Published in final edited form as:

ACM Trans Inf Syst. 2017 September ; 36(2): . doi:10.1145/3086701.

Using Replicates in Information Retrieval Evaluation

ELLEN M. VOORHEES, **DANIEL SAMAROV**, and **IAN SOBOROFF**

National Institute of Standards and Technology

Abstract

This article explores a method for more accurately estimating the main effect of the system in a typical test-collection-based evaluation of information retrieval systems, thus increasing the sensitivity of system comparisons. Randomly partitioning the test document collection allows for multiple tests of a given system and topic (replicates). Bootstrap ANOVA can use these replicates to extract system-topic interactions—something not possible without replicates—yielding a more precise value for the system effect and a narrower confidence interval around that value. Experiments using multiple TREC collections demonstrate that removing the topic-system interactions substantially reduces the confidence intervals around the system effect as well as increases the number of significant pairwise differences found. Further, the method is robust against small changes in the number of partitions used, against variability in the documents that constitute the partitions, and the measure of effectiveness used to quantify system effectiveness.

Additional Key Words and Pharses

Information retrieval; statistical analysis; test collections; topic variance

CCS Concepts

Information systems → Evaluation of retrieval results

1 INTRODUCTION

Test collections are an important tool in the development of effective information retrieval systems [26, 32]. Test collections consist of a set of documents, a set of information-need statements called *topics*, and relevance judgments that say which documents should be retrieved for which topics. A retrieval system creates a *run* containing a ranked list of documents for each topic, such that each list is sorted by decreasing likelihood that the document should be retrieved for that topic. An evaluation measure is computed for each topic based on the ranks at which the relevant documents appear. The overall score for the run is calculated as the mean of the individual topic scores. Retrieval system A is considered

Authors' addresses: E. M. Voorhees and I. Soboroff, NIST, 100 Bureau Drive, STOP 8940, Gaithersburg, MD 20899-8940; {ellen.voorhees, ian.soboroff }@nist.gov; D. Samarov, NIST, 100 Bureau Drive, STOP 8980, Gaithersburg, MD 20899-8980; daniel.samarov@nist.gov.

to be better than system B if the mean score for the run produced by A is sufficiently better than the mean score for run B, where "sufficiently" is frequently determined by a statistical hypothesis test.

This basic experimental framework was introduced by Cleverdon and his colleagues in the Cranfield experiments [12], and the retrieval literature is full of debates on how best to apply it: which measures [7, 8, 16, 19, 20], which statistical tests [15, 28–30], and how best to build good test collections [25, 31, 34].[1] Other work has looked at the statistical validity of the framework. Savoy proposed using medians, rather than means, as the summary statistic because of the nature of information retrieval data [29]. Cormack and Lynam argued that no summary statistic is suitable, since each individual topic is essentially its own experiment [13]; they suggest that instead individual topic results should be combined using techniques drawn from meta-analysis. Carterette warns against the common practice of not correcting for multiple comparisons when performing tests as well as shows that any difference in mean scores can appear significant with an arbitrarily large confidence level by using sufficient numbers of topics [11]. More recently, he proposed replacing generic statistical hypothesis tests with an information-retrieval-specific method based on tests that directly model relevance (as opposed to indirectly modeling relevance through effectiveness measures) [9]. In light of these problems with statistical hypothesis tests, Sakai challenged the retrieval research community to focus instead on effect sizes and confidence intervals [23].

The goal of the test collection methodology is to be able to make reliable conclusions regarding the role the differences in retrieval systems have on the difference in retrieval evaluation scores, that is, to measure the *system effect*. The difficulty is that evaluation scores vary for more reasons than simply the differences in retrieval systems. Banks and colleagues showed that in addition to the system effect, the topic effect (i.e., which particular topics the test collection happens to contain) as well as the interaction between the system and topic effects (i.e., different systems find different topics relatively harder or easier) were not only significant but were often larger than the system effect [3]. They also noted that the primary limitation in exploiting the interaction effect to improve system comparisons is the lack of replicate measurements of system performance for each topic—a given system is run only once per topic on a given document collection.

Since then there have been efforts to accommodate topic variability to improve test-collection-based retrieval evaluation. One approach is to use *standardized scores*, a technique that normalizes a run's score for each topic by the mean score obtained for that topic over a set of runs, which has the desired result of reducing the effect of topic variability in system comparisons [24, 33]. Carterette and colleagues used a mixed-effect model to account for variance due to both the topic sample and an effect they called the user effect, which represents differences in how patient different users are with respect to finding relevant documents [10]. Bailey and colleagues characterize a similar sort of user effect to make recommendations regarding test collection design [2] but do not incorporate that variance within significance testing. In work that is most similar to this article, Robertson and

[1]The given references are representative but by no means exhaustive.

Kanoulas produced multiple simulated measurements per run-topic combination and used the replicate measurements in a mixed-effects model to test for statistically significant differences between system pairs [22]. Under the assumption that a test collection contains both a sample of topics from a larger universe of topics and a sample of documents from a larger universe of documents, they separately modeled the distribution of relevant documents and the distribution of nonrelevant documents contained within a ranked list for each topic and run using a probability distribution. Once the system-topic behavior was modeled, they sampled from that probability distribution to get the desired replicate measurements; these simulated measurements represent scores that might have been observed from that system had the document sample within the test collection been different. They used the replicate measures to implement different types of significance tests: using a standard *t*-test on the mean of the replicate measures, and deriving *p*-values for the likelihood that the runs were different from either a heteroscedastic or a homoscedastic mixed-effects model. They concluded that the power of all the tests was comparable in that each test found roughly the same number of significantly different run pairs, though the particular pairs found to differ changed somewhat for the different tests.

The work reported on in this article has the same goal as the Robertson and Kanoulas work, namely to use replicate measurements to model system-topic interactions and thus increase the sensitivity of system comparisons, but uses a different approach to accomplish that goal. We obtain replicate measurements by randomly partitioning the actual document set and measuring system performance on each partition. We use the replicate measurements to build a bootstrapped analysis of variance (ANOVA) model to estimate system, topic, and interaction effects; use the model to build false discovery rate (FDR) corrected confidence intervals on the estimated system effects; and finally, we use the confidence intervals to determine statistical significance of run differences. The next section shows that this procedure produces confidence intervals on system effect that are substantially tighter than confidence intervals that do not incorporate system-topic interactions. Section 3 describes the methodology used to find statistically different system pairs and shows that the procedure finds more significantly different pairs than do other current tests. In Section 4, we apply the procedure to different test collections as well as vary the number of replicates on which it is based to test the procedure's robustness. These experiments show that the procedure is insensitive to small changes in the number of replicates and that the reduction in error estimates holds for collections of varying size, different document genres, and a variety of evaluation measures. Since the methodology incorporates a bootstrap model, it has a stochastic component. We investigate the effect of different starting configurations in Section 5 and show that different initial partitions can, in fact, lead to different decisions (to reject or not to reject the null hypothesis) for a given run pair. This suggests that the partitioning process itself should be repeated several times with a final decision based on an aggregation of the different partitionings' decisions. The most conservative aggregation policy of rejecting the null hypothesis only if the hypothesis would be rejected using each of the initial configurations still finds more statistically different run pairs than current tests. Finally, preliminary experiments described in Section 6 suggest the methodology can be used with just a few related runs, such as the set of runs an individual research team would produce.

2 METHODOLOGY USING REPLICATES

The goal of test-collection-based comparative evaluation is to distinguish more effective systems from less effective systems. Isolating the system effect from other sources of variation allows finer distinctions between systems to be drawn, but doing so requires a means by which the system, topic, and system-topic interaction effects can be estimated.

Consider the example in Figure 1 where the average precision (AP) scores for three systems are shown for three topics, and assume we will compare system effectiveness using analysis of variance (ANOVA) [11]. A one-way ANOVA model for just the system effect is

$$y_{ij} = \mu + s_i + \varepsilon_{ij}, \quad (1)$$

where y_{ij} is the AP score, μ is the overall mean, s_i, $i = 1,\ldots, 3$ is the system effect variable, $j = 1,\ldots,3$ is the index for topic, and $\varepsilon_{ij} \sim \mathcal{N}(0,\sigma)$ are errors (assumed to be normally distributed). A pairwise-comparison between systems using the output from this model will be dominated by the variability in AP scores resulting from the dip in performance across systems on topic 3. Using both system and topic effects (denoted by t_j) gives a two-way ANOVA model of the form

$$y_{ij} = \mu + s_i + t_j + \varepsilon_{ij}. \quad (2)$$

Including the topic effect allows the model to capture the dip in performance, resulting in a better estimate of the model error and thereby affording more reliable inference in comparing system performance. (Traditional IR evaluation methodology uses *paired* tests of significance as a means of capturing the topic effect.)

