-expression details how term frequency should be treated in both the document and query weight, e.g., as inverse document/query frequencies. Finally, the first character determines how the four quantities (document term weight, query term weight, document normalization, and query normalization) are combined to give a similarity measure between any given document and query. To determine the exact meaning of each character, the five tables appearing in the Zobel and Moffat paper must be consulted [9]. Each character provides an index into the appropriate table for the character in that position.

Although the Q-expressions permit thousands of possible permutations to be expressed, several generalizations can be made. Q-expressions starting with a B use the cosine measure for combining weights, while those starting with an A do not divide the similarity measure through by document or query normalization factors. A B in the second position indicates that the natural logarithm of one plus the number of documents divided by term frequency is used as a term's weight, while a D in this position indicates that the natural logarithm of one plus the maximum term frequency divided by term frequency is used. A C in the fourth position indicates a cosine measure based term frequency treatment, while an F in this position indicates Okapi-style usage [10]. Varying the fifth character alters the document length normalization scheme. Letters greater than H use pivoted normalization [11].

Methods

In order to determine the best batch-mode weighting scheme, we needed to convert the prior interactive data (from TREC-6 and TREC-7) into a test collection for batch-mode studies. This was done by using the description section of the interactive query as the query and designating documents as relevant to the query if one or more instances were identified in it. The batch experiments set out to determine a baseline performance and one with maximum improvement that could be used in subsequent user experiments. Each Q-expression was used to retrieve documents from the 1991-1994 Financial Times collection (used in the Interactive Track for the past three years) for the 14 TREC-6 and TREC-7 Interactive Track topics. Average precision was calculated using the trec_eval program.

Results

Table 1 shows the results of our batch experiments using TREC-6 and TREC-7 Interactive Track data. The first column shows average precision, while the next column gives the percent improvement over the baseline, which in this case was the BB-ACB-BAA (basic vector space TF*IDF) approach. The baseline was improved upon by other approaches shown to be effective in other TREC tasks (e.g., ad hoc), in particular pivoted normalization (second and third rows - with slope of pivot listed in parentheses) and the Okapi weighing function (remaining rows). The best improvement was seen with the AB-BFD-BAA measure, a variant of the Okapi weighing function, with an 81% increase in average precision. This measure was designated for use in our user experiments.

	Average precision	**% improvement**
BB-ACB-BAA	0.2129	0%
BD-ACI-BCA (0.5)	0.2853	34%
BB-ACM-BCB (0.275)	0.2821	33%
AB-BFC-BAA	0.3612	70%
AB-BFD-BAA	0.3850	**81%**
AB-BFE-BAA	0.3517	65%
AB-BFF-BAA	0.3287	54%
AB-BFG-BAA	0.3833	80%
AD-AFD-BAA	0.2432	14%
AI-AFD-BCA	0.2523	19%

Table 1 - Average precision and improvement for batch runs on TREC-6 and TREC-7 interactive data.

Experiment 2 - Interactive searching to assess weighting scheme with real users

Based on the results from Experiment 1, the goal of our interactive experiment was to assess whether the AB-BFD-BAA (Okapi) weighting scheme provided benefits to real users in the TREC interactive setting. We performed our experiments with the risk that this benefit might not hold for TREC-8 interactive data, but as seen in Experiment 3 below, this was not the case.

The OHSU TREC-8 experiments were carried out according to the consensus protocol developed for TREC-7 Interactive Track and continued this year [12]. We used all of the instructions, worksheets, and questionnaires developed by consensus, augmented with some additional instruments, such as tests of cognitive abilities and a validated user interface questionnaire.

Methods

The performance measures used in the TREC-8 interactive track were instance recall and instance precision. The searcher was instructed to look for instances of each topic. Relevance assessors at NIST defined the instances from pooled searching results from all experimental groups. Instance recall was defined as the proportion of true instances identified during a topic, while instance precision was defined as the number of documents with true instances identified divided by the number of documents saved by the user.

