Repeatable Evaluation of Search Services in Dynamic Environments

ERIC C. JENSEN
Summize, Inc.
STEVEN M. BEITZEL
Illinois Institute of Technology
ABDUR CHOWDHURY
Summize, Inc.
and
OPHIR FRIEDER
Illinois Institute of Technology and Georgetown University

In dynamic environments, such as the World Wide Web, a changing document collection, query population, and set of search services demands frequent repetition of search effectiveness (relevance) evaluations. Reconstructing static test collections, such as in TREC, requires considerable human effort, as large collection sizes demand judgments deep into retrieved pools. In practice it is common to perform shallow evaluations over small numbers of live engines (often pairwise, engine A vs. engine B) without system pooling. Although these evaluations are not intended to construct reusable test collections, their utility depends on conclusions generalizing to the query population as a whole. We leverage the bootstrap estimate of the reproducibility probability of hypothesis tests in determining the query sample sizes required to ensure this, finding they are much larger than those required for static collections. We propose a semiautomatic evaluation framework to reduce this effort. We validate this framework against a manual evaluation of the top ten results of ten Web search engines across 896 queries in navigational and informational tasks. Augmenting manual judgments with pseudo-relevance judgments mined from Web taxonomies reduces both the chances of missing a correct pairwise conclusion, and those of finding an errant conclusion, by approximately 50%.

Categories and Subject Descriptors: H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Evaluation, Web search

S. M. Beitzel is now at Telcordia Technologies.
Author's address: E. C. Jensen, email: ej@ir.iit.edu.

ACM Reference Format:
Jensen, E. C., Beitzel, S. M., Chowdhury, A., and Frieder, O. 2007. Repeatable evaluation of search services in dynamic environments. ACM Trans. Inform. Syst. 26, 1, Article 1 (November 2007), 38 pages. DOI = 10.1145/1292591.1292592 http://doi.acm.org/10.1145/1292591.1292592.

1. INTRODUCTION

Evaluating the effectiveness of information retrieval systems, in terms of relevance, requires a large amount of human effort. Many environments, such as the World Wide Web, grow and change too rapidly for a single evaluation to carry meaning for any extended period. Changes in their document collection, query population, and set of search services demand the repetition of evaluations over time. In these environments, static test collections become outdated too quickly and require too much effort to reconstruct. Rather, practitioners often compare a small number of live engines by judging every result retrieved at a shallow depth—without system pooling. The number of queries necessary for such an evaluation to be reliable[1] must be determined, however. We hypothesize that combining automatic evaluation techniques with a smaller set of manual relevance judgments can provide more reliable pairwise conclusions ("engine A outperforms engine B") than the manual set alone. We propose a semiautomatic framework for combining manually judged queries with automatically evaluated ones, our ultimate goal being to reduce manual evaluation effort by finding reliable conclusions using less manually judged queries. To test our hypothesis, we adopt the reproducibility probability ("... probability of observing a significant clinical result from a future trial ..." [Shao and Chow 2002]) as our estimate of reliability. We then compare conclusions drawn with high reproducibility probability from semiautomatic evaluations against those from a manual evaluation of the top ten results of ten Web search engines over 896 queries.[2]

The available content on the Web changes 8% every week, along with dramatic changes in the number of servers and pages [Cho et al. 2000; Ntoulas et al. 2004]. In our experimentation, we found that only 61% of Web search engines' top ten results remained the same three months later on average, and only 38% for the most changed engine [Jensen 2006]. Searchers' interests and the popular queries they use to express them are also in a constant state of flux, with 20% of even the 30,000 most popular queries changing from one week to the next, and less than half remaining the same after six months [Pass et al. 2006]. Even the topical categories these queries fall into have changing relative popularities within days, weeks, months, and years [Beitzel et al. 2004b, 2006; Jansen and Spink 2005]. Not only is the query population rapidly changing, but its size and diversity also indicate that a large number of queries are required to construct a representative random sample [Pass et al. 2006]. Popular queries and even popular query terms make up only a small portion of the total query stream, with approximately half of all queries being repeated ten or fewer times

[1]We follow Tague-Sutcliffe's definition of reliability as a general term meaning "the extent to which the experimental results can be replicated" and elaborate on specific applications where necessary [Tague-Sutcliffe 1996].
[2]Available from http://ir.iit.edu/collections

over a week [Beitzel et al. 2006; Jansen et al. 2005]. Developing new algorithms, or even tuning traditional retrieval strategies for emerging applications (image search, blog search, etc.) requires reliable, repeatable[3] evaluations on their respective dynamic environments.

Static test collections, such as those constructed for the Text Retrieval Conference (TREC), become outdated too quickly to address these changes in popular queries and their associated relevant results. With typical TREC evaluations requiring well over 500 assessor-hours (see Section 2), these sorts of collections are too expensive to reconstruct when changes in effectiveness over time must be measured. This effort is exacerbated by rapidly growing collection sizes, as the reusability of such collections depends on the depth of their pooled evaluations (also detailed in Section 2). Therefore, practitioners in dynamic environments often dispense with efforts to build reusable test collections in favor of reevaluating each engine as decisions are required. However, based on analysis of our manual evaluation, we find that such shallow judgments demand a large number of queries to provide reliable conclusions (as many as 650 in our environment). A method of reducing the effort needed to draw reliable conclusions in such an environment is needed.

To make the repetition of such large evaluations over time feasible, we propose a semiautomatic framework that incorporates automatically evaluated queries (using pseudo-relevance judgments) with manually judged ones. This provides insight into conclusions earlier in the evaluation process so that poorly performing engines can be eliminated before judging every result from every engine over a large query sample. We identify two methods for integrating automatic judgments. Each provides a different form of guidance for evaluators to reach reliable conclusions with less effort than manual judgments alone:

Semiautomatic Filtering: Verify conclusions drawn from a smaller number of manual judgments based on their agreement with automatic techniques.

Semiautomatic Prediction: Directly combine automatically judged queries with manual ones to yield samples of larger sizes whose conclusions can be used as an estimate of those that might be found with that many manual judgments.

To test our hypothesis that this semiautomatic framework yields more reliable conclusions than those available from the manually judged sample alone, we must adopt a specific method of estimating reliability. We use reproducibility probability (how likely a pairwise conclusion is to hold across any query sample of a given size) as our estimate of reliability for two reasons. First, measuring changes in performance over time in a dynamic environment demands conclusions that generalize to the query population as a whole at the time of evaluation. If applying an identical evaluation methodology to different query samples from the same time period yields inconsistent conclusions, nothing can be concluded about changes in engine performance over time. Comparing the conclusions from any two evaluations that use different query samples would

[3]We take some liberty with the term "repeatable" to reference both the feasibility in terms of effort of repeating evaluations and to emphasize that reproducibility is the fundamental criterion of reliability.

be impossible. Implicit in this assertion is our view of the query stream at a given point in time as a hypothetical infinite population, in following with the frequentist approach we adopt (well reviewed recently for information retrieval in Cormack and Lynam [2006]). Second, we seek to reduce manual evaluation effort by exploiting the fact that larger differences in evaluation scores are detectable with smaller query sample sizes, possibly available from an evaluation in progress. Information retrieval traditionally uses a priori heuristics for determining the necessary query sample size to yield a desired level of reliability, such as TREC "rules of thumb" about the minimum absolute difference between scores often derived from empirical meta-evaluation (see Section 2.2.3). However, these do not address the problem of detecting reliable conclusions from an evaluation in progress. We leverage the pointwise bootstrap estimate of reproducibility probability of hypothesis tests that quantifies the reliability of conclusions from any pairwise evaluation, without the prerequisite of a sufficient query sample size to estimate parameters such as the mean score difference or a context of meta-evaluation over a diverse set of engine pairs. The ability to develop intelligent evaluation strategies, such as discarding results from an engine that is clearly inferior based on a small number of judgments, is largely unexplored because the "running averages" available from evaluations over small query sample sizes have been shown to be unreliable when viewed as whole [Voorhees and Buckley 2002]. Quantifying the utility of intelligent evaluation strategies is also difficult using existing methods of comparing evaluations (meta-evaluation). For example, prior automatic evaluation and implicit preference research (reviewed in Section 2.3) focuses on optimizing the correlation of engine rankings from a purely automatic evaluation to a manual one. However, critical decisions such as which search service to employ, and so forth, demand a more rigorous comparison of conclusions drawn by these methods with those from manual judgment. By leveraging reproducibility probability, we ensure only conclusions with high reproducibility probability are compared; those that would not generalize to other query samples using the same evaluation technique are considered "ties."

Next, we review related work in information retrieval evaluation and reliability estimation. In Section 3, we show that our manual Web search evaluation is reliable and we validate reproducibility probability estimation techniques on it. Having established that prerequisite, we propose and validate our semiautomatic framework in Section 4 using two simple automatic evaluation techniques. Even with these naïve techniques, errors are often reduced by half compared to using small sets of manual judgments alone. More importantly, metrics for comparing evaluations and measuring the utility of semiautomatic methods are developed.

2. RELATED WORK

First, we review evaluation of information retrieval systems on the Web. We then examine four methods for estimating the reliability of evaluations: hypothesis testing, confidence intervals, empirical meta-evaluation, and reproducibility probability estimation. Finally, we review prior work in automatic evaluation techniques.

2.1 Web Search Evaluation

Evaluating the effectiveness (relevance) of live Web search engines provides many unique challenges because they operate on data that are continually changing [Hawking et al. 1999; Savoy and Picard 2001]. The set of popular Web queries and the relevant documents for those queries changes dramatically over time [Pass et al. 2006]. Previous studies concluded that overlap among results from different Web search engines was too high for them to be deemed significantly different [Ding and Marchionini 1996]. However, when a decision must be made, some form of reliable evaluation is necessary. Most of the work in evaluating search effectiveness follows the Text Retrieval Conference (TREC) methodology for constructing reusable test collections. TREC holds constant the document collection and query set, pooling the top ranked results up to a given depth (typically 100) from each engine and manually judging each document in this pool as relevant or not relevant. If this judgment depth is large enough, these collections are reusable, in that the relative effectiveness of runs from new engines over the same documents and queries can be evaluated simply by applying the existing judgments and assuming documents that are not judged are not relevant [Zobel 1998]. Studies of evaluation in TREC (meta-evaluations) have shown that although relevance is an ambiguous concept [Borlund 2003], variations in relevance judgments due to assessor disagreement do not destabilize evaluation [Voorhees 1998]. The TREC Web track applies this methodology to static Web document collections. The recognition that Web search users perform tasks other than the TREC standard informational task (searching for many relevant documents topically related to the query) has led to the incorporation of navigational homepage or named-page-finding evaluations that assume there is a single best-known item (sans duplications) the searcher wants to find. Most recently, TREC has begun to address the question of whether building reusable collections through pooled evaluation is scalable to terabyte-sized collections [Clarke et al. 2005]. Recent work by Sanderson and Zobel [2005] shows that judging only the top ten results of each engine provides reliable evaluation for less effort than system pooling.

Evaluations are very labor intensive. Our own precision oriented evaluation of the top ten results of ten Web search engines over 896 queries required 225 assessor-hours to complete [Jensen et al. 2005; Jensen 2006]. This is approximately 15 minutes per query, to assign binary relevance and choose the best result from an average of 43 distinct results. A previous navigational evaluation we performed, selecting only the best page and its duplicates from a pool of six Web search engines' results (about 25 on average) over 418 queries, required 87 hours, or approximately 12 minutes per query on average [Beitzel et al. 2003b]. Creating reusable test collections such as those developed in TREC requires a larger amount of effort. Recent TREC efforts have employed six assessors generally working 20 hour weeks for over a month [Soboroff 2006]. The TREC 2001 Web ad hoc search task required 761.25 assessor-hours to perform judgments over 50 topics, and an additional 283 hours to develop those topics. The 2004 terabyte track ad hoc task required 1037.5 hours total, with over half spent performing judgments over 50 topics. Even in the 2003 homepage/named page

task where the query was developed for a prechosen best document, the process of simply checking shallow retrieved pools for duplicates over the 300 queries required as many as 100–120 hours.

2.2 Estimating Evaluation Reliability

The ultimate goal of evaluation is to facilitate the construction of engines that are a "meaningful" improvement over the state of the art. However, this improvement (often characterized as a level of difference discernable to users) may be achieved through several iterations of reliable improvements. We specialize on Tague-Sutcliffe's [1996] definition of reliability for the case of a pairwise conclusion from an information retrieval evaluation as its reproducibility probability across any random query sample of equivalent size. We focus only on the reliability of conclusions, as minimum levels of difference could easily be incorporated into such analysis by selecting a different null hypothesis, and would only increase the required sample sizes. Next, we review several methods of estimating reliability.