To see why the inclusion of the topic effect results in a more reliable model, it is helpful to look at the decomposition of the different sources of variability and how they impact the estimates of the model standard error. Let $SS_M(E)$ denote the sum-of-squares error (SSE) for model $M \in \{A,B\}$, where A is the one-way model of Equation (1) and B is the two-way model of Equation (2), and let $SS(\mathcal{S})$ be the sum-of-squares for the $\mathcal{S}$th source, $\mathcal{S} \in \{Total, System, Topic\} = \{TOT, S, T\}$. The model standard error is defined as $\sigma_M = \sqrt{SS_M(E)/df_M}$, where df_M is the appropriate error degrees of freedom for model M. For a fully balanced design—which we assume throughout the article—$SS_A(E) = SS(TOT) - SS(S)$ and $SS_B(E) = SS(TOT) - SS(S) - SS(T)$. Since $SS(TOT)$ and $SS(S)$ are the same for both model A and B and $SS(T)$ is non-negative, $SS_B(E)$ can be no larger than $SS_A(E)$ and will be smaller to the extent that the topic predicts the score.

Banks and colleagues showed that the system-topic interaction effect is another significant source of variability in search results [3]. An example of such an interaction is shown in Figure 1 for system C on topic 2. While system C generally performs worse than both systems A and B, the relative performance of C on topic 2 suggests it might have particular

issues with that topic. We could thus reduce the SSE even further if we incorporate a system-topic interaction term into the model, but to be able to estimate a model of the interaction effect, we need replicate measurements.[2] This section describes our technique for first obtaining the necessary replicate measurements and then using them to estimate system, topic, and system-topic interaction effects.

2.1 Random Partitioning

Our approach to obtain the required replicate measurements is to split an existing test collection into partitions and to evaluate partitioned runs on each of the document set partitions. This partitioning results in multiple scores per topic-run combination of the original collection.

More concretely, the document set is split into n_p partitions by considering each document in the collection in turn and rolling a n_p-sided die to determine the partition to which that document belongs. The document set partitions then induce partitions on the relevance judgments (called *qrels* in the rest of the article) and runs by restricting the original qrels/run to just the documents in the given document partition. That is, the run partition for document partition D is just the original run minus all documents not in D. The retrieval system that produced the original run is *not* re-run on the individual partitions.

Each run partition is evaluated using the corresponding qrels partition, with one score per topic being produced for each partition. With n_p partitions, we obtain n_p scores for each of the topics for each original run.

The critical assumption here is that per-topic scores computed from partitions are representative of a run's per-topic scores in the original collection. For random assignment of documents to partitions and small n_p, this is likely to be true: the smaller the number of partitions, the more similar each document partition is likely to be to the original. The use of partitioned runs also has empirical support from the work by Sanderson and colleagues in their investigation of subcollections [27]. They compared AP scores computed from partitioned runs (i.e., no re-running) to the AP scores computed from running the corresponding retrieval system on just the subcollection for nine different retrieval system configurations using the TREC-8 ad hoc collection. They found the AP scores matched very closely, achieving a Pearson correlation of 0.995.

An additional consideration is the split of the relevant documents among the partitions. As we noted above, we are assuming a fully balanced design, so we must have the same number of scores per partition. Many IR measures are undefined when a topic has no relevant documents, and there is no good choice to use as a "default" value.[3] To maintain a balanced design and avoid the issues surrounding undefined values, we enforce that all topics have relevant documents in all partitions in this work.

[2]There are methods such as Tukey's and Mandel's methods that do not require replicates that can indicate whether interaction effects exist, but these methods cannot capture the effects themselves without replicates.

[3]The lack of a good choice is not specific to our work here but true for IR evaluation in general. The usual suggestion is one of the extreme values the measure can take on, generally 0 or 1, but these values then dominate the means and thus put the evaluation emphasis on the topics for which there is no real data.

Robertson showed that the number of relevant documents is a primary factor in the statistical precision of a retrieval measurement [21], so having roughly the same number of relevant documents in each partition might increase the fidelity of the partitioned scores to the full-collection scores. One way of accomplishing an even split is round-robin assignment of relevant documents to partitions followed by random assignment of non-relevant documents. However, such a scheme invalidates the claim of random assignment of documents to partitions. Requiring that each partition has at least one relevant document for each topic also means that not all partitions of the original document set are equally likely, but it affects many fewer partitionings than round-robin assignment and should have little effect in practice. In the case that a particular generated partitioning contains a partition with no relevant documents for some topic, we do not use that partitioning and simply generate an entirely new partitioning.

2.2 Bootstrap ANOVA

The partitioning process is the means we use to get replicate scores for all topic-run combinations in the original data set. With replicate scores, we can learn a model that includes terms for the system effect, the topic effect, and the system-topic interaction effect:

$$y_{ijk} = \mu + s_i + t_j + (st)_{ij} + \varepsilon_{ijk}, \quad (3)$$

where μ denotes the "grand" mean, s_i the system effect, t_j the topic effect, and $(st)_{ij}$ the system-topic interaction effect. Correspondingly, $i = 1,\ldots,n_s$, $j = 1,\ldots,n_t$, and $k = 1,\ldots,n_p$ denote the indices for system, topic, and partition (i.e., replicates), respectively.

Since each run contains results for all topics (and we have relevant documents in each partition), we have a fully balanced design. Therefore, the solution to each of the parameters in the model has a closed form solution when using a least-squares framework [17]. Let

$$\bar{y}_{\ldots} = \frac{1}{n_s n_t n_p} \sum_{ijk} y_{ijk},$$
$$y_{i..} = \frac{1}{n_t n_p} \sum_{jk} y_{ijk},$$
$$\bar{y}_{.jk} = \frac{1}{n_s} \sum_{i} y_{ijk},$$

where $\bar{\cdot}$ denotes a mean and the "dots" in the subscript correspond to indices over which we are taking the mean, for example, $\bar{y}_{1..}$ indicates the mean of system 1 across all topics and partitions. Similar notation is used for other combinations of the subscripts *i*, *j*, and *k*. Then our parameter estimates can be expressed as

$$\begin{aligned}\hat{\mu} &= \bar{y}\ldots, \\ \hat{s}_i &= \bar{y}_{i..} - \bar{y}\ldots, \\ \hat{t}_j &= \bar{y}_{.j.} - \bar{y}\ldots, \\ \widehat{(st)}_{ij} &= \bar{y}_{ij.} - \bar{y}_{i..} - \bar{y}_{.j.} + \bar{y}_{...},\end{aligned} \tag{4}$$

and the predictions are

$$\hat{y}_{ijk} = \hat{\mu} + \hat{s}_i + \hat{t}_j + \widehat{(st)}_{ij}.$$

where $\hat{\cdot}$ denote the corresponding parameter estimates.

To determine parameter significance, we use a bootstrap-based model described below [14]. Our motivation for *not* using a standard ANOVA-based approach for inference (i.e., for not using t and F tests to determine parameter and model significance) is to minimize the assumptions we impose on the distribution of the data. This is partially motivated by the fact that our responses fall between 0 and 1 (for all of the evaluation measures we consider), and assuming normal or near-normal behavior is not appropriate.[4]

Our bootstrap model uses the residuals generated from a least-squares fitting of the data as the basis for the sampling scheme. Fixing the predictions $\hat{y}_{ijk}$, we perform $N = n_s \times n_t \times n_p$ random draws (with replacement) of $r_{ijk}^{(m)} \in \{r_{ijk}\}_{i,j,k=1}^{n_s, n_t, n_p}$ from the residuals of the initial ANOVA model fit where $m = 1,\ldots,M$ is the mth of M total random samplings (this is agnostic to topic/system). Our bootstrap sample is generated as $y_{ijk}^{(m)} = \hat{y}_{ijk} + r_{ijk}^{(m)}$, for $i = 1, \ldots, n_s$, $j = 1,\ldots,n_t$, and $p = 1, \ldots, n_p$. For each of the M draws, these $y_{ijk}^{(m)}$'s are used to compute the parameters' estimates. In other words, each bootstrap sample randomly assigns the residuals of the initial ANOVA model fit across the run-topic combinations and computes a new set of estimates of the model parameters using those values. The bootstrap samples from only the residuals in each iteration.