Both the baseline and Okapi systems used the same Web-based, natural language interface shown in Figure 2. MG was run on a Sun Ultrasparc 140 with 256 megabytes of RAM running the Solaris 2.5.1 operating system. The user interface accessed MG via CGI scripts which contained JavaScript code for designating the appropriate weighting scheme and logging search strategies, documents viewed (title displayed to user), and documents seen (all of document displayed by user). Searchers accessed each system with either a Windows 95 PC or an Apple PowerMac, running Netscape Navigator 4.0.

Librarians were recruited by advertising over several librarian-oriented listservs in the Pacific Northwest. The advertisement explicitly stated that we sought information professionals with a library degree and that they would be paid a modest honorarium for their participation. Graduate students were recruited from the Master of Science in Medical Informatics Program at OHSU. They had a variety of backgrounds, from physicians or other health care professionals to having completed non-health undergraduate studies.

The experiments took place in a computer lab. Each session took three and one-half hours, broken into three parts, separated by short breaks: personal data and attributes collection, searching with one system, and searching with the other system. The personal data and attributes collection consisted of the following steps, as described in more detail in the track plenary paper [12]:

1. Orientation to experiment (10 minutes)
2. Collection of Demographic/Experience data listed in Table 2 (10 minutes)
3. Collection of Cognitive data listed in Table 2 (40 minutes)
4. Orientation to searching session and retrieval system, with demonstration of a search (10 minutes)
5. Practice search using a topic from a previous interactive track (10 minutes)

The cognitive data was obtained by using tests from the Educational Testing Service (ETS) shown in past IR research to be associated with some aspect of successful searching.

Each participant was assigned to search three queries in a block with one system followed by three queries with the other system. A pseudo-random approach was used to insure that all topic and system order effects were nullified. (A series of random orders of topics with subject by treatment blocks were generated (for balance) and used to assign topics.) Table 2 shows a sample subject-block-topic assignment.