2.2.1 *Hypothesis Tests.* Applying statistical hypothesis tests to information retrieval evaluations has a history of controversy, as most tests rely on observations conforming to continuous, often particular, distributions, but typical information retrieval evaluation metrics are bounded, discrete and often non-normal in nature [van-Rijsbergen 1979]. Bootstrap hypothesis tests, such as those applied to information retrieval evaluation by Savoy [1997] or Sakai [2006], do not require these assumptions because they estimate the empirical distribution by resampling thousands of times. When performing a handful of these tests, this computational cost is not of consequence, but estimating their power is computationally and theoretically challenging [Davidson and MacKinnon 2006]. *Therefore, we choose the Wilcoxon signed rank test with standard corrections for noncontinuity in our experimentation because its nonparametric nature does not require assuming a particular distribution, but it is easily calculable and maintains higher power than very simple tests such as the sign test* [Hollander and Wolfe 1973]. Although the reproducibility probability of any test could be estimated with the nonparametric bootstrap we leverage below, the distribution of scores in our evaluation motivated this decision. They failed a Shapiro-Wilk test for normality but did appear to be symmetric, as required for the Wilcoxon test [Jensen 2006]. Our own and others' experimentation with the t-test, the sign test with and without the "zero fudge," Wilcoxon test with both the continuity correction and normal approximation, and also an exact version of the Wilcoxon test that computes every permutation in the case of tied ranks, found none that resulted in substantially more reliable reproducibility probability estimates than others [Jensen 2006; Sanderson and Zobel 2005]. Our same prior investigation showed that reliable reproducibility probability estimates with a 95% confidence level would have required more queries, so we chose $\alpha = 0.10$. The fundamental problem with relying on the p-value from a single hypothesis test is that it does not address the problem of adequately representing the query population to ensure reproducibility [Goodman 1992]. Because of this, it is possible to find statistically significant differences

over a particular sample of queries that may not generalize to the query population.

Another factor that must be considered in applying hypothesis tests to information retrieval evaluation is that performing multiple tests with the same null hypothesis requires simultaneous testing procedures (such as the commonly used Bonferroni correction) to account for the overall larger probability of finding a significant result by random chance. Miller distinguishes between experiments designed for "uncovering leads that can be pursued further to determine their relevance to the problem" versus those that report final conclusions, suggesting that multiple test procedures are more important in the latter case [Miller 1981]. Our primary endpoint is comparing conclusions that result from pairwise hypothesis tests of semiautomatic evaluations versus those of benchmark manual ones. Because each comparison between a pair of engines has its own hypothesis, differing from others, multiple testing procedures are not required in our analysis. However, if the primary endpoint is to find the best engine or to rank the engines, multiple testing procedures for step-wise and pairwise comparisons should be considered, to ensure conservative estimates [Munzel 2001].

2.2.2 *Confidence Intervals.* Many advocate reporting confidence intervals for the parameter of interest (which in evaluation is typically the score difference) rather than hypothesis testing because they are easier to interpret correctly. Cormack and Lynam [2006] construct confidence intervals of average precision over varying document collections in the TREC informational task, using the bootstrap. If we use a confidence interval to decide whether or not one engine significantly outperforms another (by checking whether the null hypothesis, typically zero difference in scores, lies outside the interval), we are performing exactly the same analysis as the equivalent hypothesis test. Again, this does not address the problem of adequately representing the query population or quantifying reproducibility.

2.2.3 *Empirical Meta-Evaluation.* Empirical meta-evaluation (studying the results of an evaluation over a large number of engines) focuses on estimating the reliability of an evaluation as a whole. This sort of analysis is a key component of TREC, where it is almost exclusively applied due to the lack of such a large, diverse set of engines in proprietary environments. *In empirical meta-evaluation, reliability is defined as the stability (consistency) of the ranked list of engines across query sets.* Kendall's Tau or Spearman's rank correlation measures are often used to compare evaluations based on their ranking of engines. However, the most relevant metric to our work is the error rate (probability of a pair of engines flipping positions relative to one another in the ranked list of engines when using a different query sample) as defined in Buckley and Voorhees [2000]. It is estimated post hoc by counting the number of pairwise flips in the rankings of a large number of engines across varying query samples by resampling the pilot query set (total available judged queries) into smaller samples. In Voorhees and Buckley [2002], they focused on performing this calculation for several different query sample sizes up to half the size of the pilot sample and then extrapolating to estimate the error rate at the total pilot set's

size. By also calculating the error rate for several different fuzziness (minimum difference in average scores to not be considered a tie) values and leveraging the extrapolations to the pilot set size, these estimates can be used to devise a priori heuristics sometimes cited when planning or analyzing experiments at TREC. Typically, these take the form of "an X% difference in mean average precision is needed to ensure an error rate of less than 5% with 50 queries." However, these heuristics do not account for the differences in distribution of a particular pair of engines' scores (their variance, for example). While error rate is useful for post hoc comparison, these general heuristics derived from it are only applicable to a completed evaluation. Applying these heuristics to an evaluation in progress is questionable, as preliminary differences in average scores are subject to influence from outlying scores, such as zero and one.

Recent work builds on error rate with improved theoretical foundations. Sanderson and Zobel [2005] mitigate distributional issues by requiring both that a pair of engines pass a hypothesis test, and have a difference in average scores large enough to correspond with a low error rate. Lin and Hauptmann [2005] derive error rate from statistical principles, showing that variance in engines' scores dramatically impacts the reliability of evaluations. Sakai [2006] uses bootstrapping to find the score difference required to achieve a given significance level in bootstrap hypothesis testing. Each of these methods has in common the use of a large number of diverse runs to provide a general rule for the difference in average scores required. They do not address the problem of reducing the effort needed to answer specific questions without such a context, such as "does engine A outperform engine B?" from an evaluation in progress.

2.2.4 *Reproducibility Probability.* Although we are unaware of its application to information retrieval evaluation, we adopt reproducibility probability because it directly measures the likelihood that a particular pairwise conclusion generalizes to the query population as a whole, while its generality enables its application to evaluations in progress and does not require a large number of diverse engines as a context. Shao and Chow [2002] analyze several methods of estimating reproducibility probability. We follow their first in which "the reproducibility probability can be defined as an estimated power of the future trial using the data from the previous trial(s)." For clarity, we briefly diverge to differentiate between the true power (typically referred to simply as the "power"), or probability of rejecting a legitimately false null hypothesis, and this "estimated power" described by Shao and Chow. The true power (defined as an expectation in Equation 1 borrowing notation from Lehmann [1986]) is commonly used a priori in experimental design to determine what sample size will be large enough to detect a significance difference if one exists. However, calculating the true power depends on specifying a particular distribution, F, that satisfies the alternative hypothesis. This can be inaccurate when little is known about the actual distribution of observations [Bacchetti 2002]. When F is unknown, the true power of nonparametric tests is most accurately estimated using the bootstrap by creating artificial subsamples of a pilot sample in which the alternative hypothesis is enforced [Troendle 1999]. We are primarily concerned with comparing evaluations based only on their conclusions with

high reproducibility probability, not the true power of hypothesis tests, which determines the likelihood of detecting a significant difference where one exists.

$$\pi_{n,\alpha,F} = P(\Phi_\alpha(X_1, \ldots, X_n) = 1) = E_F[\Phi_\alpha(X_1, \ldots, X_n)]$$

Where: $X_1, \ldots, X_n$ are any random sample of independent random variables from identical distribution F satisfying H_1

Φ_α is a hypothesis test returning one for p-values less than or equal to α and zero otherwise

Equation 1. True power of a hypothesis test.

The "estimated power" method of estimating reproducibility probability described by Shao and Chow differs from this true power in that it comes from observed experimental data where the truth-value of the null hypothesis is unknown. For this reason, it is also known as the "observed power." Like Shao and Chow, however, we prefer the term "reproducibility probability" to avoid any implication about the truth of the null hypothesis (inference). Post hoc power analysis has drawn criticism for the way it has been misinterpreted as evidence against the null hypothesis for tests that do not reject the null hypothesis (since the observed power is greater than zero, we must simply not have a large enough sample to support our conclusions) [Hoenig and Heisey 2001]. This is the trap of the large sample; that any two nonidentical engines are significantly different with a large enough sample. We focus only on tests that do reject the null hypothesis and have high reproducibility probability at sample sizes just large enough to reliably estimate reproducibility probability (as analyzed in Section 3.3).

Although their definition is general, Shao and Chow [2002] only apply the "estimated power" approach to reproducibility probability for the parametric t-test. However, it can be applied in the general case (to include nonparametrics) using the point-wise bootstrap estimate, i.e. as done by De Martini [2006]. This pointwise estimate (Equation 2) is based on Efron and Tibshirani's [1993] nonparametric bootstrap, "A preliminary data set, $data_n$, is used to estimate a probability distribution, in this case $\hat{F}$. Then the desired power or sample size calculations are carried out as if $\hat{F}$ were the true distribution," as discussed in their example of estimating the true power of a bioequivalence test. We provide an algorithm implementing Equation 2 specifically for information retrieval in Section 3.2. Note that this is in the same spirit as the error rate heuristic discussed in Section 2.2.3, but formalizes reproducibility probability of a particular pairwise conclusion without requiring a completed evaluation over a large variety of engines.

$$p_{m,\alpha}(x_1, \ldots, x_n) = E_{\hat{F}}[\Phi_\alpha(x_1^*, \ldots, x_m^*)] = \frac{1}{B}\sum_{b=1}^{B} \Phi_\alpha\left(x_1^{*b}, \ldots, x_m^{*b}\right)$$

Where: $x_1, \ldots, x_n$ are the observed values of a pilot sample of independent random variables with empirical distribution $\hat{F}$

$x_1^*, \ldots, x_m^*$ are random subsamples (with repetition) of size m from $x_1, \ldots, x_n$

Equation 2. Point-wise nonparametric bootstrap estimate of reproducibility probability.

However, these reproducibility probability estimates are just that, estimates that are influenced by the variability in the observed data (pilot sample). The necessary pilot sample size required to reliably (reproducibly across pilot samples) estimate them must be established. We leverage graphical methods for comparing true power to estimates that have been developed for just this purpose [Collings and Hamilton 1988]. Aggregated numerical methods have also been introduced, but they are targeted at relative comparison of power estimation techniques rather than our focus of determining necessary sample sizes [De Martini and Rapallo 2003]. One method of making more conservative bootstrap estimates is to perform a double bootstrap, essentially performing secondary bootstrap replications of each of the bootstrap samples [De Martini 2006; Hall and Martin 1988]. When performing a hypothesis test for each bootstrap sample of reasonable size, this is computationally prohibitive. Our validation of the reliability of reproducibility probability estimates in Section 3.3 follows a limited version of this procedure, visualizing differences in estimates over several pilot samples.

2.3 Reducing Evaluation Effort

Two aspects of reducing evaluation effort have been studied in prior work: evaluation strategies that reduce the number of judgments needed in a manual evaluation, and automatic evaluation techniques that heuristically infer pseudo-relevance judgments. Studies in each of these areas suffer from a difficulty in comparing conclusions drawn from one evaluation to another: to ensure lower-effort techniques provide correct conclusions with respect to more thorough methods. Simply knowing that the engine rankings of one evaluation correlate with another does not address differing levels of confidence in conclusions and the associated issue of whether too many errant conclusions are being drawn or too few correct conclusions (too many ties) are found.

Several evaluation strategies are proposed as extensions or alternatives to the TREC pooling methodology to reduce manual effort. Soboroff [2006] focused on the problem of changes in the document collection, proposing to maintain existing TREC collections to limit the impact of these changes over time. Recent work dramatically improves on the evaluation effort required in TREC by intelligently selecting results to be evaluated [Aslam et al. 2006; Carterette et al. 2006]. Cormack et al. [1998] proposed interactive searching and judging, in which no system pooling is used; evaluators simply perform various queries for a topic, marking relevant documents as they proceed. Sanderson and Joho [2004] analyze methods of producing test collections without any system pooling and find that their quality correlates with that of TREC collections. Sanderson and Zobel [2005] quantified the relative advantage of not pooling in terms of the evaluation effort required to achieve a desired error rate.

Fully automatic evaluation techniques are widely employed in domains where manual evaluation would require a prohibitive amount of effort [Goldstein et al. 2005]. Two categories of automatic evaluation techniques

proposed for information retrieval are inferring pseudo-relevance judgments from the retrieved documents themselves, and using external resources to aid in this inference. Several approaches randomly sample the documents from the retrieved pools, based on known statistics about the typical distribution of relevant documents, as pseudo-relevant documents, but find that the effectiveness of only typical engines, but not the best engines, can be predicted [Aslam et al. 2003; Nuray and Can 2006; Soboroff et al. 2001; Wu and Crestani 2003]. Others use similarity functions between documents and the query to automatically estimate relevance [Shang and Li 2002].

Several methods of leveraging external resources to infer pseudo-relevance judgments have been proposed. Some advocate the use of click-through data (tuples consisting of a query and a user-clicked result) for automatic assessment. However, there is a well-known presentation bias inherent in these data: users are more likely to click on highly ranked documents regardless of their quality [Boyan et al. 1996]. Joachims et al. [2005] find that clickthrough data can, however, be used to infer relative preferences between documents. Others have made use of taxonomies to fuel automatic evaluation, such as the Open Directory Project (referred to as DMOZ or ODP), Yahoo's directory, and Looksmart [Haveliwala et al. 2002; Srinivasan et al. 2005]. These taxonomies divide the Web into a hierarchy of categories, with some pages placed in multiple categories. Each category has a title, and a path that represents its placement in the hierarchy. They also typically have editor-entered page titles that do not necessarily correspond to the titles of the pages themselves.

3. RELIABLE MANUAL EVALUATION

Evaluating our semiautomatic framework requires a reliable manual evaluation for comparison. We are unaware of currently available large manual evaluation in a dynamic environment, such as the Web. Therefore, we performed our own evaluation of ten Web search engines over 896 queries (based on the assessor time we allocated, with no preference for this particular number). We briefly review this experimental environment, with more details available in Jensen [2006]. We then examine the question of reliability of conclusions drawn from such an evaluation. With prior techniques for estimating reliability inapplicable, we review reproducibility probability, and specifically the pointwise bootstrap estimate that we leverage. As with any reliability estimate, the conditions for the estimate itself to be reliable must be verified. We therefore continue by validating the reliability of these reproducibility probability estimates themselves, finding the minimum query sample size necessary in our environment to ensure that high reproducibility probability estimates from a sample correspond to similarly high levels on larger samples.