The result of the bootstrap process is a set of M different estimates of the model parameters. These estimates in turn provide an estimated probability density function of the mean value of a parameter. In particular, we have M different estimates of the system effect and thus an estimated probability density function for the mean system effect for each run. With a probability density function, we can build a confidence interval for the value of the system effect (details of how we build the intervals are given in the next section). Figure 2 shows 95% confidence intervals of system effect computed using models with and without system-topic interaction effects for the TREC-3 ad hoc task data set used by Banks and colleagues. The test collection consists of about 750,000 mostly newswire documents and 50 topics, and

[4]QQ-plots of the residuals using a standard ANOVA approach showed "heavy-tailed" behavior indicative of non-normality. Basing inferences on an assumption of underlying normality could yield unreliable results. We found similar behavior in the residuals using various other transformations on the response as well other other model parameterizations (e.g., beta regression).

the relevance judgments were constructed from the 40 runs submitted to the task. System effect for each TREC-3 run evaluated using AP is plotted in the top of the figure and evaluated using P10 is plotted in the bottom of the figure. Confidence intervals computed using the model without system-topic interactions (i.e., the model of Equation (2)) are plotted in black. Intervals computed using the model with interactions (Equation (3)) are plotted in red. In each case the model parameters were estimated using $M = 10{,}000$ bootstrap iterations and $n_p = 3$ document set partitions. The range of estimates of $\hat{s}_i$ computed by the model without interactions is clearly larger than the range of estimates computed by the model with interactions, since the red intervals are strictly contained within the black intervals for all runs and both measures. As discussed above, this is because the model without the interaction effect has more unexplained variability so its residual error terms are larger. Defining the length of a confidence interval to be *upperBound–lowerBound*, the mean, shortest, and longest confidence interval lengths over the 40 TREC-3 runs are given in Table 1.

The fact that incorporating the system-topic interactions substantially reduces the size of the confidence interval of the system effect re-confirms the significance of the system-topic interaction effect for search results. AP-based confidence intervals are smaller than P10-based intervals, because AP is an inherently more stable (less variable) measure than P10 [6]. Tighter bounds on estimates of the system effect means that comparisons between systems are more sensitive.

3 DISTINGUISHING AMONG SYSTEMS

Our motivation for obtaining more precise estimates of the system effect in IR experiments is to improve our ability to distinguish more effective systems from less effective systems. In this section, we describe how to infer the likelihood that systems are different.

3.1 Computing *p*-values

A result of the bootstrap process is an estimated probability density function of the mean score for each run. A *p*-value for a run pair is the probability of the observed difference in scores, or one more extreme, when assuming that the two systems are actually equally effective. These *p*-values can be computed directly from the estimated probability density functions as shown in Figure 3.

However, because we are comparing many runs at once, we need to appropriately adjust the *p*-values for multiple comparisons when using them for the purpose of inference. There are a variety of options for performing such corrections; both References [5] and [11], for example, discuss the problem in the context of information retrieval. We focus on procedures that control for false discovery rate (FDR), as opposed to family-wise error rate (FWER). The primary difference between the two approaches is that FWER tries to reduce the probability that even one false discovery is made while FDR controls the proportion of expected false discoveries. FWER-based procedures tend to be highly under-powered, especially when a large number of hypotheses are being tested. FDR-based procedures tend to have greater power to detect differences at the cost of increased false positives. Since we

are dealing with $N = \binom{n_s}{2}$ tests, where the number of runs in a typical TREC data set (n_s) is at least several dozen, an FDR-based approach is a good fit.

The particular FDR correction we use is the Benjamini-Hochberg (BH) correction [4]. The input to the correction procedure is the set of N (the number of system pairs) p-values computed directly from the estimated probability density functions and sorted from smallest to largest: $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(N)}$. Let α be the target level of significance, that is, the probability of concluding that two systems are different when, in fact, they are not, and k be an index over the sorted uncorrected p-values, $k = 1 \ldots N$. We first find the largest k for which $p_{(k)} \leq \frac{k}{N}\alpha$. The corrected p-values are obtained by normalizing each original value by $\frac{k}{N}$, that is, by multiplying the original value by $\frac{N}{k}$. Note that $\frac{N}{k} \geq 1$ by construction, so the corrected p-values are no smaller than the original values and will generally be larger. Using the corrected p-values makes it more difficult to reject the null hypothesis of equality than using the original values.

The value of k is also used to create adjusted confidence intervals for a mean. For a (1-α)% confidence interval, we discard the $\alpha\frac{k}{2N}$ smallest and the $\alpha\frac{k}{2N}$ largest of the M estimates and use the minimum and maximum values of the remainder as the upper and lower bounds of the confidence interval. In practice, since we are primarily interested in whether one system outperforms another, we pre-sort the runs by mean score. For example, assuming AP as the evaluation measure, we order the scores such that $MAP_{(1)} \geq MAP_{(2)} \geq \cdots \geq MAP_{(n_s)}$. Then, letting $H_l^{(i)}$ denote the test,

$$MAP_{(i)} = MAP_{(l)} \text{ vs. } MAP_{(i)} > MAP_{(l)},$$

for $l \in \{i+1,\ldots,n_s\}$ our adjusted p-values are then derived for each $i = 1,\ldots,n_s$.

3.2 Significantly Different TREC-3 Runs

In this subsection, we compare the sets of run pairs inferred to be significantly different from one another by three methods: a paired t-test, a partial randomization test [18], and the p-values that result from the with-interactions model created using the partition method. In each case, we use $\alpha = 0.05$ as the significance threshold. The t-test and partial randomization methods are included as representative of current practice in IR research [30] (and are used without using any correction for multiple comparisons).

Table 2 gives the count of the number of significantly different run pairs found by each significance test over all run pairs using both AP and P10 scores. Since there are 40 runs, there are a total 780 run pairs; a percentage given in the table is the percentage of all run pairs found to be different.

Figure 4 shows the same data in more detail. Within each graph, each item on both axes is a run and runs are sorted by decreasing raw mean score as computed on the original test

collection. A point plotted at $\{x,y\}$ summarizes the statistical significance decisions reached by the different tests: the set of tests that rejected the null hypothesis in favor of the alternate hypothesis that run x is better than run y, or the fact that no test rejected the null. The diagonal represents comparing a system to itself. A point plotted below the diagonal means that the significance test found a run pair to be significantly different from one another such that the better run is the run with the *smaller* mean score as computed on the original collection. The graph at the top in Figure 4 reports results when using AP as the evaluation measure and the graph at the bottom when using P10.

All three tests agree regarding the distinguishability of the majority of the run pairs. The t-test can distinguish many fewer run pairs than the other two tests, while the partition method distinguishes somewhat more pairs than the partial randomization test. The set of pairs distinguished by the partial randomization test is not a strict subset of the set of pairs distinguished by the partition method, however—there are a few pairs distinguished only by the partial randomization test. In general, the partition method cannot distinguish runs when the difference between the runs is small relative to the size of the residuals across the entire run set.

Note that there are a few points plotted below the diagonal for P10 in Figure 4, showing that the partition method found significance but in the opposite direction as the original mean would suggest. The average P10 score computed on the original collection is a single data point, and its value is dominated by the high-performing topics. The estimates computed by the partition method are smoothed by the replicates and are thus more likely to be reliable. In no case (either for TREC-3 or for any of the additional collections and conditions described below) have we observed a conflict in which two different tests of significance each rejected the null hypothesis but preferred different runs in the pair.

4 ROBUSTNESS

To test the robustness of the partition method, we compute confidence intervals and derive p-values for different TREC datasets and by partitioning the document sets into different numbers of partitions. In each case, we use 10,000 bootstrap iterations.

We use three TREC datasets: the TREC-3 ad hoc task runs described above, the TREC-8 ad hoc task runs, and the runs submitted to (the ad hoc task of) the Terabyte track in 2006. Each of the test collections built from the runs has 50 topics. The TREC-8 document set is similar to the TREC-3 document set (about 500,000 mostly newswire documents), but there are many more TREC-8 runs—129 versus 40—and there are several high-quality manual runs among the TREC-8 runs. Also as with the TREC-3 dataset, the TREC-8 qrels were built using pooling to depth 100. The combination of many and highly effective runs plus deep pools means the TREC-8 test collection is a high-quality collection that has been extensively studied, and we use it here for precisely this reason.

The dataset from the 2006 Terabyte track is starkly different, and we use it because it is so different from the other two. The document set in the 2006 Terabyte collection is the “GOV2” document set containing roughly 25 million web documents. Qrels were not

created by pooling because of the document set size; instead, the 80 runs were sampled such that extended inferred measures [35] could be computed. This sampling process is a stratified sampling method where the strata are document rank ranges. The qrels that results from the sampling process records a document's stratum, defined to be the stratum containing the best rank that document was retrieved at across the run set, for each document in the collection. The computation of the extended inferred measures uses the qrels to count the number of documents in a given stratum and uses those counts in its estimates of the measures' values. Here, we need to compute infAP on partitioned runs using partitioned qrels. We do so by using the stratum assigned to the document in the original collection. While a partitioned run will likely have a different absolute rank for a document than does the original run, since documents not in the current partition are removed from the partitioned run, the strata still correctly reflect the relative positioning of documents and the relative sizes of the strata remain approximately the same.