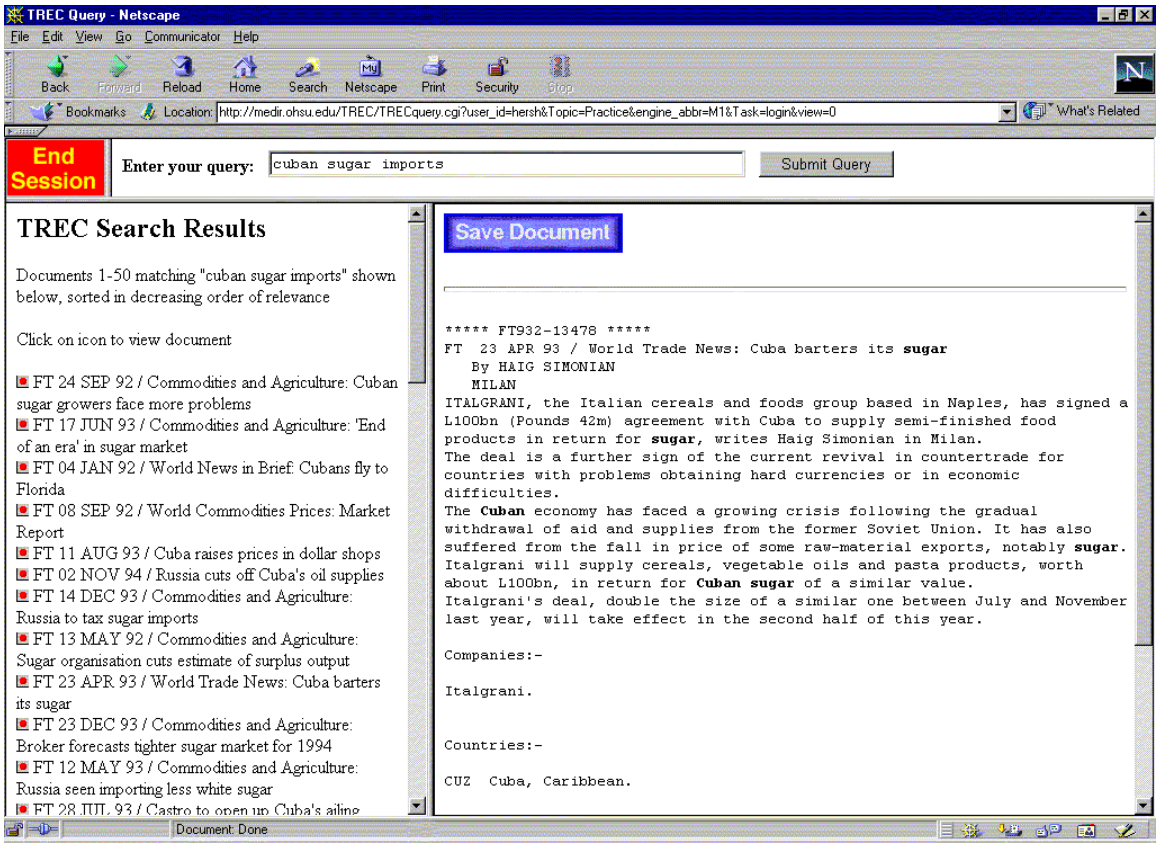

Figure 2 – Searching interface.

Subject	Block #1	Block #2
1	System 1: 6-1-2	System 2: 3-4-5
2	System 2: 1-2-3	System 1: 4-5-6
3	System 2: 2-3-4	System 1: 5-6-1
4	System 2: 3-4-5	System 1: 6-1-2
5	System 1: 4-5-6	System 2: 1-2-3
6	System 1: 5-6-1	System 2: 2-3-4
7	System 2: 6-1-2	System 1: 3-4-5
8	System 1: 1-2-3	System 2: 4-5-6
9	System 1: 2-3-4	System 2: 5-6-1
10	System 1: 3-4-5	System 2: 6-1-2
11	System 2: 4-5-6	System 1: 1-2-3
12	System 2: 5-6-1	System 1: 2-3-4

Table 2 – Sample subject-block-topic assignment for users.

The personal data and attributes collection was followed by a 10 minute break. The searching portion of the experiment consisted of the following steps:
1. Searching on first three topics with assigned system using searcher worksheet and post-topic questionnaire (60 minutes)
2. Post-System questionnaire for system used on first three topics (5 minutes)
3. Break (15 minutes)
4. Searching on second three topics with assigned system using searcher worksheet and post-topic questionnaire (60 minutes)
5. Post-System questionnaire for system used on second three topics and exit questionnaire (10 minutes)

Per the consensus protocol, each participant was allowed 20 minutes per query. Participants were instructed to identify as many instances as they could for each query. They were also instructed for each query to write each instance on the searcher worksheet and save any document associated with an instance (either by using the "save" function of the system or writing its document identifier down on the searcher worksheet).

The exit questionnaire was augmented from the consensus protocol to include the Questionnaire for User Interface Satisfaction (QUIS) 5.0 instrument [13]. QUIS provides a score from 0 (poor) to 9 (excellent) on a variety of user factors, with the overall score determined by averaging responses to each item. QUIS was given only at the end as a measure of overall user interface satisfaction since the interfaces for the two systems were identical.

An analysis of variance (ANOVA) model was fit to instance recall for these data. The factors in the model included type of searcher, the individual ID (nested in type), system, and topic. In the analysis, ID and topic were random factors, while type and system were fixed factors. Two-factor interactions (among system, topic, and type) were also included in the analysis. Residuals were examined for deviations from normality. All analyses were run in Version 6.12 of SAS for Windows 95.

Results

A total of 24 searchers consisting of 12 librarians and 12 graduate students completed the experiment. The average age of the librarians was 43.9 years, with seven women and five men. The average age of the graduate students was 36.5 years, with eight women and four men. All searchers were highly experienced in using a point-and-click interface as well as on-line and Web searching.