3.1 Experimental Environment

We manually evaluated the top ten results of ten Web search engines over 896 queries without system pooling. The engines evaluated (AltaVista, AllTheWeb, Gigablast, Google, Lycos, MSN, MSN Tech Preview (now their main engine), Teoma, Wisenut, and Yahoo) are anonymized in no particular order as E1, E2,

Table I. Overall Scores for Manual Web Evaluation of 896 Queries

	AvgP			**P@10**			**MRR**	
Ranking	***Mean***	***Median***	***Ranking***	***Mean***	***Median***	***Ranking***	***Mean***	***Median***
E1	0.632	0.686	E1	0.690	0.800	E1	0.359	0.125
E2	0.620	0.676	E2	0.681	0.800	E2	0.338	0.111
E3	0.611	0.646	E3	0.676	0.800	E10	0.313	0.000
E5	0.607	0.642	E5	0.672	0.800	E3	0.312	0.000
E4	0.600	0.630	E4	0.667	0.800	E5	0.311	0.000
E7	0.585	0.600	E7	0.657	0.700	E7	0.300	0.050
E10	0.573	0.580	E9	0.635	0.700	E6	0.291	0.000
E6	0.572	0.600	E10	0.634	0.700	E4	0.283	0.000
E9	0.568	0.600	E8	0.634	0.700	E8	0.282	0.000
E8	0.562	0.600	E6	0.625	0.700	E9	0.241	0.000

. . . E10. We randomly sampled 896 distinct queries from an AOL Search query log consisting of the entire search traffic, hundreds of millions of queries, for the two days 9/17 and 9/18, 2004. Queries in the log are lowercased and stripped of most punctuation. We were careful to randomly select from the true distribution of queries, creating a sample that approximates the frequency distribution of the query population [Beitzel et al. 2004b, 2006]. Results from all ten engines were pooled in a uniform interface based on canonicalized URL, including the surrogate document representations consisting of title, snippet, and the link to the page that assessors could optionally click through. For each query, a group of AOL editors, undergraduate and graduate computer science student assessors manually assigned each result as relevant or not relevant and selected a single best result from the entire pool. Assessors were instructed to imagine they had posed the query to determine the most likely information need based on only the typically short query from the log. Of course, this environment may suffer from problems of assigning navigational and informational interpretations to each query, shifting definitions of relevance, or differing perceptions of relevance based on the quality of surrogates. Our focus on finding conclusions that generalize to the query population as a whole, motivated us to use the limited number of assessor hours available to us to judge more queries rather than attempt to reduce such sources of random error. A more controlled evaluation environment would likely reduce the number of queries required, but at a cost of higher effort per query. Detailed statistics about this evaluation, including score distributions, and so on, are available in Jensen [2006]. For each engine over each query, we calculated three evaluation metrics: average precision at ten (precision averaged at each retrieved relevant document, limiting the denominator to the maximum number of retrieved results, ten), denoted as AvgP, precision at ten, denoted as P@10, and reciprocal rank of the best page, denoted as MRR for familiarity despite our point-wise use of it. See Table I for mean and median scores, ranked by mean.

Next, we performed pairwise hypothesis testing for significant differences in median score using the Wilcoxon test as motivated by Section 2.2.1. While overall median is not terribly descriptive in Table I due to the discretized nature of a top-ten evaluation, simply the discrepancies between means and medians are

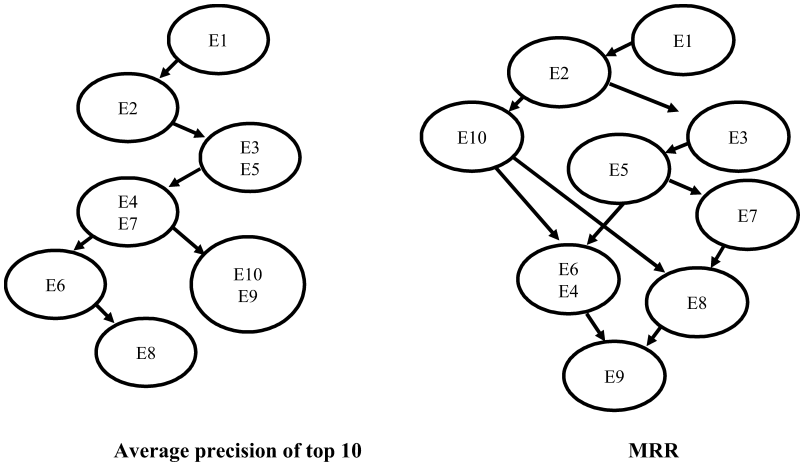

Fig. 1. Significant differences with Wilcoxon test $\alpha = 0.10$ using all 896 queries.

indicative of non-normal distributions. We visualize the significant differences found as a hierarchy, where any path to a lower node represents that the higher node significantly outperforms the lower one (Figure 1). These hierarchies are simply a visualization conveying the same information as more common textual approaches to represent groups, such as those provided in TREC using IR-STAT-PAK [Blustein and Tague-Sutcliffe 1995]. We believe they are more readable than purely textual approaches when engines are not strictly ranked by their average scores. Nodes are collapsed together when they have an equivalent set of relationships. For example, E1 significantly outperforms every other engine under the MRR evaluation metric because there is a path from E1 to E2 to E3 and E10, and so on. E6 and E4 under MRR, by contrast, neither outperform nor are outperformed by E8, but they both significantly outperform E9. We make our best effort to place engines with larger scores higher, but favor readability over enforcing this strictly. As one would hope for any measure of reliability, all of our results produce figures that are associative, never requiring more than one node to represent an engine.

Precision at ten is not shown, as it is nearly identical to average precision over the top ten results, with only two differences in significant conclusions, E6 $>$ E8 with AvgP (read "engine six significantly outperforms engine eight") and E9 $>$ E6 with P@10. While average precision is not typically used for retrieved sets of ten, it does help to reduce the number of tied scores across engines, compared to the more discretized P@10 (see Jensen [2006]), which we hypothesized would increase reliability. However, we do not find any meaningful differences in either the reliability or conclusions of AvgP versus P@10 (see Section 3.3), so for the remainder of this article we simply choose AvgP.

3.2 Bootstrapping Reproducibility Probability

The algorithm we employ for bootstrap estimates of reproducibility probability in pairwise information retrieval evaluations is detailed in Figure 2. This is

Given:

- A random sample of a distinct set of queries Q with size n from a query population
- Scores $x_{A,1}...x_{A,n}$ and $x_{B,1}...x_{B,n}$ for each query in Q from engines $\mathrm{E_A}$ and $\mathrm{E_B}$

Set $\mathrm{C(E_A > E_B) = 0}$, $\mathrm{C(E_B > E_A) = 0}$

For B iterations:

1. Randomly select, with repetition, a sample of queries Q^* with size m and their corresponding scores $x^*_{A,1}...x^*_{A,m}$ and $x^*_{B,1}...x^*_{B,m}$ from the pilot set Q
2. If $\Phi_\alpha(x^*_{A,1}...x^*_{A,m}, x^*_{B,1}...x^*_{B,m}) = 1$, a paired, one-sided test with $H_1 : E_A > E_B$ over Q^* rejects the null hypothesis with level α, increment $\mathrm{C(E_A > E_B)}$
3. If $\Phi_\alpha(x^*_{B,1}...x^*_{B,m}, x^*_{A,1}...x^*_{A,m}) = 1$, a paired, one-sided test with $H_1 : E_B > E_A$ over Q^* rejects the null hypothesis with level α, increment $\mathrm{C(E_B > E_A)}$

Estimate reproducibility probability of conclusion $\mathrm{E_A > E_B}$ using observed power $p_{m,\alpha} = \frac{\mathrm{C(E_A > E_B)}}{B}$ and the converse, discarding estimates below threshold p_{min}

Fig. 2. Bootstrap reproducibility probability estimates for pairwise evaluation.

a specialization of the nonparametric bootstrap (from prior work described in Section 2.2.4, particularly an implementation of Equation 2) for the pairwise information retrieval evaluation problem. We first analyzed point-wise estimates such as this in a preliminary investigation [Jensen et al. 2005]. For generality, we leave the hypotheses stated as $E_A > E_B$ ("engine A significantly outperforms engine B"), and the converse, because the specific hypotheses depend on the test chosen. The null hypothesis for both tests is that there is no difference between the two engines. We favor one-sided tests because the conclusions we are ultimately interested in are whether one engine outperforms another, not simply whether they differ. Implicit in deciding the direction of differences is the risk of type III error ("actually drawing firm but incorrect conclusions"), but for even minimal differences in engines this risk is small [Spiegelhalter and Freedman 1986]. For the conclusions included in our comparisons, calculating reproducibility probability for each direction makes this choice abundantly clear: we compare only conclusions with at least 90% reproducibility probability, in which case the converse conclusions typically have reproducibility probability less than 1%. Performing this procedure for every pair of $k = 10$ engines results in $k(k-1) = 90$ reproducibility probability estimates, from which we simply discard the weakest estimate of each pair $E_A > E_B$ or $E_B > E_A$ leaving $k(k-1)/2 = 45$ estimates in our analyses. Throughout our experimentation, we

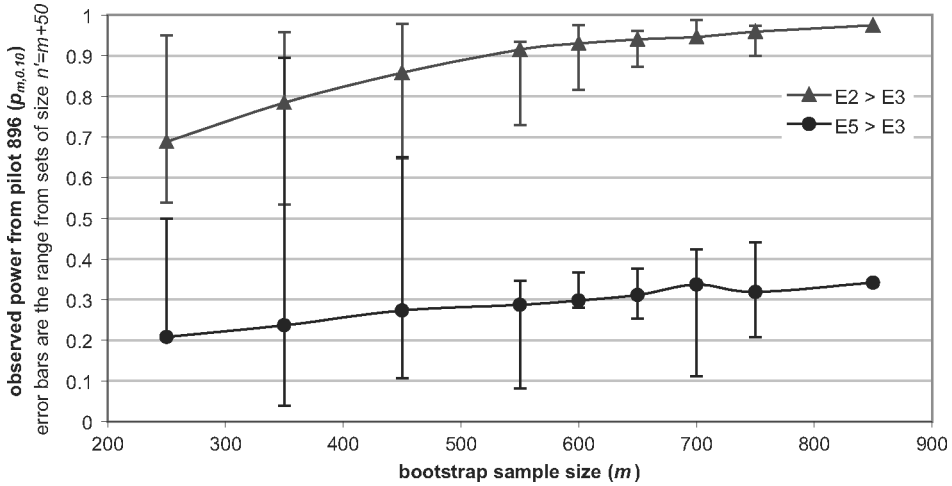

Fig. 3. Example growth and error of reproducibility probability estimates using AvgP.

set the number of bootstrap iterations, $B = 2{,}401$ (where B has no relation to engine B which we always represent as E_B). We have no preference for such an odd number, except that it is larger than the recommended minimums for bootstrap calculations, including those for bootstrapped hypothesis tests [Davidson and MacKinnon 2000]. Preliminary experimentation also confirmed this was more than sufficient.

3.3 Reliability of Point-wise Bootstrap Power Estimates

The margin of error for reliability estimates due to variability in their pilot samples is rarely studied. Since we cannot evaluate the entire query population, any estimate of reliability is biased by the pilot query sample used to calculate it. We focus on determining the sample size required to ensure that high reproducibility probability estimates from any pilot sample correspond to similarly high reproducibility probability estimates for the same engine pair from our entire sample of 896 queries. Although this analysis must be performed separately in each evaluation environment, it serves as a simple method for establishing that high reproducibility probability estimates converge, in that they remain high across pilot samples at a particular pilot sample size. Requiring a certain number of pilot queries simply to estimate reproducibility probability would seem to dissolve all hope of reducing evaluation effort, but, as we demonstrate in Section 5, incorporating automatic judgments allows us to meet this minimum sample size without manually evaluating each query.

In Figure 3, we provide an example of the growth of reproducibility probability for two example engine pairs with increasing bootstrap sample size m (and corresponding size of pilot samples n'). Hereafter, the Wilcoxon test with $\alpha = 0.10$ is assumed. The scores for these three engines and their associated rankings are detailed in Section 3.1. The points on the lines of Figure 3 provide relatively smooth curves because they are estimates from the same pilot sample Q of all 896 queries. The error bars, however, represent the range of

reproducibility probability estimates calculated using several other pilot samples Q' created by randomly sampling $m + 50$ queries from Q. Throughout, we use a bootstrap sample size of 50 less than the pilot sample to dampen the issue of tied score differences due to duplicated queries created by sampling with repetition. Equivalent score differences result in tied ranks in the Wilcoxon test that reduce its accuracy. Error bars are not shown for $m = 850$, because creating pilot samples that vary substantially out of the 896 queries available is not possible. With over 600 queries, we are able to conclude that E2 reliably outperforms E3 (their median AvgP scores are .676 and .646, respectively). The candidate conclusion E5 > E3, however, clearly lacks the reproducibility probability to support it with these sample sizes. With a very large number of queries, we might expect to be able to distinguish between E3 and E5 reliably. As discussed in Section 2.2.4, increasing the sample size until significant differences are found is a dangerous and inefficient method of comparing engines. Any nonidentical engines can be declared significantly different with a large enough sample size. *As our goal is to compare evaluations that use differing query samples, the sample sizes used in our analysis are determined by the reliability of reproducibility probability estimates for any engine pair, not the significance or reproducibility of particular conclusions*.