The number of partitions a test collection is divided into is the primary parameter of the partition method, since the number of replicate scores per topic and run is equal to the number of partitions. We used three partitions in the initial experiments simply as a convenient starting place. In addition to varying the dataset, we also examine the effect of using two, three, or five partitions per dataset.

As mentioned earlier, one consideration in selecting the number of partitions is that we assume a fully balanced design in our model so we need to ensure that all topics have relevant documents in each partition. The first random split into two and three partitions had relevant documents in all partitions for all topics for all datasets. For five partitions, however, both the TREC-3 and TREC-8 datasets had 49 of the 50 topics with relevant documents in all five partitions, so we used just 49 topics for the five-partition split. The Terabyte dataset lost approximately five topics in five-partition splits. Since (as discussed below) smaller number of partitions are both more convenient in practice and appear to be more effective, we did not use a five-partition split for the Terabyte dataset.

Table 3 gives the lengths of 95% confidence intervals on the system effect for all combinations for models without system-topic interactions (top row of a cell) and with interactions (bottom row of a cell). The TREC-3, three-partition data is repeated from Table 1 for convenience. Consistent with the TREC-3 findings, the confidence intervals computed from the with-interactions model are much smaller than those computed from the without-interaction model in all cases, demonstrating that the system-topic effect is highly significant.

With-interactions intervals are roughly the same length across changes in number of partitions. For AP and infAP, slightly smaller intervals are produced with smaller numbers of partitions. Five-partition splits produce slightly smaller intervals for P10, though that may be an artifact of using only 49 topics in the five-partition case (the one topic dropped has the smallest number of relevant in the original collection and thus P10 for that topic is highly variable). While it may seem counterintuitive that smaller numbers of replicates would lead to smaller confidence intervals, the confidence interval size is a function of the total variability in the data. A larger number of partitions results in individual partitions that are

overall more unlike each other as retrieval test collections than a smaller number of partitions because of the confounding effects of the distribution of relevant and similar-but-not-relevant documents across the partitions. Since an evaluation methodology that requires splitting a collection into a small number of partitions—especially just two—is much easier to use in practice than one that requires creating many partitions (which requires ensuring the partition supports the balanced design requirement and creating and tracking more pieces per run), it is fortuitous that smaller numbers of partitions are also more effective. The remainder of our experiments are run using only two or three partitions.

Table 4 gives the number and percentage of run pairs found to be significantly different with $\alpha = 0.05$ or $\alpha = 0.01$ for four tests: the *t*-test, the partial randomization test, the partition method with three partitions, and the partition method with two partitions. (The final line of the table reports counts for the multiple partition method discussed in Section 5.) The *t*-test consistently finds many fewer run pairs significantly different than the other tests. The partial randomization test finds fewer significant differences than the two partition methods. The partition methods are similar, with the two-partition case finding slightly more significant differences than the three-partition case for all datasets and evaluation measures.

Figure 5 shows that the individual decisions regarding significance can differ between the two partition methods despite similar numbers of significantly different pairs found. The plot in the figure has the same structure as Figure 4: axes represent runs sorted by decreasing mean score and the point plotted at $\{x, y\}$ shows the significance decision for "run x better than run y". The three significance tests plotted in Figure 5 are the partial randomization test and the two partition methods, and the data is taken from the Terabyte track so the evaluation measure used is infAP. (The black circle labeled "random and 1p tests reject null" in the figure means that the partial randomization test and exactly one of the two partition methods rejected the null hypothesis.) Each of the three tests has some run pairs for which it alone rejected the null hypothesis.

The reduction in the number of pairs inferred to be significantly different when changing from $\alpha = 0.05$ to $\alpha = 0.01$ is more modest for the partition method than for either the *t*-test or the partial randomization test. This small reduction is corroborating evidence for the observation made when inspecting the *p*-values themselves that the partition method tends to produce relatively many extreme *p*-values, that is, many *p*-values that round to zero to six decimal places.

5 VARYING DOCUMENT SPLITS

The entire partition methodology is driven by the original split of the document set into partitions. The resulting document set split induces the partitioned qrels and runs and hence the scores and residuals in the models. Different splits can, and indeed are very likely to, cause differences in the downstream processing. Each of the experiments described above has been performed on a single (random) split of the document set into the target number of partitions. In this section, we keep a constant target of two-partition splits and investigate the effect of different assignments of documents to those two parts.

As the basis for this set of experiments, we create 10 new random assignments of documents into two partitions for each of the datasets, resulting in a total of 11 two-partition splits per dataset (the original split used in the previous experiments plus the 10 new splits). Each split contains relevant documents for all topics, so no further processing is necessary to support the balanced design requirement. We then perform the entire bootstrap modeling process independently on each split, resulting in 11 different p-values for each run pair in a dataset. The research question is the extent to which we would infer the same decision as to whether the runs in a pair are significantly different from one another across the different initial starting configurations.

Figure 6 plots the amount of agreement observed across the 11 splits. For these experiments, we again use 10,000 bootstrap iterations and $\alpha = 0.05$, and we use only the model with system-topic interactions. A dot is plotted for each pair of runs A,B where the color of the dot represents the number of splits that concur on the significance decision for A,B—either that A and B are not distinguishable from one another or that A is better than B. With 11 splits, there are six possible agreement outcomes: all 11 splits lead to the same decision (0:11), one split leads to a different decision than the other 10 (1:10), and so on, up to a nearly evenly divided decision of 5:6. The darker the dot the more disagreement there is across the splits, so perfect agreement is plotted in white and a 5:6 decision is plotted in black. Runs along each axis are ordered by mean score on the original collection as in previous graphs, though in this case there is no distinction made between whether the x-axis run is better than the y-axis run or vice versa if there is a difference. (We never observed a case where two splits each led to a significantly different decision but preferred different runs.) For the TREC-3 and TREC-8 datasets, AP decisions are plotted in the upper diagonal portion and P10 decisions in the lower diagonal portion of the same graph; runs are sorted by AP score on the original collection.

The total number of run pairs with disagreements is given in Table 5, which also gives counts for each level of disagreement for both $\alpha = 0.05$ and $\alpha = 0.01$. Inspection of the p-values produced by each split for a run pair with disagreements shows that, in general, disagreements are *not* caused by similar p-values that happen to fall on either side of the α threshold. This is consistent with the observation made earlier that the partition method produces relatively many p-values that are essentially zero. Because of this, the different significance thresholds make very little difference.

Figure 6 clearly shows that the particular split of documents into partitions does affect the significance decisions that are inferred. For AP scores, the disagreements are clustered a small off-set from the diagonal, a result caused by the runs being sorted by mean score. Runs in pairs far from the diagonal are easily recognized as being different so agreement is high, while runs in pairs very close to the diagonal are not distinguished by any test so agreement is again high. P10 scores are inherently more variable then AP scores, and this variability is reflected in a more widespread pattern of disagreements. The pattern of disagreements for infAP is less widespread than for P10 but noticeably more dispersed than for AP. While the total number of pairs that have disagreements for different splits is less than 15% of all pairs in all cases, the impact is greater than the 15% suggests: the run pairs with disagreements are

the run pairs that we would most want to submit to a statistical test, because they are the pairs with the least obvious decision.

The number of run pairs with different decisions on different splits is evidence that the model constructed by a single instance of the partition method captures too much detail of the particular assignment of documents to partitions to make reliable inferences. To produce more reliable inferences, an application of the method should itself use multiple splits and combine the splits' decisions into one grand decision as outlined in Figure 7.

Both the best way of combining the different p-values into a single final inference and the number of splits to use (J in Figure 7) are areas for future research. Here, we used $J = 11$ and the brute-force aggregation method of rejecting the null hypothesis that two systems are equally effective only if the hypothesis would be rejected in each of the splits. This is the most conservative aggregation approach in that it will infer the fewest pairs to be different. Using this strict policy of perfect agreement, the partition method still finds more significantly different run pairs than either the partial randomization or paired t tests, as shown in the final row of Table 4.

6 SMALL RUN SETS

Carterette describes different approaches for analyzing system results depending on which systems are used to fit a model and how p-values are adjusted from inferences on the model [11]. As he explains, a model built from all runs in a large run set resulting from an evaluation exercise such as a TREC track (his option 1a) retains the most amount of knowledge of the world, so comparisons corrected for all pairs in that set can be considered the most "honest." All of the experiments described so far in this article fit in this category, since they are based on the entire run set. A tool that can distinguish among runs in the context of a whole track is very useful, but a tool that individual research teams can use in the course of their own research is even more useful. In this case, we would be fitting a model only on the set of runs produced by a single team (Carterette's option 2a), which means a model fit on a very much smaller set of runs. This section reports on preliminary experiments examining the efficacy of the partition method when used to compare a small set of runs such as the set of runs that result from using the same retrieval system with different parameter settings.