Table 3 shows instance recall and precision comparing systems and user types. While there was essentially no difference between searcher types, the Okapi system showed an 18.2% improvement in instance recall and an 8.1% improvement in instance precision, both of which were not statistically significant. Table 4 shows the p-values for the ANOVA model. Of importance was that while the difference between the systems alone was not statistically significant, the interaction between system and topic was. In fact, as shown by Figure 3, all of the difference between the systems occurred in just one query, 414i, which is shown above in Figure 1.

	Instance Recall	Instance Precision
System		
Baseline	0.33	0.74
Okapi	0.39	0.80
Type		
Librarian	0.36	0.76
Graduate Student	0.36	0.78

Table 3 - Instance recall and precision across systems and user types

Source	P-value
System	0.226
Topic	0.0516
Type	0.914
ID(Type)	0.0516
System * Topic	0.0269
System * Type	0.0881
Topic * Type	0.108

Table 4 - Summary of analysis of variance model for instance recall

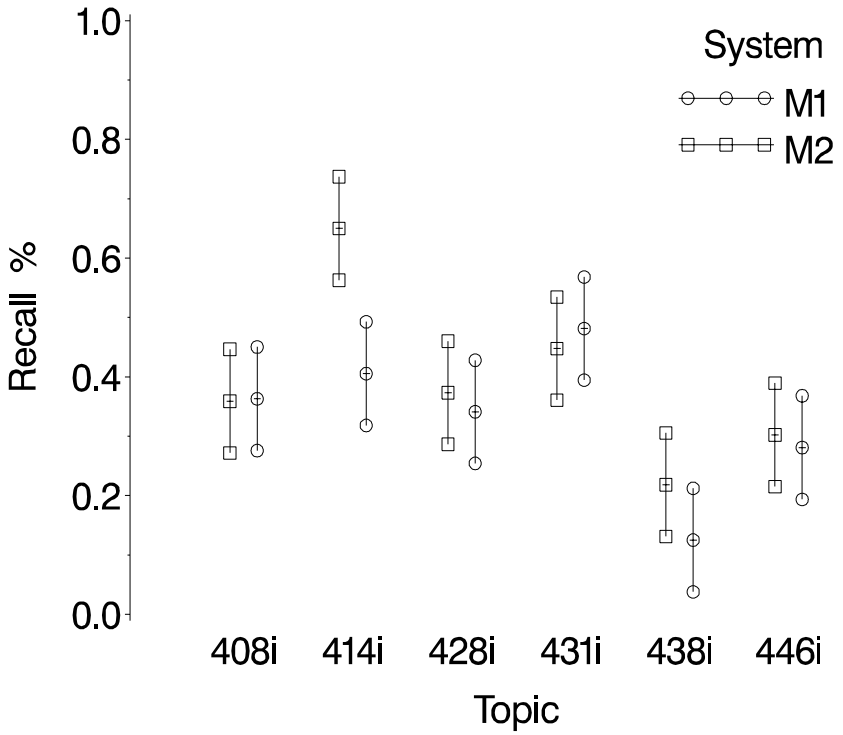

Figure 3 - Instance recall for each topic with each system (M1 = baseline, M2 = Okapi).

Experiment 3 - Verifying weighting scheme with TREC-8 data

Methods

Our final experiment consisted of verifying that the improvements in batch evaluation detected with TREC-6 and TREC-7 data held with TREC-8 data. The batch runs for the baseline and Okapi systems were repeated using the same approach of developing and using a test collection.

Results

Table 5 lists the average precision for both systems used in the user studies along with percent improvement. The Okapi AB-BFD-BAA still outperformed the baseline system, BB-ACB-BAA, but by the lesser amount of 17.6%. This happened to be very similar to the difference in instance recall noted in Experiment 2.

One possible reason for the smaller gains on the TREC-8 vs. TREC-6 and TREC-7 queries was that the average number of relevant documents for a TREC-8 query was three times higher than a query in the TREC-6 or TREC-7 sets. On average, TREC-6 interactive queries had 36 relevant documents, TREC-7 had queries 30 relevant documents, and TREC-8 queries had 92 relevant documents. The higher number of relevant documents may have given the baseline TF*IDF system a better chance of performing well, narrowing the gap between the different ranking schemes.