How can we use reproducibility probability to determine the sample size necessary to ensure reliable conclusions? One option would be to extrapolate reproducibility probability estimates from smaller sample sizes to project the sample size at which a conclusion will be reliable, as is often done for error rate. However, we can see from the error bars in Figure 3 that estimates based on small pilot samples vary wildly. Having only evaluated 450 queries, for example, we might extrapolate that with 650 we would find a reliable difference between E5 and E3. Instead, we favor a conservative approach of evaluating enough queries to make it clear that reproducibility probability estimates are converging to similar values across varying pilot samples for all pairs of engines.

The discrepancies between reproducibility probability estimates from one pilot sample to another can be dramatic, even with substantial numbers of evaluated queries. For example, in Figure 4 we plot reproducibility probability estimates over all 45 pairs' candidate conclusions ($E_A > E_B$) at bootstrap sample size 450 from varying pilot samples of 500 queries (created as described for Figure 3) versus identically sized estimates using all 896 queries as the pilot sample. Just as varying pilot samples produced large error margins in Figure 3, here we see that reproducibility probability estimates above 0.9 from a pilot of 500 queries might correspond to estimates as low as 0.4 for the same conclusion when using all 896 queries.

To determine the minimum query sample size necessary to ensure that highly reproducible probability estimates from a given pilot sample will correspond to similarly high estimates from other samples, we employ a simple metric: the minimum reproducibility probability estimate from a pilot sample to ensure a reproducibility probability of at least 90% using our entire sample of 896 queries as the pilot. In Figure 4, for example, we would judge that 500 queries are insufficient because only sample estimates very near 1.0 meet this criterion. The corresponding y-axis estimates from all 896 queries are below 0.9 for even

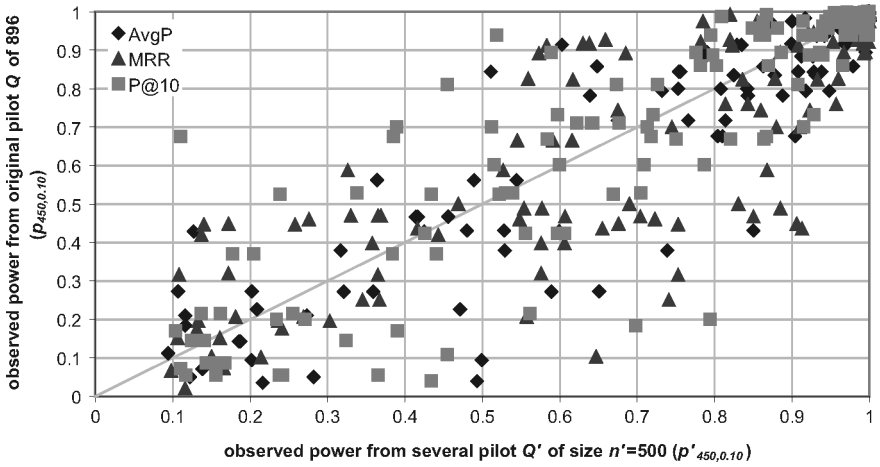

Fig. 4. Example discrepancies in bootstrap reproducibility probability estimation from varying pilot samples.

Table II. Minimum Reproducibility Probability from Pilot to Ensure Reproducibility Probability from All 896 of at Least 0.90

n'	AvgP	P@10	MRR
300	1.000	None	0.996
350	0.997	0.999	1.000
400	0.991	1.000	1.000
450	0.997	0.995	None
500	0.996	0.999	0.996
550	0.995	0.993	0.991
600	0.997	0.984	0.994
650	0.984	0.986	0.982
800	0.971	0.969	0.980

high sample estimates. As our metric decreases with larger sample sizes, the entire discrepancy graph continues to grow tighter to the diagonal. This analysis is a limited version of the conservative double bootstrap method proposed by De Martini [2006], which is computationally infeasible for our sample sizes. While such analysis could be performed on each engine pair individually, or by bucketing pairs by levels of difference, this creates the same dependencies that make error rate difficult to apply in new environments: defining the level of difference from unreliable preliminary values and an exaggerated dependence on the diversity of engines evaluated. Ensuring that none of the pairs of engines (especially those with small differences) yields a falsely high reproducibility probability estimate removes the dependence on determining levels of differences from small query sets. Rather than generating synthetic differences or engines, this analysis provides a minimum sample size that makes false positive errors unlikely for any new engine with similar score distribution in the given environment.

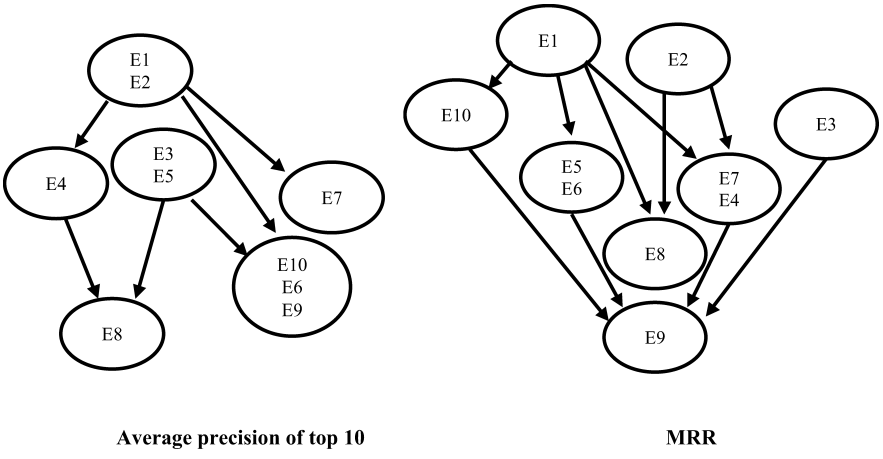

Fig. 5. Manual engine ranking with 99% reproducibility probability at $m = 850$.

In Table II, we detail this analysis for our web search evaluation, presenting the minimum $p'_{m=n'-50,0.10}$ from 20 varying pilot samples of size n' to ensure $p_{m=n'-50,0.10} \geq 0.90$ using all 896 as the pilot sample. For P@10 with samples of 300 queries and MRR with 450, even an estimate of 1.0 does not guarantee the estimate from all 896 is above 0.90 for the same candidate conclusion. Because of the margin of error for bootstrap estimates from a single pilot sample (which depends on B), minimums of 0.99 and above are difficult to enforce. With pilot samples of size 650, however, estimates begin to converge to ensure that high reproducibility probability from a sample corresponds to a high reproducibility probability estimate using all 896. Therefore, we conclude that 650 queries are necessary to estimate reproducibility probability reliably in our environment. Because this convergence takes place nearly 250 queries below the size of our entire sample of queries, we conclude that it is not an artifact of pilot sample size approaching that of our entire sample. This convergence takes place near the same size for each evaluation metric, leading us to hypothesize that the size of the pilot samples has more impact than the distributions under evaluation, and providing further evidence that even different engines would likely have reliable reproducibility probability estimates with this number of queries. We performed this same analysis on several TREC collections in Jensen [2006], finding conclusions difficult to generalize here, as such collections are not intended to represent a query population.

3.4 Conclusions From Manual Web Search Evaluation

Having established that the point-wise bootstrap estimate of reproducibility probability is reliable for high reproducibility probability estimates on large enough sample sizes, we conclude by applying it to our manual evaluation. The metric we chose for measuring reliability is also convenient for providing an ad hoc correction to our reproducibility probability estimates. While we are interested in conclusions with at least 90% reproducibility probability, we saw

that estimates from a pilot sample of even 800 queries must be above 98% to ensure this is valid for the population. To ensure we only examine reliable conclusions, therefore, we only include those with reproducibility probability of at least 99% for the remainder of our investigation. These benchmark high reproducibility probability conclusions from Wilcoxon tests using $\alpha = 0.10$ are shown in Figure 5. While many conclusions are significant based on a Wilcoxon test (see Figure 1), approximately half of these have high reproducibility probability.

4. SEMIAUTOMATIC EVALUATION

We have shown that evaluations in dynamic environments are capable of yielding conclusions that are reproducible across query samples. However, the sample sizes necessary to ensure this are large, demanding substantial effort to evaluate each query manually. To reduce the required manual judgment effort so that evaluations can feasibly be repeated as the environment changes, we propose a semiautomatic evaluation framework for integrating automatic judgments with manual ones. Whereas small numbers of manually evaluated queries are of little use on their own due to the large number of false positives we saw in Section 3, combining them with automatic evaluation provides insight into conclusions. Although any automatic evaluation technique; using implicit preferences such as clickthrough data, fusion or metasearch based approaches, and so on, could be applied in this framework, we leverage the resource-based approach we developed in prior work: mining pseudo-relevance judgments from taxonomies such as the Open Directory Project (referred to as DMOZ) [Chowdhury 2005]. This serves as both an analysis of the utility of our resource-based automatic evaluation technique, and more importantly, a vehicle for developing our semiautomatic framework and demonstrating how to apply and validate it.

First, we provide an overview of mining pseudo-relevance judgments from taxonomies and give conclusions derived from its automatic judgments alone. Next, we present the two basic ways in which automatic techniques can augment manual ones: by predicting conclusions that are likely to be found with larger query sets, by using a combination of a smaller number of manual judgments with automatic ones, and by filtering conclusions from small manual evaluations to improve their reliability. Finally, we present simple methods for leveraging each of these two aspects. We compare the reliable pairwise (engine A vs. engine B) conclusions they provide, with those drawn from our manual evaluation. Our analysis serves as an example of that which would be required using any automatic evaluation technique in a given environment, thus illustrating our framework and corresponding metrics for analyzing the utility of semiautomatic methods.

4.1 Mining Automatic Relevance Judgments

To validate our semiautomatic framework, we employ automatic evaluation techniques developed in our previous work that address both the informational and navigational tasks [Beitzel et al. 2003a; Chowdhury 2005; Jensen 2006]. These automatic techniques leverage two types of resources that are likely to

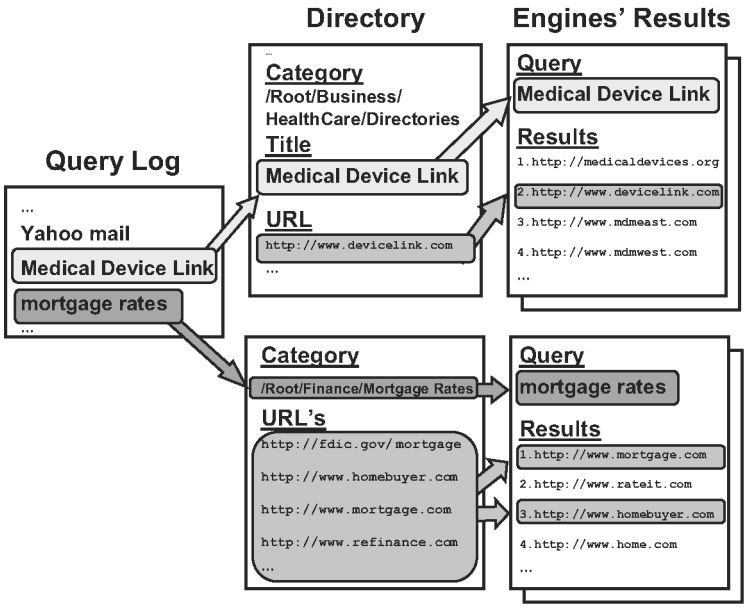

Fig. 6. Automatic techniques: Title-Match (top) and Category-Match (bottom).

be available in most dynamic search environments: a log sufficiently representing the population of queries, and a human-edited taxonomy of documents in the collection that is large enough to include a representative sample of the collection. This could be any form of taxonomy, such as a corporate intranet directory, Web taxonomy, or large collection of categorized bookmarks, but it must represent human matches of topics to documents and not be biased towards particular search services. Our initial investigations into automatic evaluation used the DMOZ and LookSmart taxonomies to show that on the Web these techniques are not biased towards particular engines by the choice of taxonomy to mine judgments from, finding a 0.931 Pearson correlation between MRR1 scores (the reciprocal rank of the first relevant result in the retrieved list) of automatic evaluations using each [Beitzel et al. 2003b; Chowdhury and Soboroff 2002]. These purely automatic techniques have correlations in the 0.7 range with manual evaluation scores [Beitzel et al. 2003a; Chowdhury 2005]. For the following experimentation, we repeated our automatic evaluations on the Web using more recent DMOZ data (downloaded on 12/8/2004) applying their judgments to queries from the same log and results from the same set of ten Web search engines as in our manual evaluation. Details of this process are provided in Appendix A.1.