For these initial experiments, we use only AP as the evaluation measure and hence just the TREC-3 and TREC-8 collections. We use a single three-partition split of the document set and $M = 1000$ bootstrap iterations. In TREC-3, a given participant was restricted to submitting at most two runs; in TREC-8 a participant could submit up to five runs. We call the set of runs submitted by a single participant *related* runs. We create bootstrap ANOVA models using just the runs in each related run set in turn, which we call the *Within* condition. We compare the confidence intervals and significance decisions induced in the Within condition to those when the entire run set is used to build the model, the *Across* condition.

The 95% confidence interval results are summarized in Table 6 for both models with and without a system-topic interaction effect. The table gives the mean length of the confidence

interval on the system effect over all runs that occur in a related run set. Confidence intervals shrink when computed using only the runs in a related set. This can be explained by the fact that a set of runs from a given participant will generally exhibit much less total variability than a larger set of runs from many disparate systems. The with-interactions model causes the confidence intervals to shrink further, indicating that some system-topic interaction effect nevertheless remains for related runs.

We now compare the agreement in significance decisions between the Within and Across conditions (using the with-interactions model and $\alpha = 0.05$). For TREC-3 there are 17 related-run pairs. Of those, 14 pairs (82.4%) have the same significance decision in both conditions. For TREC-8 there are 180 related-run pairs, 171 (95%) of which have the same significance decision in both conditions. Except for a related pair of extremely ineffective runs (ineffective enough so the residuals in the Across model are comparable to the runs' scores), differences in significance decisions are always such that the Within condition finds the runs indistinguishable when the Across condition rejects the null hypothesis.

To echo Carterette, models built from a small number of related runs necessarily contain less information than models built from larger, more diverse run sets. Individual research teams could incorporate others' runs on the same test collection from a repository of runs as suggested by Armstrong and colleagues [1] to benefit from increased diversity.

7 CONCLUDING REMARKS

Randomly partitioning the document set of a test collection into just two or three parts creates sufficient replicate scores for system-topic combinations to build bootstrap ANOVA models that can account for system, topic, and system-topic interaction effects. The with-interaction models yield tighter estimates of the system effect than do models without the interaction effect thereby increasing the sensitivity of system comparisons. Significance tests based on confidence intervals of system-effect sizes constructed using these with-interaction models find more significantly different pairs than do the tests currently in common use.

Because the partition method assumes a balanced design in an ANOVA, solutions for finding the parameters of the models have closed forms. This means the technique requires only modest computational resources. Thus, the partition method is a both powerful and practical tool for comparing retrieval systems' effectiveness.

There remain a variety of additional questions to investigate with regard to the methodology. As noted earlier, one such question is how best to combine multiple initial assignments of documents to partitions to compute a single p-value for a run pair. The effect of the number and diversity of runs used to build the bootstrap model requires further investigation. There are different corrections for multiple comparisons that can be tried.

The decision to assume a balanced design in the ANOVA can also can be revisited. While the balanced design makes computing the models much more efficient, it also severely restricts the document set partitions that can be used, essentially making the topic with the smallest number of relevant documents the controlling factor. Allowing an unbalanced

design—or controlling for different numbers of relevant documents per topic in some other manner—might show some benefit for larger numbers of partitions.

Acknowledgments

The authors thank Paul Over for extensive discussions in the initial phases of this work. The anonymous reviewers made many helpful comments that improved the article.

References

1. Armstrong, Timothy G., Moffat, Alistair, Webber, William, Zobel, Justin. EvaluatIR: An Online Tool for Evaluating and Comparing IR Systems. Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09); 2009. p. 833-833.DOI: http://dx.doi.org/10.1145/1571941.1572153
2. Bailey, Peter, Moffat, Alistair, Scholer, Falk, Thomas, Paul. User variability and IR system evaluation. Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'15); 2015. p. 625-634.DOI:http://dx.doi.org/10.1145/2766462.2767728
3. Banks, David, Over, Paul, Zhang, Nien-Fan. Blind men and elephants: Six approaches to TREC data. Info Retriev. 1999 May 1–2.1:7–34. DOI:http://dx.doi.org/10.1023/A:1009984519381.
4. Benjamini, Yoav, Hochberg, Yosef. Controlling the False Discovery Rate–A Practical and Powerful Approach to Multiple Testing. J Roy Stat Soc Ser B (Methodol). 1995; 57(1):289–300. DOI:http://dx.doi.org/10.2307/2346101.
5. Boytsov, Leonid, Belova, Anna, Westfall, Peter. Deciding on an adjustment for multiplicity in IR experiments. Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13); 2013. p. 403-412.DOI:http://dx.doi.org/10.1145/2484028.2484034
6. Buckley, Chris, Voorhees, Ellen M. Evaluating evaluation measure stability. Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00); 2000. p. 33-40.DOI: http://dx.doi.org/10.1145/345508.345543
7. Buckley, Chris, Voorhees, Ellen M. Retrieval system evaluation. In: Voorhees, Ellen M., Harman, Donna K., editors. TREC: Experiment and Evaluation in Information Retrieval. Vol. Chapter 3. MIT Press; 2005. p. 53-75.
8. Carterette, Ben. System effectiveness, user models, and user utility: A conceptual framework for investigation. Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11); 2011. p. 903-912.DOI:http://dx.doi.org/10.1145/2009916.2010037
9. Carterette, Ben. Bayesian inference for information retrieval evaluation. Proceedings of the ACM SIGIR International Conference on the Theory of Information Retrieval (ICTIR'15); 2015. p. 31-40.DOI:http://dx.doi.org/10.1145/2808194.2809469
10. Carterette, Ben, Kanoulas, Evangelos, Yilmaz, Emine. Simulating simple user behavior for system effectiveness evaluation. Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11); 2011. p. 611-620.DOI:http://dx.doi.org/10.1145/2063576.2063668
11. Carterette, Benjamin A. Multiple testing in statistical analysis of systems-based information retrieval experiments. ACM Trans Info Syst (TOIS). 2012; 30(1) Article 4. DOI:http://dx.doi.org/10.1145/2094072.2094076.
12. Cleverdon CW. The Cranfield tests on index language devices. Aslib Proc. 1967; 19(6):173–194. DOI: http://dx.doi.org/10.1108/eb050097 (Reprinted in *Readings in Information Retrieval*, K. Spärck-Jones and P. Willett, editors, Morgan Kaufmann, 1997.).
13. Cormack, Gordon V., Lynam, Thomas R. Statistical precision of information retrieval evaluation. Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06); 2006. p. 533-540.DOI:http://dx.doi.org/10.1145/1148170.1148262

14. Efron, Bradley, Tibshirani, RJ. An Introduction to the Bootstrap. Chapman and Hall/CRC; 1994.
15. Hull, David. Using statistical testing in the evaluation of retrieval experiments. Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93); 1993. p. 329-338.DOI:http://dx.doi.org/10.1145/160688.160758
16. Järvelin, Kalervo, Kekäläinen, Jaana. Cumulated gain-based evaluation of IR techniques. ACM Trans Info Syst (TOIS). 2002; 20(4):422–446. DOI:http://dx.doi.org/10.1145/582415.582418.
17. Kutner, Michael H., Nachtsheim, Christopher J., Neter, John. Applied Linear Regression Models. 4. McGraw-Hill/Irwin; 2004. international ed
18. Manly, Bryan FJ. Randomization, Bootstrap, and Monte Carlo Methods in Biology. 2. Chapman & Hall; London, UK: 1997.
19. Moffat, Alistair, Thomas, Paul, Scholer, Falk. Users versus models: What observation tells us about effectiveness metrics. Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM'13); 2013. p. 659-668.DOI:http://dx.doi.org/10.1145/2505515.2507665
20. Moffat, Alistair, Zobel, Justin. Rank-biased precision for measurement of retrieval effectiveness. ACM Trans Info Syst (TOIS). 2008; 27(1) Article 2. DOI:http://dx.doi.org/10.1145/1416950.1416952.
21. Robertson, Stephen. On Document Populations and Measures of IR Effectiveness. Foundation for Information Science; Budapest: 2007.
22. Robertson, Stephen E., Kanoulas, Evangelos. On per-topic variance in IR evaluation. Proceedings of the 35th International ACM SIGIR Conference in Research and Development in Information Retrieval (SIGIR'12); 2012. p. 891-900.DOI:http://dx.doi.org/10.1145/2348283.2348402
23. Sakai, Tetsuya. Statistical reform in information retrieval? SIGIR Forum. 2014; 48(1):3–12. DOI:http://dx.doi.org/10.1145/2641383.2641385.
24. Sakai, Tetsuya. A simple and effective approach to score standardisation. Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval (ICTIR'16); 2016. p. 95-104.DOI:http://dx.doi.org/10.1145/2970398.2970399
25. Sakai, Tetsuya, Kando, Noriko. On information retrieval metrics designed for evaluation with incomplete relevance assessments. Info Retriev. 2008; 11(5):447–470. DOI:http://dx.doi.org/10.1007/s10791-008-9059-7.
26. Sanderson, Mark. Test collection based evaluation of information retrieval systems. Found Trends Info Retriev. 2010; 4(4):247–375. DOI:http://dx.doi.org/10.1561/1500000009.
27. Sanderson, Mark, Turpin, Andrew, Zhang, Ying, Scholer, Falk. Differences in effectiveness across sub-collections. Proceedings of the 21st International ACM Conference on Information and Knowledge Management (CIKM'12); 2012. p. 1965-1969.DOI:http://dx.doi.org/10.1145/2396761.2398553
28. Sanderson, Mark, Zobel, Justin. Information retrieval system evaluation: Effort, sensitivity, and reliability. Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05); 2005. p. 162-169.DOI:http://dx.doi.org/10.1145/1076034.1076064
29. Savoy, Jacques. Statistical inference in retrieval effectiveness evaluation. Info Process Manage. 1997; 33(4):495–512. DOI:http://dx.doi.org/10.1016/S0306-4573(97)00027-7.
30. Smucker, Mark D., Allan, James, Carterette, Ben. A comparison of statistical significance tests for information retrieval evaluation. Proceedings of the 16th ACM Conference on Information and Knowledge Management (CIKM'07); 2007. p. 623-632.DOI:http://dx.doi.org/10.1145/1321440.1321528
31. Spärck Jones K, van Rijsbergen CJ. Information retrieval test collections. J Document. 1976; 32(1): 59–75. DOI:http://dx.doi.org/10.1108/eb026616.
32. Voorhees, Ellen M. The philosophy of information retrieval evaluation; Evaluation of Cross-Language Information Retrieval Systems. Proceedings of CLEF 2001 (Lecture Notes in Computer Science). 2002. p. 355-370.DOI:http://dx.doi.org/10.1007/3-540-45691-0_34
33. Webber, William, Moffat, Alistair, Zobel, Justin. Score standardization for inter-collection comparison of retrieval systems. Proceedings of the 31st Annual International ACM SIGIR