Also noteworthy in these results is that while query 414i achieved the second-best improvement of the six in average precision, it was far less than the improvement for 428i, which showed no improvement in the user studies. In fact, two queries showed a decrease in performance for Okapi with no difference in the user studies.

Discussion

While our experiments might be construed to suggest that retrieval systems which perform better in batch studies also do so in user studies, the actual picture is more complex. Although an improvement in the average performance was seen for a system that also performed better in batch studies, the difference was not statistically significant and occurred solely due to one query, 414i. The subject matter for this query was not markedly different from the others. The only difference was that it has far fewer relevant documents than the rest, which is likely to amplify random differences in user search strategies.

Query	Instances	Relevant Documents	Baseline	Okapi	% Improvement
408i	24	71	0.5873	0.6272	6.8%
414i	12	16	0.2053	0.2848	38.7%
428i	26	40	0.0546	0.2285	318.5%
431i	40	161	0.4689	0.5688	21.3%
438i	56	206	0.2862	0.2124	-25.8%
446i	16	58	0.0495	0.0215	-56.6%
Average	29	92	0.2753	0.3239	17.6%

Table 5 - Average precision and improvement for batch runs of TREC-8 data

Another possible interpretation of these data is that query 414i was an outlier and that differences in batch searching do not translate into better user searching. This view is supported by the large differences in the baseline and Okapi systems (positive and negative) which had no accompanying difference in the user studies.

The ultimate answer to the question of whether batch and user searching evaluations give the same results must ultimately be answered by further experiments that use a larger number of queries. The 20 queries accumulated for the Interactive Track over the last three years provides a larger base from which to start further investigations. Of course, to fully answer the question, other retrieval tasks must be represented as well, such as question-answering and high-recall situations as well.

References

1. Cleverdon CW and Keen EM, *Factors determining the performance of indexing systems*, . 1966, Cranfield UK: Aslib Cranfield Research Project.
2. Harman D. Overview of the first Text REtrieval Conference. In *Proceedings of the 16th Annual International ACM Special Interest Group in Information Retrieval*. 1993. Pittsburgh: ACM Press. 36-47.
3. Sparck-Jones K, Information Retrieval Experiment. 1981, London: Butterworths.
4. Meadow CT, Relevance? *Journal of the American Society for Information Science*, 1985. 36: 354-355.
5. Swanson DR, Information retrieval as a trial-and-error process. *Library Quarterly*, 1977. 47: 128-148.
6. Hersh WR, Relevance and retrieval evaluation: perspectives from medicine. *Journal of the American Society for Information Science*, 1994. 45: 201-206.
7. Lagergren E and Over P. Comparing interactive information retrieval systems across sites: the TREC-6 interactive track matrix experiment. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research And Development in Information Retrieval*. 1998. Melbroune, Australia: ACM Press. 162-172.
8. Witten IH, Moffat A, and Bell TC, Managing Gigabytes - Compressing and Indexing Documents and Images. 1994, New York: Van Nostrand Reinhold.
9. Zobel J and Moffat A, Exploring the similarity space. *SIGIR Forum*, 1998. 32: 18-34.
10. Robertson SE and Walker S. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM Special Interest Group in Information Retrieval*. 1994. Dublin: Springer-Verlag. 232-241.
11. Singhal A, Buckley C, and Mitra M. Pivoted document length normalization. In *Proceedings of the 19th Annual International ACM Special Interest Group in Information Retrieval*. 1996. Zurich, Switzerland: ACM Press. 21-29.
12. Hersh W and Over P. TREC-8 interactive track report. In *Proceedings of the 8th Text REtrieval Conference (TREC-8)*. 2000. Gaithersburg, MD: NIST. In press.
13. Chin JP, Diehl VA, and Norman KL. Development of an instrument measuring user satisfaction of the human-computer interface. In *Proceedings of CHI '88 - Human Factors in Computing Systems*. 1988. New York: ACM Press. 213-218.