An example of each technique is provided in Figure 6. For the navigational homepage/named page-finding task, we mine pseudo-relevance judgments using a technique we term Title-Match. It collects documents from the taxonomy whose editor-supplied titles exactly match a given query. These documents are treated as the "best" or "most relevant" documents for that query. For the informational, topical search task, we use a technique termed Category-Match. If

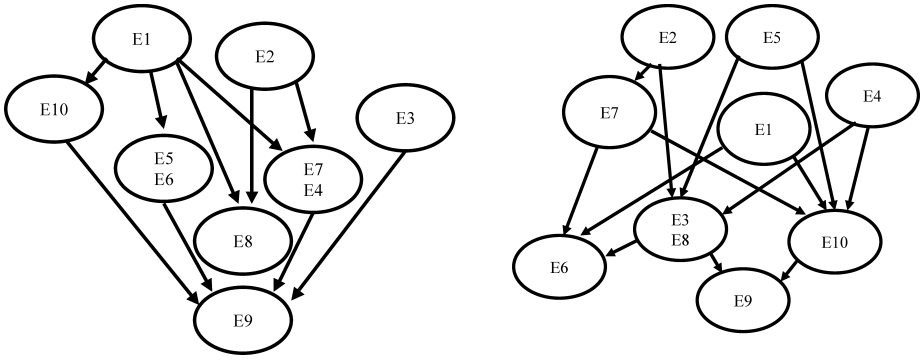

Fig. 7. Comparison of benchmark MRR manual conclusions (left) to purely automatic Title-Match MRR1 engine ranking (right) with 99% reproducibility probability at $m = 850$.

the most specific component of a category name exactly matches a given query, all documents from that category are used as the pseudo-relevant set.

Scores for the ten engines using these automatic techniques are available in Appendix A.1. As with manual evaluations, ranking engines by their average score and comparing rankings using correlations is insufficient. To compare only the reliable conclusions drawn from automatic evaluations with those from manual ones, we apply the same reproducibility probability analysis. Using the randomly selected Title-Matched queries as the pilot sample, and setting the bootstrap sample size equivalent to that of our manual evaluation, so that we would detect differences of comparable magnitude, we found those diagrammed in Figure 7. Comparing these conclusions with those of our manual evaluation in Figure 5 (duplicated for convenience), the automatic technique ranks E10 and E6 relatively lower, while it ranks E4 and E5 higher. Category-Match has a similar correlation, (see Appendix A.1). Although our focus is on demonstrating our framework, we investigated several methods of improving this correlation, including correcting for query popularity distribution, topical category distribution, and number of relevant results. None of these preliminary investigations substantially improved correlation [Jensen 2006].

4.2 Integrating Manual and Automatic Judgments

Although they are useful in examining evaluation characteristics over query sample sizes difficult to evaluate manually, we have seen that these purely automatic techniques are often inaccurate. We have also shown, in Section 3.3, that evaluation of search engines in dynamic environments demands a large query sample size even to estimate reproducibility probability. Incorporating automatic techniques with smaller numbers of manual judgments provides a sort of evaluation roadmap where there would otherwise have been little information about engines' relative performances. We focus on providing guidance for developing an intelligent evaluation strategy without having to manually evaluate the requisite number of queries for a reliable evaluation over every engine. We examine the two basic advantages semiautomatic methods can offer towards this goal: expanding the set of conclusions by predicting which will

Given:

- r_{man}, the desired average proportion of manual queries in each bootstrap sample such that the expected number of manually evaluated queries per sample is $E(m^*_{man}) = m * r_{man}$ and expected number of automatically evaluated ones is $E(m^*_{auto}) = m * (1 - r_{man})$ where $m = m^*_{man} + m^*_{auto}$ for each sample
- Manual relevance judgments for the engines to be compared over a small query sample Q_{man} of size n_{man}
- Pseudo-relevance judgments for the engines to be compared over a large query sample Q_{auto} of size $n_{auto} >> E(m^*_{auto})$

Estimate the reproducibility probability $p_{m,\alpha}$ of conclusions using a modified version of the bootstrapping procedure described in Figure 2 which at step 1 draws the sets Q^* of size m randomly from Q_{man} with probability r_{man} and Q_{auto} with $1 - r_{man}$

Fig. 8. Semiautomatic prediction.

have high reproducibility probability with more manual evaluation, and pruning the set of conclusions from a manually judged query sample by removing those that do not seem to be reproducible across samples of this size.

4.2.1 *Semiautomatic Prediction.* To aid evaluators in focusing on conclusions that are likely to be reliable with further manual evaluation, we propose the technique detailed in Figure 8. Although automatic and manual judgments could also be combined per-result rather than on a query-by-query basis, we hypothesized that evaluating only some of the results from a query is not dramatically less effort than evaluating all of a query's results.

We employ this probabilistic sampling rather than simply using the same, entire Q_{man} sample in each bootstrap replication to reduce false positives by increasing the diversity of the samples. We assume the number of queries with automatic judgments is much larger than that used in each bootstrap replication to prevent a large number of tied scores. The primary goal of the following experimentation is to determine the range of r_{man} and n_{man} parameters at which the semiautomatic method predicts more of the correct conclusions than simply using Q_{man} alone, while maintaining a relatively low probability of finding errant, false positive, conclusions.

4.2.2 *Semiautomatic Filtering.* To finalize conclusions from manually evaluated query samples too small to provide reliable conclusions on their own (removing the need for further judgments of the associated engines), we propose the technique detailed in Figure 9.

This technique leverages the large sample sizes possible using automatic techniques to reduce the likelihood that initial conclusions are simply artifacts of the insufficient manual sample size. For sizes n_{man} too small to yield reliable conclusions on their own (as discussed in Section 3.3), we hypothesize that filtering their conclusions with those from an automatic evaluation can reduce false positive errors enough to allow them to be accepted. The primary goal

Given:

- The minimum reproducibility probability $p_{\min}$ to draw a binary pairwise conclusion of the form $E_A > E_B$
- The set of candidate binary pairwise conclusions C_{man} for each engine pair of interest where $p_{m,\alpha} > p_{\min}$ based on manual relevance judgments for a query sample of size n_{man}
- The set of candidate binary pairwise conclusions C_{auto} for each engine pair of interest where $p_{m,\alpha} > p_{\min}$ based on pseudo-relevance judgments for a query sample of size n_{auto}

Set $C_{filt} = C_{auto} \cap C_{man}$

Fig. 9. Semiautomatic filtering.

of our experimentation with this technique is to determine the range of sizes n_{man} for which this effect is achieved, while not discarding too many of the conclusions from the purely manual evaluation that are actually correct.

4.3 Utility of Semiautomatic Evaluation

The primary goal of these semiautomatic methods is to make repeating evaluations feasible in large, dynamic environments. They address this by providing insight into conclusions before completing an evaluation of every engine's results over the entire query sample size required to ensure reliability. This enables the development of intelligent evaluation strategies that reduce manual effort by removing engines from an evaluation in progress. However, acceptable levels of error for making decisions such as discarding an engine, depend on factors specific to evaluation goals, making conclusions about total effort difficult to generalize. The level of investigation (are we trying to divide the best engines from the worst, or determine whether one of the top two is truly better than the other?), or even the relative efficiency, monetary cost, and so on, of the engines considered to be likely, determines whether we are willing to tolerate some false alarms or missed conclusions. This is outside the scope of comparing the relative utility of various semiautomatic techniques. Therefore, we focus only on the general utility of these semiautomatic techniques versus manual judgments at finding the correct pairwise $E_A > E_B$ conclusions using only a small pilot sample of manually evaluated queries. We quantify this utility by measuring the number of errant pairwise conclusions each of them yield and the number of correct conclusions they miss. This is a typical method of evaluating pairwise conclusions in filtering and categorization [Beitzel et al. 2004a; Manmatha et al. 2002]. Our motivation for focusing on binary pairwise conclusions themselves, as opposed to the underlying reproducibility probability estimates is twofold. First, we found in Section 3.3, that for reasonable sample sizes, only very high reproducibility probability estimates are reliable. Based on that analysis, throughout the following evaluation we only treat reproducibility probability estimates greater than 99% as asserting a conclusion. Second,

practitioners are likely more concerned with making errant conclusions, rather than the accuracy of actual values of reproducibility probability estimates. For the same reasons, we provide the raw counts of errors rather than their percentages, as the magnitude of number of errors is often of at least as much concern as their proportions. Unlike using only the correlation of engine rankings to compare evaluations, this framework focuses on conclusions with high reproducibility probability, accounting for ties, and exposing whether an evaluation is too weak to find correct conclusions, or too confident in errant conclusions.

Comparing evaluations is complicated by the need to define the "correct" conclusions. For example, if an evaluation of 300 queries finds that E_A outperforms E_B, and a larger evaluation of 800 queries finds the same thing, but if it also shows that 300 was not enough to reliably conclude that, is the conclusion $E_A > E_B$ based on the initial 300 queries "errant?". To mitigate these issues each of our analyses spans several benchmark query sample sizes (most easily characterized by the bootstrap sample size, m, since we vary the size of the pilot samples). Because our baseline is purely manual judgments, the following analysis also provides an interesting corollary to our investigation into the reliability of reproducibility probability estimates from manual judgments as it further describes the type of errant conclusions they cause.

4.3.1 *Results of Predicting from Auto-Manual Mixed Samples.* First, we evaluate the utility of the prediction procedure described in Section 4.2.1 against simply using the pilot sample of manually evaluated queries alone. In the task of predicting what conclusions will be found with larger query sample sizes than those that have been evaluated, we seek to determine the range of r_{man} (the ratio of manual to automatically judged queries) and n_{man} (the size of the pilot sample) parameters for which the semiautomatic procedure substantially reduces errors compared to the manual. As we did in Section 3.3, we analyze the manual method by finding the set of conclusions from each of 20 different distinct query samples, Q'_{man}, with 50 more queries than the size we bootstrap. With a mixture of automatically and manually evaluated queries in the semiautomatic method, the need for a larger pilot manual sample than the bootstrap sample size needed to prevent a large number of ties is diminished. To ensure a conservative evaluation, we therefore used sets Q'_{man} of size $n_{man} = E(m^*_{man})$ for the semiautomatic method.

We begin with an examination of the navigational evaluation, using the best page MRR manual evaluation and the Title-Match automatic approach. In Figure 10 we compare the correct set of manual conclusions based on our benchmark pilot of all 896 queries bootstrapped into sets of 850 (a copy of Figure 5 for convenience) to those from one of the twenty semiautomatic prediction runs. This is in fact the worst case (the largest number of missed conclusions) out of the twenty pilot Q'_{man} samples of size 350 for the $m = 850$, $E(m^*_{man}) = 350$ test. Comparing these example semiautomatic conclusions in Figure 10 to those of the purely automatic technique in Figure 7, shows that the same general discrepancies exist, but their severity is markedly decreased. The semiautomatic still ranks E10 and E6 relatively too low and E4 and E5 higher than the manual, just as the automatic method did. However, the number of errors is dramatically

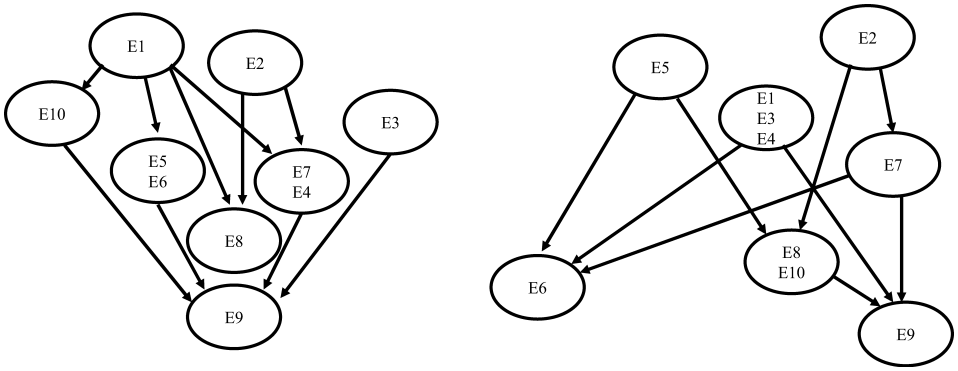

Fig. 10. Comparison of benchmark MRR manual conclusions (left) to worst misses case of semiautomatic prediction with $m = 850$ and $E(m^*_{man}) = 350$ (right).

Table III. Selected Sizes of Predicting from Combined Manual MRR and Title-Match

		Manual MRR				**Predicting with Combined Manual MRR and Title-Match**			
		False alarms		***Misses***		***False alarms***		***Misses***	
m	$E(m^*_{man})$	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*
300	0	N/A	N/A	N/A	N/A	14/16	14/16	1/3	1/3
300	200	1.30/3.15	6/8	1.15/3	3/3	4.45/6.70	8/10	0.75/3	1/3
450	200	0.50/3.15	3/7	7.35/10	10/10	6.15/13.60	10/18	2.55/10	4/10
600	200	0.15/3.15	2/7	10.00/13	13/13	9.90/18.35	14/24	4.55/13	5/13
850	200	0.05/3.15	1/7	12.90/16	16/16	13.85/24.25	16/26	5.60/16	6/16
850	350	0.25/6.55	2/11	9.70/16	14/16	9.80/21.00	12/23	4.80/16	7/16

fewer because it commits these infractions in only a small number of engine pairs, whereas the automatic method is certain of its incorrectness in many more cases.