Conference on Research and Development in Information Retrieval (SIGIR'08); 2008. p. 51-58.DOI:http://dx.doi.org/10.1145/1390334.1390346

34. Webber, William, Moffat, Alistair, Zobel, Justin. Statistical power in retrieval experimentation. Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM'08); 2008. p. 571-580.DOI:http://dx.doi.org/10.1145/1458082.1458158
35. Yilmaz, Emine, Kanoulas, Evangelos, Aslam, Javed A. A simple and efficient sampling method for estimating AP and NDCG. Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08); 2008. p. 603-610.DOI:http://dx.doi.org/10.1145/1390334.1390437

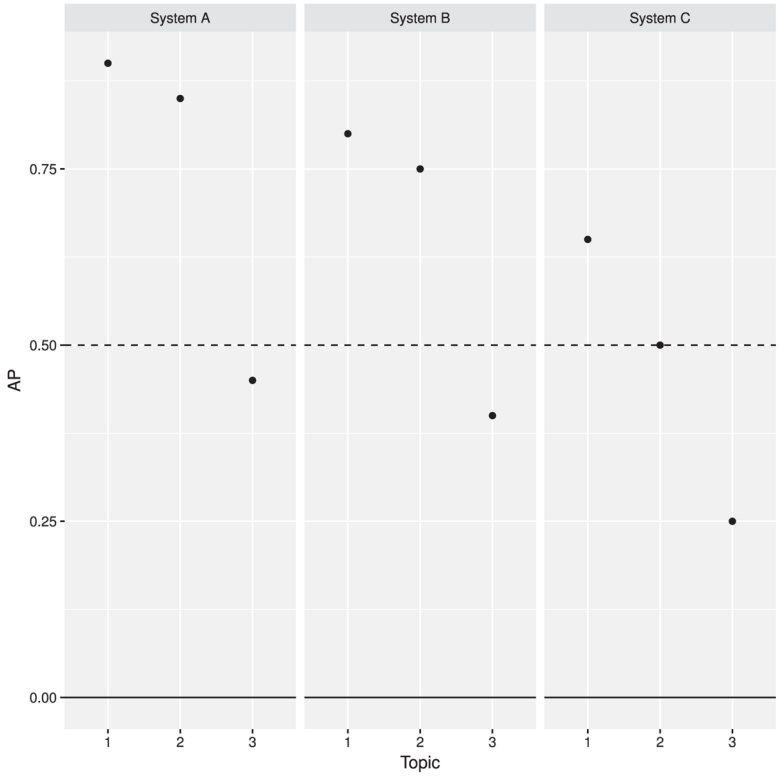

Fig. 1.

Example of the types of effects generally present in search results. Here three systems report Average Precision scores for each of three topics, where each system does relatively less well on topic 3. An ANOVA model that contains only a single term that captures the system effect will have a large error term, because there is no way for that model to capture the dip in performance for all systems for topic 3. Models that include both system and topic effect terms can capture the variability in topics and thus produce less variable estimates of the system effect. Models that contain a third term for the system-topic interaction effects can capture yet more of the variability and produce even tighter bounds on the system effect.

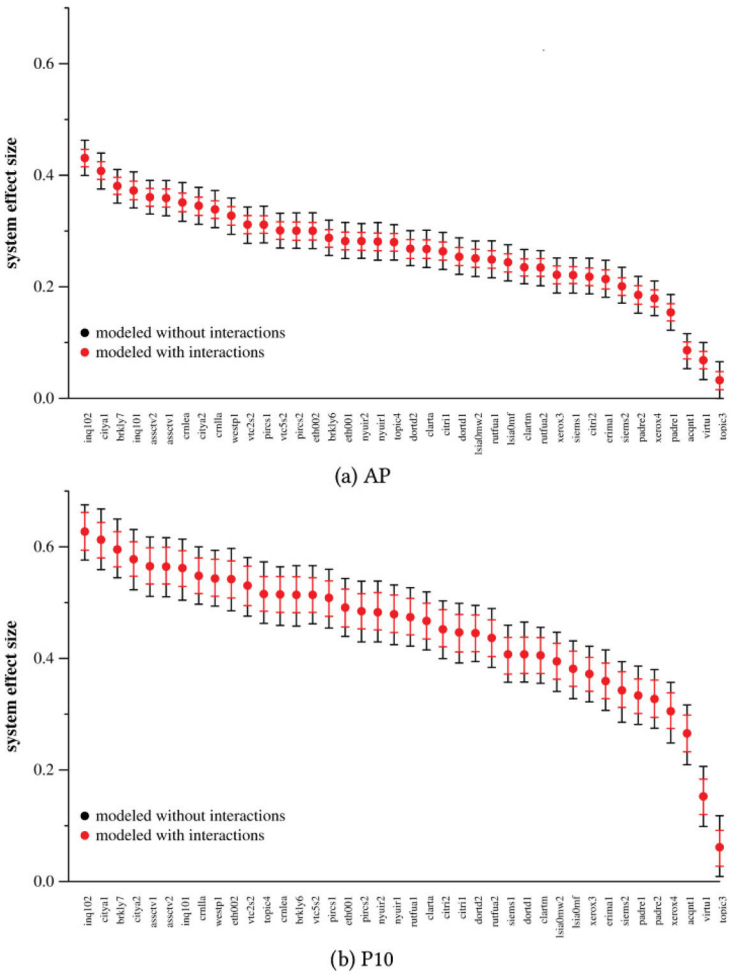

Fig. 2.

95% confidence intervals on the system effect computed using models with (red inner bars) and without (black outer bars) system-topic interactions for TREC-3 runs. Runs evaluated using AP are plotted in the top of the figure and runs evaluated using P10 in the bottom of the figure.