This serves as an illustrative example of how the aggregated errors in the following tables, such as Table III, are counted. Recalling that any path from a higher node to a lower one implies that engine outperforms the lower one, each of these sets contain 16 distinct conclusions (by chance). As recorded in the final row of Table III, this case of the semiautomatic technique misses 7 of the 16 correct conclusions, the largest absolute number of them across all 20 pilot samples. The missed conclusions are:

- {E1, E2} > E4
- E1 > {E5, E7, E8, E10}
- E6 > E9

Of the 16 conclusions this case draws, 9 are false alarms (errant false positives):

- {E2, E5} > {E10, E6}
- {E3, E4, E7} > E6
- E5 > E8
- E8 > E9

The first column is the benchmark bootstrap sample size taken from the pilot of all 896 that we compare with both the small manual and semiautomatic. The expected number of manual queries in each test bootstrap sample for the semiautomatic approach is given in the second column. This is equivalent to the test bootstrap sample size for the purely manual approach, as we are interested in how well a small number of manually evaluated queries predict the conclusions of a larger number. The probability of a false alarm is expressed as the ratio of the average number of false alarms to the average number of conclusions drawn. The maximum absolute number of false alarms across all 20 runs is given with its associated number of conclusions on that pilot sample. The probability of a miss is based on the number of correct conclusions, which is constant for each benchmark m (the same for the manual and semiautomatic method). There is one special case, $E(m^*_{man}) = 0$, where a purely automatic approach is provided. That case, does not make use of any pilot manual samples so there is only a single result.

Table III includes selected rows where the semiautomatic approach reduces errors dramatically compared to the manual. Complete results for these and other ratios of manual to automatic results are provided in Appendix A.2. Predictions based on expanding the small manual sample with queries automatically evaluated using Title-Match typically miss approximately half as many of the correct conclusions as those from the manual sample alone. We examine predictions to four larger sizes: 300, to investigate our ability to predict dramatic differences with very few judgments, 450, the first point when high reproducibility probability estimates in the manual case begin to become reliable (see Table II), 600, where manual conclusions are reliable, and 850, the most detailed conclusions our set of judgments can support. The small number of correct conclusions (three) in the 300 queries case makes it difficult to choose one over the other as both the manual and semiautomatic methods have difficulty. The manual one often misses all three, while the semiautomatic one draws far too many conclusions in general, with over half of them being false alarms. Random performance, however, would draw nearly all false alarms, as only three conclusions of 45 are correct. Across the other prediction sizes, the manual method often misses nearly all the correct conclusions at a maximum; the semiautomatic often cuts this by half. Its number of false alarms, however, is greater than when using manual queries alone. This can be mitigated by incorporating a large enough ratio of manual queries (see Appendix A.2), which also reduces the number of conclusions it draws in general (the denominator of the probability of false alarm). Of course, larger available pilot samples for the manual method increase the number of conclusions it draws on average, subsequently decreasing the average number of misses with little increase in false alarms. When a larger number of manual judgments are available, the semiautomatic method may not be justified. Compared to not being able to draw any conclusions at all, even a prediction method prone to some degree of false alarms is likely useful; but how do we determine the bounds of this utility? Clearly, we need a combined metric to compare these two methods and determine when the semiautomatic method's relative benefits justify its use.

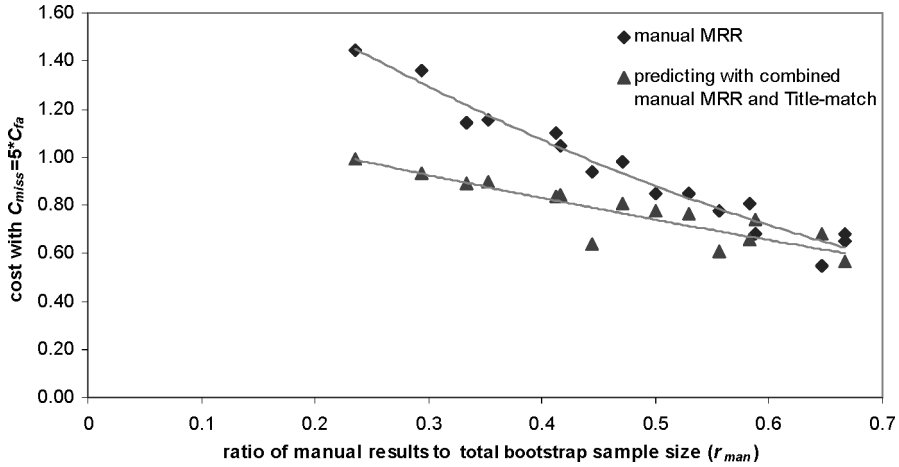

Fig. 11. Cost of errors in manual MRR and predicting with combination of Title-Match for $m =$ 450, 600, and 850.

To directly compare the cost of errors in the manual and semiautomatic methods, we leverage a standard cost function (Equation 3) adopted from the Topic Detection and Tracking (TDT) conference [Manmatha et al. 2002]. A lower cost indicates fewer errors were made. This combines the ratios of errors shown in the table with relative costs for each type of error, and normalizes them by the relative number of correct conclusions in general. In our calculation of *P(rel)*, we assume the maximum number of pair-wise conclusions that could be found among our 10 engines, with 45 as the denominator. Because the actual numbers of correct conclusions for our four prediction sizes are much less than 45, this may inherently provide extra weight to the false alarm errors.

$$\text{Cost} = C_{miss}P(miss)P(rel) + C_{fa}P(fa)(1 - P(rel))$$

Equation 3: The TDT cost function.

In the prediction task, we set $C_{miss} = 5 * C_{fa}$ to reflect the importance of finding correct conclusions over suggesting errant ones. With the cost of misses twice, or equal to, false alarms, the manual method typically outperforms the semiautomatic, although this may be inflated by the aforementioned bias from $P(rel)$. We hypothesized that the key parameter was the ratio of manually judged queries in the bootstrap samples, regardless of the overall magnitude of the sample. In Figure 11, we show the costs for the manual and semiautomatic methods at various ratios of manually evaluated queries to the predicted query sample size when predicting sizes of 450, 600, and 850. This, and each of the following cost graphs include trend lines for readability, created using a second order polynomial regression since that yielded the largest R^2 fitness measure for each graph. We do not intend to make any general assertions about the shape of such curves, as it is obvious they differ depending on the automatic technique used. Errors are very highly correlated to the ratio of manual judgments regardless of total sample size. When less than 50% of the sample size to be predicted has been manually evaluated, the semiautomatic technique is more effective

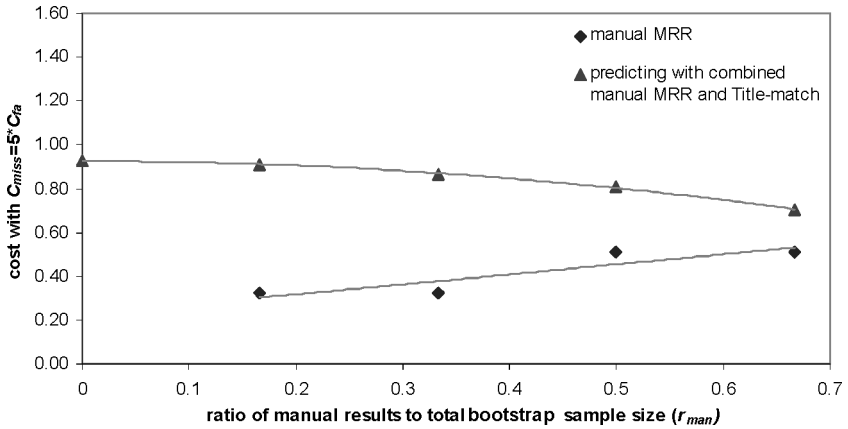

Fig. 12. Cost of errors in manual and predicting with combined Title-Match for $m = 300$.

at predicting conclusions than the smaller number of manually judged queries alone. When roughly half of the sample size to be predicted has been manually evaluated, the cost of false alarms introduced by the semiautomatic method outweighs the reduction in missed correct conclusions compared to using the manual sample alone.

We also hypothesized that conclusions with dramatically differing engines could be predicted with very few manual judgments. As we saw in the raw error counts of Table III, however, predicting conclusions at sample size 300 using the semiautomatic technique results in so many false alarms that its cost is higher than using the small manual sets alone, despite their propensity to miss many relevant conclusions (see Figure 12). When no manual judgments are available, however, the cost of errors from the purely automatic method is not terribly high. Again, it is likely to be useful compared to not being able to predict any conclusions whatsoever.

The proposed semiautomatic framework and metrics enable us to compare the effectiveness of different automatic judgment techniques, in the hopes of moving beyond these naïve ones. We performed the same experiments and analysis with combining the average precision at 10 manual judgments and Category-Match automatic judgments. The complete error counts are included in Appendix A.2. Errors in the semiautomatic informational evaluation are also very highly correlated with the ratio of manual results, as evidenced by Figure 13. As with Title-Match, predicting distant conclusions is more useful than nearer ones. However, integrating the Category-Match judgments does not offer as much benefit as those of Title-Match in the navigational evaluation. The number of false alarms does not decrease as quickly with larger ratios of manually evaluated queries, and the number of misses actually increases slightly; whereas it decreases with Title-Match. We believe this is because the Category-Match evaluation has more disagreement with the manual AvgP evaluation, causing the integration of more manual judgments to reduce the number of both correct and incorrect Category-Match predictions. Like the navigational evaluation, however, the manual samples alone often miss nearly all of the correct conclusions. Just as with the navigational evaluation, predicting

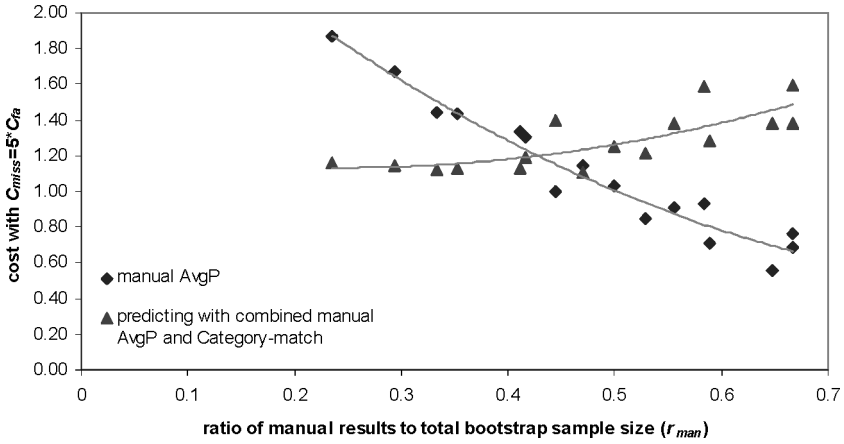

Fig. 13. Cost of errors in manual AvgP and predicting with combination of Category-Match for $m = 450$, 600, and 850.

Table IV. Filtering Conclusions from Small Manual MRR with Title-Match

		Manual MRR				**Manual MRR Filtered with Title-Match**			
		False alarms		***Misses***		***False alarms***		***Misses***	
m	n_{man}	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*
250	300	2.15/3.85	6/8	0.30/2	2/2	0.95/2.65	4/6	0.30/2	2/2
300	350	3.10/5.80	5/8	0.30/3	1/3	1.75/3.70	5/7	1.05/3	2/3
350	400	3.30/6.55	8/12	0.75/4	2/4	1.35/3.90	4/7	1.45/4	2/4
400	450	3.55/7.80	7/11	0.75/5	2/5	1.80/5.30	4/8	1.50/5	3/5
450	500	1.30/8.60	4/14	2.70/10	5/10	0.10/5.35	2/9	4.75/10	7/10
500	550	2.05/10.30	6/15	1.75/10	6/10	0.10/5.90	1/6	4.20/10	8/10
550	600	1.10/11.50	3/14	1.60/12	3/12	0.15/7.35	1/8	4.80/12	6/12
600	650	1.15/13.00	3/15	1.15/13	4/13	0.20/8.75	1/10	4.45/13	5/13

conclusions for small sample sizes such as 300, is better achieved with very few manual judgments than with the semiautomatic technique, due to the large number of false alarms (see Appendix A.2).

4.3.2 *Results of Filtering Conclusions from Small Manual Samples.* Next, we evaluate the utility of the filtering procedure described in Section 4.2.2 as opposed to simply using the manually evaluated queries alone. The intent here is to reduce the number of false alarms from sample sizes too small to ensure reliability (as per Section 3.3). We seek to determine the range of manually evaluated queries n_{man} for which the semiautomatic technique is beneficial. As in our analysis of prediction, we create 20 distinct query samples, Q'_{man}, of size $n_{man} = m + 50$, and compare the set of conclusions from each to that of bootstrapping our entire pilot sample of 896 into sets of size m. We begin with an examination of the navigational evaluation, using the best page MRR manual evaluation and the Title-Match automatic approach (see Table IV).

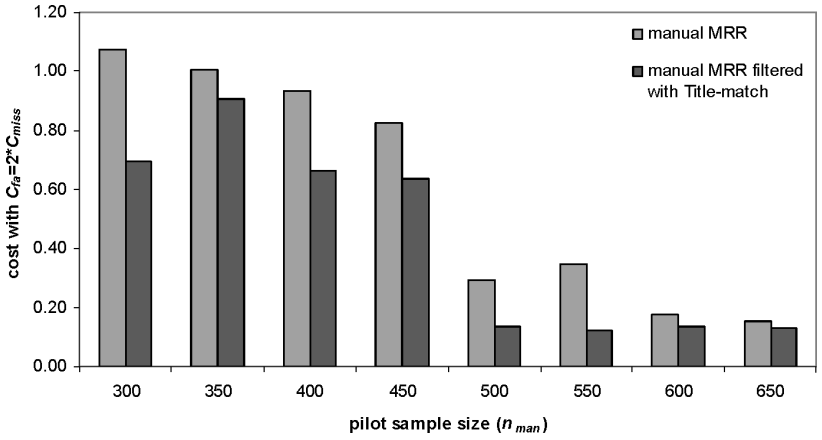

Fig. 14. Cost of errors in manual MRR and filtered with Title-Match.

Using the same metrics as in the previous section, it is clear from Table IV that semiautomatic filtering reduces false alarms by approximately half throughout the experiments, while not substantially increasing misses, especially the maximum number of them. At $n_{man} = 500$ there is a dramatic decrease in the number of false alarms. Interestingly, this correlates with the smallest size at which reproducibility probability estimates begin to become reliable across all metrics in Table II.

To compare the semiautomatic method to the manual, with a single metric, we again use the TDT cost function defined in Equation 3. In contrast to predicting conclusions, filtering increases reliability of candidate conclusions, so we set the cost of false alarms to be twice that of misses. With the costs set equal, the manual approach is preferred for some sample sizes. In Figure 14, we show the cost of the manual and semiautomatic methods at increasing sample sizes. Here, the steep drop in false alarms causes the corresponding total cost to drop dramatically with samples of size 500 and above. By 600, the costs are roughly equivalent, but filtering can still be useful to ensure the reliability of a conclusion to a stricter standard, as evidenced by the raw counts in Table IV.

The results for the informational search task and Category-Match automatic judgments are similar. Unlike the navigational evaluation, however, the number of false alarm and miss errors for the semiautomatic technique increases consistently with sample size. However, it still cuts the average number of false alarms by approximately half. Like the navigational evaluation, there is a drop in cost (see Figure 15) with samples of size 500 and above, but unlike it, the utility of filtering is also immediately diminished at that same point.

5. CONCLUSIONS

Dynamic environments such as the World Wide Web demand frequent repetition of costly search effectiveness evaluations. We have detailed a semiautomatic framework that combines automatic evaluation with manual judgments to make this feasible. We employ methods for comparing conclusions of one

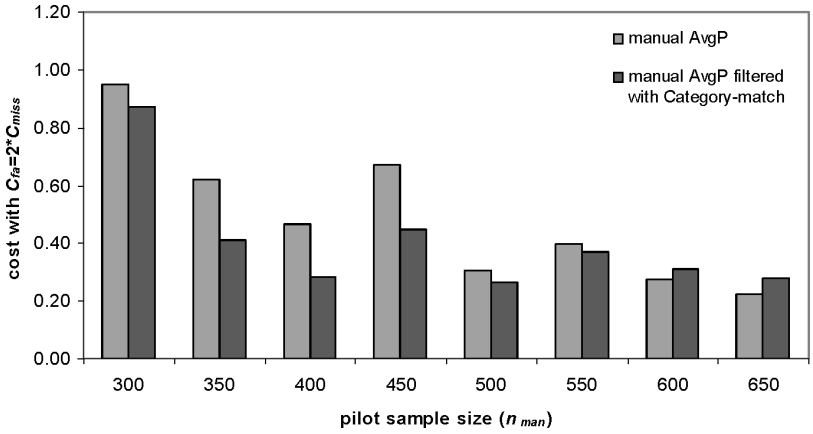

Fig. 15. Cost of errors in manual AvgP and filtered with Category-Match.

evaluation to another that go beyond simple correlation of engine rankings. Compared to small numbers of manually judged queries alone, semiautomatic prediction often reduces the number of missed correct conclusions by half, and semiautomatic filtering reduces the number of errant conclusions by half. This provides evaluators with insight into conclusions before naively evaluating every engine over the requisite number of queries for a reliable evaluation.

To validate this framework, we leveraged reproducibility probability to determine which conclusions generalize to the query population as a whole. Applying this method to our own precision-oriented manual Web search evaluation over 896 queries shows that the query sample sizes required to ensure reliability in such evaluations are often much larger than those previously studied (650 in our environment). Because precision-oriented evaluations are performed without system pooling, they do not depend on the number of engines being judged, enabling evaluation strategies that reduce effort by discarding poorly performing engines early. However, semiautomatic methods such as those proposed, are needed to exploit this by building query samples of sufficient size before manually evaluating each one. In a conservative example from our navigational evaluation, a combination of semiautomatic filtering and prediction using only 300 manually judged queries would enable us to reliably conclude that E6 and E9 are indeed the worst performing engines. Removing them from the evaluation would reduce the size of the result pools in the following 350 queries left to evaluate by 19% based on overlap analysis in Jensen [2006].

There is a great deal of future work in this area. Using this framework, we will evaluate and refine other automatic evaluation techniques, especially implicit preferences such as clickthrough data, to determine which (or what combination) best enables semiautomatic methods to determine the correct conclusions with fewer manual judgments. We will also further investigate manual judgment techniques for those that optimize the effort required to reach a desired level of reliability, such as judgments with varying levels of relevance beyond binary. In addition, each automatic evaluation technique has its own spamming issues that need to be investigated.

Table V. Automatic Mean Scores Using the 2004 DMOZ

Title-Match		Category-Match	
Ranking	***MRR1***	***Ranking***	***AvgP***
E2	0.605	E10	0.194
E5	0.602	E1	0.192
E4	0.601	E2	0.191
E7	0.582	E5	0.191
E1	0.573	E4	0.188
E3	0.569	E7	0.182
E8	0.548	E3	0.181
E10	0.523	E8	0.160
E6	0.476	E6	0.137
E9	0.428	E9	0.120

APPENDIX

A.1 Automatic Evaluation Statistics

We applied the techniques overviewed in Section 4.1 by matching the Web query log described in Section 3.1 to DMOZ data downloaded on 12/8/2004 containing 4,162,714 distinct URLs with a title entry. Effort was taken to mine the directories in the same timeframe as we crawled the engines results for both manual and automatic evaluation, to reduce the effect of changing content on our evaluations. For Title-Match, we paired documents whose DMOZ title exactly matched a query (ignoring only case) with that query. Human editors enter titles for the sites listed which, therefore, do not necessarily correspond to, and likely are more consistently accurate than, the titles of the pages themselves. In the 79% of DMOZ query-document pairs from 2003 that had URLs we were capable of crawling, only 18% had edited titles in the taxonomy that exactly matched (ignoring case) those of their corresponding pages [Beitzel et al. 2003a]. We filtered the initial set of matching query-document pairs such that we only kept pairs whose resulting URLs have at least one path component, not just a hostname, and for which the query does not appear verbatim in the URL. These constraints were intended to remove trivial matches such as the query "AOL" matching "http://www.aol.com" and limit bias that might be introduced if some engines use heuristics for matching URL text. Often, multiple documents in the taxonomies matched a given query, creating a set of alternate query-document pairs for that query. We treat each of these matches as a pseudo-relevant document. On the 172,111 queries we matched, there are an average of 1.32 pages matched per query. Average scores for a random sample of over 15,000 of these on the same ten engines as in our manual evaluation are provided in Table V.

For Category-Match, we paired documents in categories whose names exactly matched a query (ignoring only case). For efficiency reasons, we used a random subset of the queries that had matched titles using Title-Match as we hypothesized that these are more likely to match category names. We filtered out the "Adult," "World," "Netscape," and "Kids & Teens" subtrees of the DMOZ

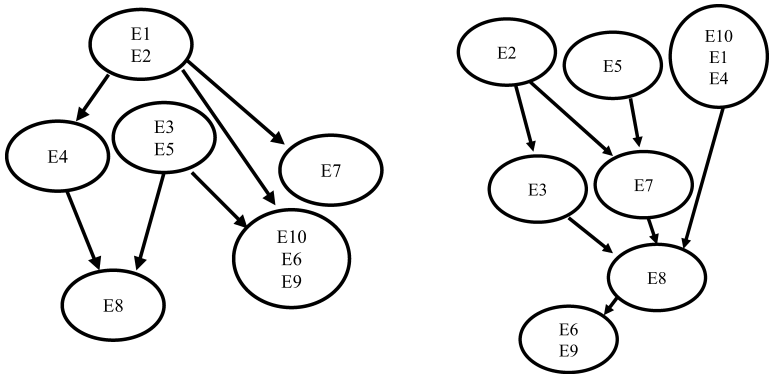

Fig. 16. Comparison of benchmark AvgP manual conclusions (left) to automatic Category-Match AvgP engine ranking (right) with 99% reproducibility probability at $m = 850$.

data because their editing policies differ from those of the rest of the directory. This left 356,537 distinct categories with at least one entry in them. Over the 12,911 queries we matched, there are an average of 70 relevant pages per query. Average scores over these matches are provided in Table V. Using these matched queries as our pilot sample, bootstrapping found the high reproducibility probability conclusions seen in Figure 16. Comparing these with those of the manual evaluation in Figure 5 (duplicated here for convenience), we see that the automatic technique is ranking E5 and E10 relatively higher, and is overly certain that E6 and E9 are the worst.

A.2 Complete Semiautomatic Evaluation Results

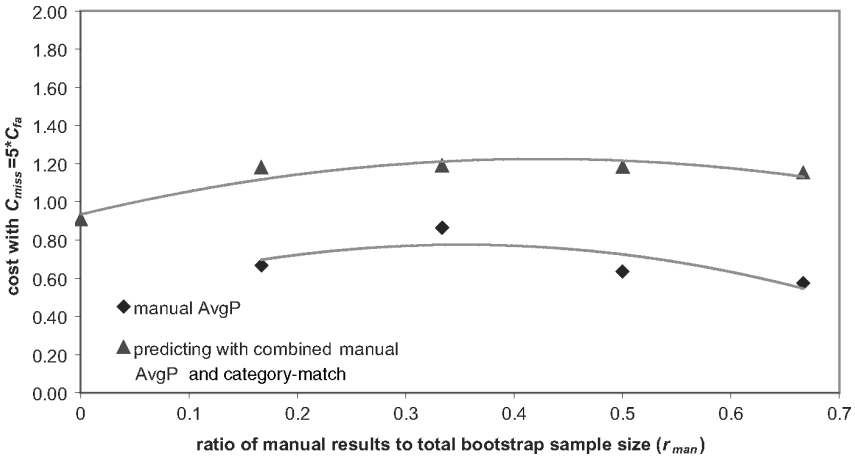

Fig. 17. Cost of errors in manual AvgP and predicting with combination of Category-Match for $m = 300$.

Table VI. Predicting Conclusions from Combined Small Manual MRR and Title-Match

		Manual MRR				**Predicting with Combined Manual MRR and Title-Match**			
		False alarms		***Misses***		***False alarms***		***Misses***	
m	$E(m^*_{man})$	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*
300	0	N/A	N/A	N/A	N/A	14/16	14/16	1/3	1/3
300	50	0.00/0.10	0/1	2.90/3	3/3	11.85/13.85	14/16	1.00/3	1/3
300	100	0.05/0.75	1/3	2.30/3	3/3	8.55/10.55	11/13	1.00/3	1/3
300	150	0.45/1.45	4/6	2.00/3	3/3	6.70/8.80	12/14	0.90/3	1/3
300	200	1.30/3.15	6/8	1.15/3	3/3	4.45/6.70	8/10	0.75/3	1/3
450	200	0.50/3.15	3/7	7.35/10	10/10	6.15/13.60	10/18	2.55/10	4/10
450	250	0.30/3.85	2/5	6.44/10	10/10	4.55/11.80	8/16	2.75/10	4/10
450	300	0.75/5.80	4/7	4.95/10	8/10	4.25/11.70	8/16	2.55/10	4/10
600	200	0.15/3.15	2/7	10.00/13	13/13	9.90/18.35	14/24	4.55/13	5/13
600	250	0.10/3.85	1/5	9.25/13	13/13	8.70/17.35	11/20	4.35/13	5/13
600	300	0.20/5.80	3/7	7.40/13	11/13	6.70/15.50	9/16	4.20/13	6/13
600	350	0.40/6.55	3/11	6.85/13	11/13	5.90/15.45	10/20	3.45/13	6/13
600	400	0.50/7.80	2/8	5.70/13	9/13	4.60/14.55	9/20	3.05/13	4/13
850	200	0.05/3.15	1/7	12.90/16	16/16	13.85/24.25	16/26	5.60/16	6/16
850	250	0.05/3.85	1/5	12.20/16	16/16	12.15/22.85	15/27	5.30/16	7/16
850	300	0.10/5.80	1/7	10.30/16	14/16	11.05/21.90	13/24	5.15/16	7/16
850	350	0.25/6.55	2/11	9.70/16	14/16	9.80/21.00	12/23	4.80/16	7/16
850	400	0.35/7.80	2/8	8.55/16	12/16	9.50/20.90	11/21	4.60/16	6/16
850	450	0.15/8.60	2/11	7.55/16	11/16	8.85/20.50	11/23	4.35/16	6/16
850	500	0.25/10.30	2/13	5.95/16	12/16	7.70/19.35	11/22	4.35/16	6/16
850	550	0.30/11.50	2/14	4.80/16	7/16	6.00/17.85	11/23	4.15/16	6/16

Table VII. Predicting Conclusions from Combined Small Manual AvgP and Category-Match