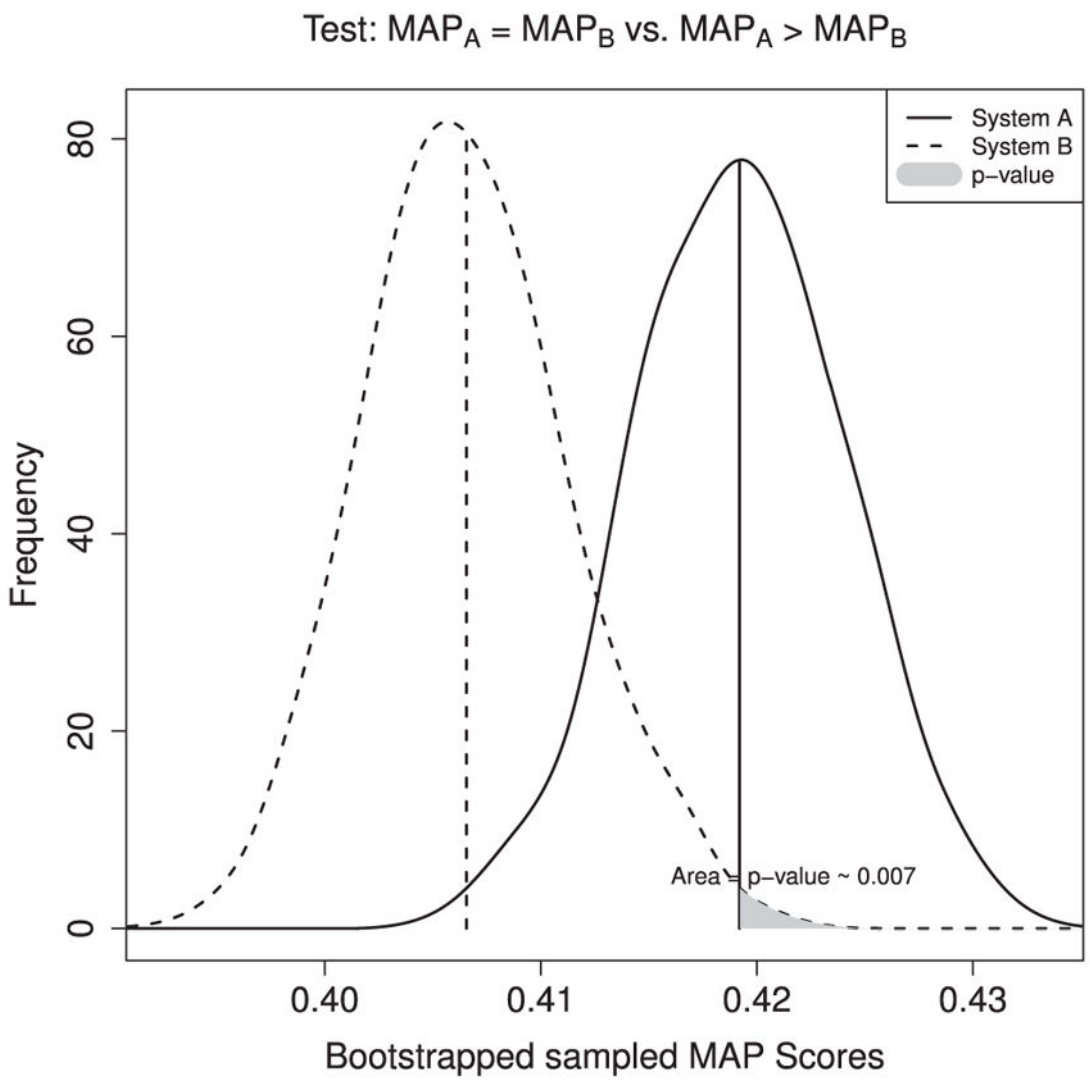

Fig. 3.

Estimated probability density functions for MAP for two runs, plotted as dashed and solid lines. The vertical bars plot the mean values over the bootstrap iterations. The shaded area shows the probability of the event that a value at least as large as MAP_A is, in fact, from the distribution associated with MAP_B.

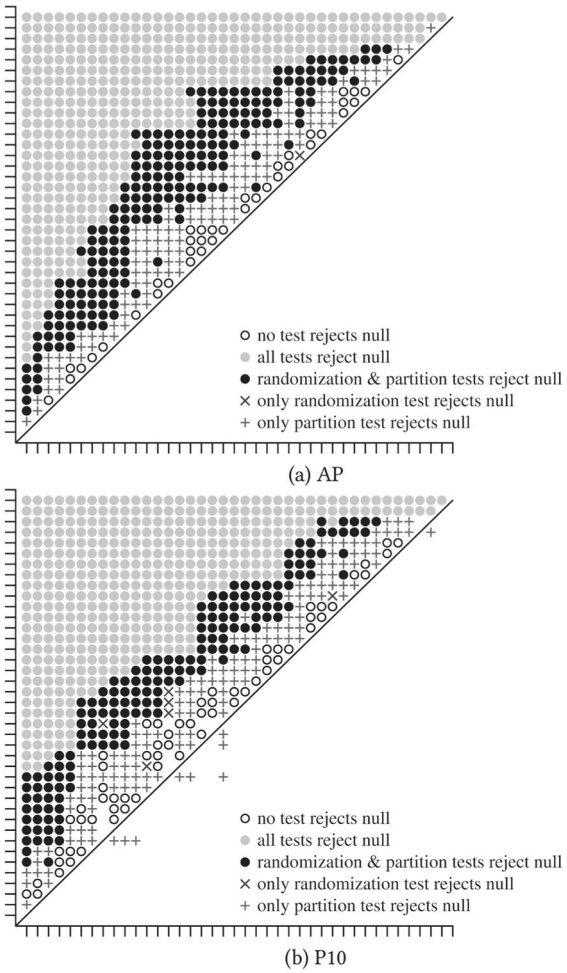

Fig. 4.

Significance decisions for run pairs as determined by different tests with $\alpha = 0.05$ for the TREC-3 dataset. The x- and y-axis are runs sorted by raw mean score computed on the original test collection. A point plotted at $\{x, y\}$ records the significance decisions as to whether system x is better than system y, with the null hypothesis that the two systems are the same. Runs were evaluated using either AP (top) or P10 (bottom).

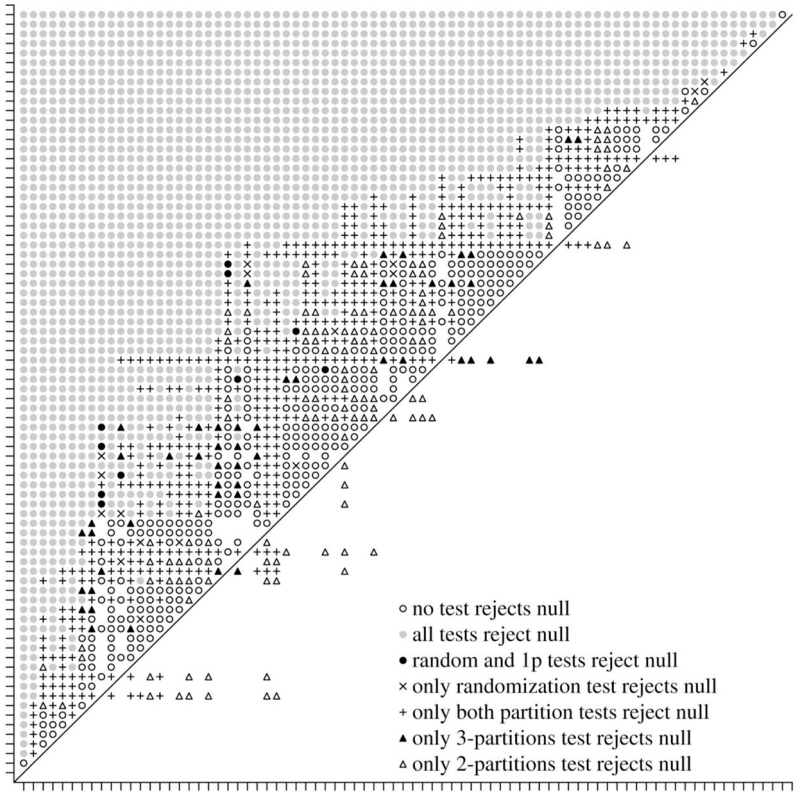

Fig. 5.

Significance decisions for run pairs as determined by the partial randomization test and two variants of the partition method (using either two or three partitions) with $\alpha = 0.05$ for the Terabyte dataset. The x-and y-axis are runs sorted by mean infAP computed on the original test collection. A point plotted at $\{x, y\}$ records the decisions as to whether system x is better than system y, with the null hypothesis that the two systems are the same.

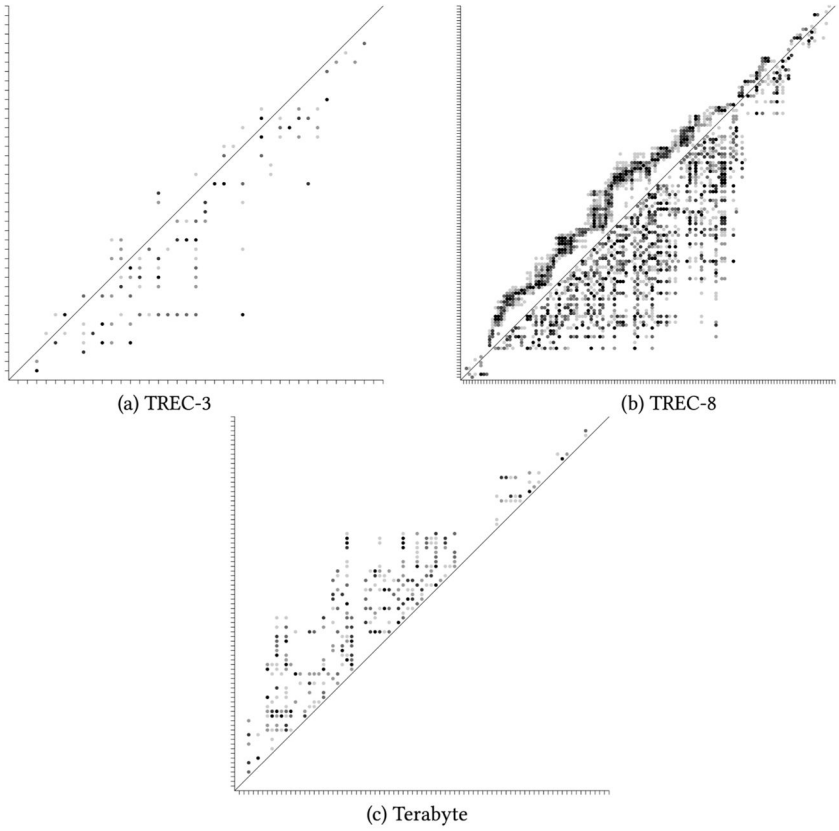

Fig. 6.