		Manual AvgP				**Predicting with Combined Manual AvgP and Category-Match**			
		False alarms		***Misses***		***False alarms***		***Misses***	
m	$E(m^*_{man})$	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*
300	0	N/A	N/A	N/A	N/A	15/19	15/19	2/6	2/6
300	50	0.00/0.00	0/0	6.00/6	6/6	13.00/15.10	13/15	3.90/6	4/6
300	100	0.20/0.70	2/5	5.35/6	6/6	12.25/14.25	13/15	4.00/6	4/6
300	150	0.20/1.40	2/6	4.60/6	6/6	11.45/13.45	12/14	4.00/6	4/6
300	200	0.55/2.80	3/9	3.65/6	6/6	8.30/10.25	9/11	4.05/6	5/6
450	200	0.00/2.80	0/0	9.00/12	12/12	10.85/15.05	11/15	7.80/12	8/12
450	250	0.25/3.90	2/7	7.75/12	11/12	10.30/14.55	11/15	7.75/12	8/12
450	300	0.50/6.50	1/12	5.70/12	9/12	9.30/13.45	10/14	7.85/12	8/12
600	200	0.00/2.80	0/0	13.00/16	16/16	11.00/20.10	11/20	6.90/16	7/16
600	250	0.10/3.90	1/7	11.60/16	15/16	10.90/19.45	11/19	7.45/16	8/16
600	300	0.05/6.50	1/12	9.25/16	13/16	10.55/18.55	11/19	8.00/16	9/16
600	350	0.05/7.40	1/11	8.35/16	14/16	9.75/15.15	10/15	10.55/16	11/16
600	400	0.35/9.40	3/16	6.65/16	11/16	9.30/14.65	10/15	10.65/16	11/16
850	200	0.00/2.80	0/0	16.80/20	20/20	11.80/23.85	13/25	7.95/20	8/20
850	250	0.05/3.90	1/7	14.95/20	19/20	10.40/22.40	12/24	8.00/20	8/20
850	300	0.00/6.50	0/12	12.90/20	17/20	10.00/22.10	10/22	7.90/20	8/20
850	350	0.00/7.40	0/13	12.00/20	18/20	9.25/21.25	10/22	8.00/20	8/20
850	400	0.20/9.40	1/9	10.20/20	15/20	8.90/21.05	9/21	7.85/20	8/20
850	450	0.10/11.30	1/14	7.60/20	13/20	9.00/20.30	9/20	8.70/20	9/20
850	500	0.30/13.85	2/14	6.25/20	10/20	8.90/19.65	9/19	9.25/20	10/20
850	550	0.45/15.20	2/12	4.85/20	10/20	8.75/18.65	9/19	10.10/20	11/20

Table VIII. Filtering Conclusions from Small Manual AvgP with Category-Match

		Manual AvgP				**Manual AvgP Filtered with Category-Match**			
		False alarms		***Misses***		***False alarms***		***Misses***	
m	n_{man}	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*	*Mean*	*Max*
250	300	1.95/3.90	6/7	0.75/3	2/3	0.60/1.35	3/3	1.95/3	3/3
300	350	2.20/6.50	6/12	1.55/6	4/6	0.85/4.15	3/7	2.55/6	4/6
350	400	1.90/7.40	6/13	1.35/7	5/7	0.60/4.60	2/7	2.85/7	5/7
400	450	3.60/9.40	9/16	1.05/7	3/7	1.25/5.45	5/10	2.65/7	5/7
450	500	2.00/11.30	4/16	2.10/12	5/12	0.85/7.65	3/11	4.60/12	7/12
500	550	3.45/13.85	7/17	1.50/12	4/12	1.60/8.85	4/12	4.65/12	6/12
550	600	2.70/15.20	6/20	1.30/14	5/14	1.30/9.55	5/14	5.55/14	9/14
600	650	2.30/16.35	5/20	1.85/16	4/16	1.05/10.30	3/13	6.65/16	8/16

ACKNOWLEDGMENTS

Special thanks to Daniele De Martini of the Università del Piemonte Orientale for his assistance. We would also like to thank the anonymous reviewers for their very thorough feedback. Thanks also to David D. Lewis, Shlomo Argamon, David Grossman, and Ian Soboroff whose discussions were helpful in shaping this work.

REFERENCES

Aslam, J., Pavlu, V., and Yilmaz, E. 2006. Statistical method for system evaluation using incomplete judgments. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

Aslam, J. A., Pavlu, V., and Savell, R. 2003. A unified model for metasearch, pooling, and system evaluation. In *Proceedings of the ACM Conference on Information and Knowledge Management*, 484–491.

Bacchetti, P. 2002. Peer review of statistics in medical research: The other problem. *Brit. Med. J. 324*, 1271–1273.

Beitzel, S. M., Jensen, E. C., Chowdhury, A., and Grossman, D. 2003a. Using titles and category names from editor-driven taxonomies for automatic evaluation. In *Proceedings of the ACM Conference on Information and Knowledge Management*.

Beitzel, S. M., Jensen, E. C., Chowdhury, A., Grossman, D., and Frieder, O. 2003b. Using manually-built Web directories for automatic evaluation of known-item retrieval. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

Beitzel, S. M., Jensen, E. C., Chowdhury, A., and Grossman, D. 2004a. Evaluation of filtering current news search results. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

Beitzel, S. M., Jensen, E. C., Chowdhury, A., Grossman, D., and Frieder, O. 2004b. Hourly analysis of a very large topically categorized web query log. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

Beitzel, S. M., Jensen, E. C., Chowdhury, A., Frieder, O., and Grossman, D. 2006. Temporal analysis of a very large topically categorized Web query log. *J. Amer. Soc. Inform. Sci. Tech. (to appear)*.

Blustein, J. and Tague-Sutcliffe, J. 1995. IR-stat-pak. In *Presented at the ACM Conference on Research and Development in Information Retrieval*.

Borlund, P. 2003. The concept of relevance in IR. *J. Amer. Soc. Inform. Sci. Tech. 54*, 10 (August), 913–925.

Boyan, J., Freitag, D., and Joachims, T. 1996. A machine learning architecture for optimizing Web search engines. In *Proceedings of the AAAI Workshop on Internet Based Information Systems*.

BUCKLEY, C. AND VOORHEES, E. M. 2000. Evaluating evaluation measure stability. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 33–40.

CARTERETTE, B., ALLAN, J., AND SITARAMAN, R. 2006. Minimal test collections for retrieval evaluation. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

CHO, J., GARCIA-MOLINA, H., AND PAGE, L. 2000. Efficient crawling through URL ordering. In *Proceedings of the World Wide Web Conference*.

CHOWDHURY, A. AND SOBOROFF, I. 2002. Automatic evaluation of World Wide Web search services. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 421–422.

CHOWDHURY, A. 2005. Automatic evaluation of Web search services. In Zelkowitz, M., Ed. *Advances in Computers*, Elsevier Academic Press.

CLARKE, C., SCHOLER, F., AND SOBOROFF, I. 2005. The TREC 2005 terabyte track. In *Proceedings of the The Text Retrieval Conference*, NIST.

COLLINGS, B. J. AND HAMILTON, M. A. 1988. Estimating the power of the two sample Wilcoxon test for location shift. *Biometrics 44*, 847–860.

CORMACK, G. V., PALMER, C. R., AND CLARKE, C. 1998. Efficient construction of large test collections. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 282–289.

CORMACK, G. V. AND LYNAM, T. 2006. Statistical precision of information retrieval evaluation. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

DAVIDSON, R. AND MACKINNON, J. G. 2000. Bootstrap tests: How many bootstraps? *Econometric Rev. 19*, 55–68.

DAVIDSON, R. AND MACKINNON, J. G. 2006. The power of bootstrap and asymptotic tests. *J. Econometrics 133*, 421–441.

DE MARTINI, D. AND RAPALLO, F. 2003. Calculating the power of permutation tests: A comparison between nonparametric estimators. *J. Appl. Stat. Sci. 11*, 109–120.

DE MARTINI, D. 2006. On the stability of statistical tests. In *Proceedings of the ASA Joint Statistical Meeting*.

DING, W. AND MARCHIONINI, G. 1996. Comparative study of Web search service performance. In *Proceedings of the ASIS 1996 Annual Conference*.

EFRON, B. AND TIBSHIRANI, R. J. 1993. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 379–381.

GOLDSTEIN, J., LAVIE, A., LIN, C.-Y., AND VOSS, C. 2005. Workshop: Intrinsic and extrinsic evaluation measures for MT and/or summarization. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics*.

GOODMAN, S. N. 1992. A comment on replication, p-values and evidence. *Stat. Med. 11*, 875–879.

HALL, P. AND MARTIN, M. A. 1988. On bootstrap resampling and iteration. *Biometrika 75*(4), 661–671.

HAVELIWALA, T., GIONIS, A., KLEIN, D., AND INDYK, P. 2002. Evaluating strategies for similarity search on the Web. In *Proceedings of the World Wide Web Conference*.

HAWKING, D., CRASWELL, N., THISTLEWAITE, P., AND HARMAN, D. K. 1999. Results and challenges in Web search evaluation. In *Proceedings of the World Wide Web Conference*, 243–252.

HOENIG, J. M. AND HEISEY, D. M. 2001. The abuse of power: The pervasive fallacy of power calculations for data analysis. *Amer. Statist. 55*(1), 19–24.

HOLLANDER, M. AND WOLFE, D. 1973. *Nonparametric Statistical Methods*. John Wiley and Sons.

JANSEN, B. J. AND SPINK, A. 2005. How are we searching the World Wide Web?: An analysis of nine search engine transaction logs. *Inform. Proc. Manag. 42*(1), 248–263.

JANSEN, B. J., SPINK, A., AND PEDERSON, J. 2005. A temporal comparison of altavista Web searching. *J. Amer. Soc. Inform. Sci. Tech. 56*(6), 559–570.

JENSEN, E. C., BEITZEL, S. M., CHOWDHURY, A., AND FRIEDER, O. 2005. A framework for determining necessary query set sizes to evaluate Web search effectiveness. In *Proceedings of the World Wide Web Conference*, 1176.

JENSEN, E. C. 2006. Repeatable evaluation of information retrieval effectiveness in dynamic environments. *Computer Science*, Illinois Institute of Technology, Chicago, 88. http://ir.iit.edu/~ej/jensen_phd_thesis.pdf

JOACHIMS, T., GRANKA, L., PAN, B., HEMBROOKE, H., AND GAY, G. 2005. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 154–161.

LEHMANN, E. 1986. *Testing Statistical Hypotheses*. Wiley, 150.

LIN, W.-H. AND HAUPTMANN, A. 2005. Revisiting the effect of topic set size on retrieval error. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

MANMATHA, R., FENG, A., AND ALLAN, J. 2002. A critical examination of TDT's cost function. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 403–404.

MILLER, R. G., Jr. 1981. *Simultaneous Statistical Inference*. Springer, New York.

MUNZEL, U. 2001. A unified approach to simultaneous rank test procedures in the unbalanced one-way layout. *Biomet. J. 43*(5), 553–569.

NTOULAS, A., CHO, J., AND OLSTON, C. 2004. What's new on the Web? The evolution of the Web from a search engine perspective. In *Proceedings of the World Wide Web Conference*.

NURAY, R. AND CAN, F. 2006. Automatic ranking of information retrieval systems using data fusion. *Information Processing & Management 42*(3), 595–614.

PASS, G., CHOWDHURY, A., AND TORGESON, C. 2006. A picture of search. In *Proceedings of the International Conference on Scalable Information Systems (to appear)*.

SAKAI, T. 2006. Evaluating evaluation metrics based on the bootstrap. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

SANDERSON, M. AND JOHO, H. 2004. Forming test collections with no system pooling. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

SANDERSON, M. AND ZOBEL, J. 2005. Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

SAVOY, J. 1997. Statistical inference in retrieval effectiveness evaluation. *Information Processing & Management 33*(4) (July), 495–512.

SAVOY, J. AND PICARD, J. 2001. Retrieval effectiveness on the Web. *Inform. Proc. Manag. 37*(4) (July), 543–569.

SHANG, Y. AND LI, L. 2002. Precision evaluation of search engines. *World Wide Web 5*, 159–173.

SHAO, J. AND CHOW, S.-C. 2002. Reproducibility probability in clinical trials. *Statistics in Medicine 21*(12), 1727–1742.

SOBOROFF, I., NICHOLAS, C., AND CAHAN, P. 2001. Ranking retrieval systems without relevance judgments. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

Soboroff, I. 2006. Dynamic test collections: Measuring search effectiveness on the live Web. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*.

SPIEGELHALTER, D. J. AND FREEDMAN, L. S. 1986. A predictive approach to selecting the size of a clinical trial, based on subjective clinical opinion. *Statistics in Medicine 5*, 1–13.

SRINIVASAN, P., MENCZER, F., AND PANT, G. 2005. A general evaluation framework for topical crawlers. *Information Retrieval 8*(3), 417–447.

TAGUE-SUTCLIFFE, J. M. 1996. Some perspectives on the evaluation of information retrieval systems. *J. Amer. Soc. Inform. Sci. Tech. 47*(1) (Jan.), 1–3.

TROENDLE, J. F. 1999. Approximating the power of wilcoxon's rank-sum test against shift alternatives. *Stat. Med. 18*(20) (Oct.), 2763–2773.

VAN-RIJSBERGEN, C. J. 1979. Chapter 7. In *Information Retrieval*. Butterworths, 178–180.

VOORHEES, E. M. 1998. Variations in relevance judgments and the measurement of retrieval effectiveness. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 315–323.

VOORHEES, E. M. AND BUCKLEY, C. 2002. The effect of topic set size on retrieval experiment error. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 316–323.

WU, S. AND CRESTANI, F. 2003. Methods for ranking information retrieval systems without relevance judgments. In *Proceedings of the ACM Symposium on Applied Computing*.

ZOBEL, J. 1998. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the ACM Conference on Research and Development in Information Retrieval*, 307–314.

Received March 2006; revised October 2006, March 2007; accepted March 2007