Amount of agreement in significance decisions across 11 different random two-partition splits of the dataset with $\alpha = 0.05$. The x- and y-axes are runs sorted by decreasing mean AP (TRECs 3 and 8) or infAP (terabyte) as computed on the original collection. The darker a dot, the more disagreement there is among the different splits as to whether run A is significantly different from run B, with white showing perfect agreement from all 11 splits and black showing a 5-splits-to-6-splits tally. For the TREC-3 and TREC-8 datasets, AP is plotted in the upper diagonal and P10 in the lower diagonal.

Algorithm *Multiple Partition Inference Procedure*
Input: An IR test collection and set of runs produced using the collection; a significance value α; the number of splits to include in the test, J
Output: For each pair of runs, a decision whether to accept or reject the null hypothesis that the runs in the pair are equally effective

```
for i ←1 to J
    do
        generate a new split of the document set into two partitions
        fit a with-interactions model using bootstrap ANOVA over the runs and qrels induced by this split
        compute a corrected p-value for each run pair for the likelihood that the runs in the pair are the same
        for each run pair
            do record the p-value for this pair on this split
        discard current document split and model
for each run pair
    do aggregate the J p-values for this pair into a final decision
```

Fig. 7.
Outline of method for making more reliable inferences about system differences by aggregating decisions over multiple document partitions.

Table 1

Mean, Shortest, and Longest Lengths of 95% Confidence Intervals on the System Effect for TREC-3 Runs

	Without Interactions			With Interactions		
	Mean	Min	Max	Mean	Min	Max
AP	0.064	0.060	0.069	0.032	0.030	0.034
P10	0.106	0.099	0.112	0.065	0.061	0.071

Table 2

Number of Significantly Different Run Pairs Found for TREC-3

	t-test	Randomization	Partition
AP	409 (52.4%)	619 (79.4%)	741 (95.0%)
P10	411 (52.7%)	579 (74.2%)	712 (91.3%)

Table 3

Mean [Minimum, Maximum] Lengths of 95% Confidence Intervals on the System Effect for Different Number of Partitions for Different TREC Datasets and Evaluation Measures. Confidence Intervals for Models with No Interactions Are on Top and for Models with Interactions Are on Bottom

Collection	Measure	2 Partitions	3 Partitions	5 partitions
TREC-3	AP	0.075 [0.071, 0.082]	0.064 [0.060, 0.069]	0.055 [0.052, 0.058]
		0.029 [0.026, 0.031]	0.032 [0.030, 0.034]	0.033 [0.031, 0.034]
	P10	0.130 [0.122, 0.140]	0.106 [0.099, 0.112]	0.081 [0.076, 0.086]
		0.065 [0.061, 0.069]	0.065 [0.061, 0.071]	0.055 [0.052, 0.060]
TREC-8	AP	0.088 [0.082, 0.094]	0.078 [0.070, 0.084]	0.069 [0.065, 0.074]
		0.039 [0.035, 0.042]	0.044 [0.040, 0.047]	0.049 [0.046, 0.053]
	P10	0.122 [0.115, 0.134]	0.098 [0.093, 0.109]	0.071 [0.066, 0.076]
		0.061 [0.055, 0.065]	0.061 [0.057, 0.067]	0.048 [0.045, 0.053]
Terabyte	infAP	0.064 [0.060, 0.071]	0.058 [0.055, 0.064]	—
		0.032 [0.030, 0.035]	0.037 [0.035, 0.040]	

Table 4

Number (Percentage) of Significantly Different Run Pairs Found

		TREC-3 (780 pairs)		TREC-8 (8256 pairs)		Terabyte (3160 pairs)
		AP	P10	AP	P10	infAP
t-test	$\alpha = 0.05$	409 (52.4%)	411 (52.7%)	4164 (50.4%)	4317 (52.3%)	1261 (39.9%)
	$\alpha = 0.01$	310 (39.7%)	324 (41.5%)	3437 (41.6%)	3695 (44.8%)	976 (30.9%)
Randomization	$\alpha = 0.05$	619 (79.4%)	579 (74.2%)	6325 (76.6%)	5663 (68.6%)	2114 (66.9%)
	$\alpha = 0.01$	544 (69.7%)	491 (62.9%)	5571 (67.5%)	4903 (59.4%)	1700 (53.8%)
3 Parts	$\alpha = 0.05$	741 (95.0%)	712 (91.3%)	7413 (89.8%)	7112 (86.1%)	2662 (84.2%)
	$\alpha = 0.01$	728 (93.3%)	693 (88.8%)	7150 (86.6%)	6786 (82.2%)	2540 (80.4%)
2 Parts	$\alpha = 0.05$	743 (95.3%)	712 (91.3%)	7510 (91.0%)	7254 (87.9%)	2742 (86.8%)
	$\alpha = 0.01$	730 (93.6%)	700 (89.7%)	7269 (88.0%)	6930 (83.9%)	2637 (83.4%)
Multiple 2 Parts	$\alpha = 0.05$	733 (94.0%)	677 (86.8%)	7152 (86.6%)	6616 (80.1%)	2595 (82.1%)
	$\alpha = 0.01$	723 (92.7%)	656 (84.1%)	6901 (83.6%)	6324 (76.6%)	2447 (77.4%)

ACM Trans Inf Syst. Author manuscript; available in PMC 2018 June 12.

Table 5

Number (Percentage) of All Run Pairs that Have Disagreements Across 11 Two-Partition Splits

		TREC-3 (780 pairs)		TREC-8 (8,256 pairs)		Terabyte (3,160 pairs)
		AP	P10	AP	P10	infAP
1:10	$\alpha = 0.05$	8 (1.0%)	23 (2.9%)	235 (2.8%)	357 (4.3%)	95 (3.0%)
	$\alpha = 0.01$	12 (1.5%)	33 (4.2%)	229 (2.8%)	343 (4.2%)	111 (3.5%)
2:9	$\alpha = 0.05$	5 (0.6%)	17 (2.2%)	133 (1.6%)	244 (3.0%)	60 (1.9%)
	$\alpha = 0.01$	5 (0.6%)	15 (1.9%)	132 (1.6%)	242 (2.9%)	62 (2.0%)
3:8	$\alpha = 0.05$	2 (0.3%)	17 (2.2%)	92 (1.1%)	182 (2.2%)	43 (1.4%)
	$\alpha = 0.01$	2 (0.3%)	17 (2.2%)	111 (1.3%)	181 (2.2%)	52 (1.6%)
4:7	$\alpha = 0.05$	1 (0.1%)	11 (1.4%)	95 (1.2%)	155 (1.9%)	38 (1.2%)
	$\alpha = 0.01$	3 (0.4%)	12 (1.5%)	96 (1.2%)	166 (2.0%)	41 (1.3%)
5:6	$\alpha = 0.05$	2 (0.3%)	14 (1.8%)	86 (1.0%)	165 (2.0%)	33 (1.0%)
	$\alpha = 0.01$	1 (0.1%)	8 (1.0%)	99 (1.2%)	141 (1.7%)	40 (1.3%)
Total	$\alpha = 0.05$	18 (2.3%)	82 (10.5%)	641 (7.8%)	1103 (13.4%)	269 (8.5%)
	$\alpha = 0.01$	23 (2.9%)	85 (10.9%)	667 (8.1%)	1073 (13.0%)	306 (9.7%)

Table 6

Mean Size of 95% Confidence Intervals on the System Effect (AP) for Runs: With Interactions-in-the-Model Versus Without and Across-All-Systems Versus Within-Related-Sets

	TREC-3		TREC-8	
	Without	**With**	**Without**	**With**
Across	0.064	0.032	0.078	0.044
Within	0.039	0.030	0.052	0.040