Time-Based Calibration of Effectiveness Measures

Mark D. Smucker
Department of Management Sciences
University of Waterloo, Canada
mark.smucker@uwaterloo.ca

Charles L. A. Clarke
School of Computer Science
University of Waterloo, Canada
claclark@plg.uwaterloo.ca

ABSTRACT

Many current effectiveness measures incorporate simplifying assumptions about user behavior. These assumptions prevent the measures from reflecting aspects of the search process that directly impact the quality of retrieval results as experienced by the user. In particular, these measures implicitly model users as working down a list of retrieval results, spending equal time assessing each document. In reality, even a careful user, intending to identify as much relevant material as possible, must spend longer on some documents than on others. Aspects such as document length, duplicates and summaries all influence the time required. In this paper, we introduce a *time-biased gain* measure, which explicitly accommodates such aspects of the search process. By conducting an appropriate user study, we calibrate and validate the measure against the TREC 2005 Robust Track test collection. We examine properties of the measure, contrasting it to traditional effectiveness measures, and exploring its extension to other aspects and environments. As its primary benefit, the measure allows us to evaluate system performance in human terms, while maintaining the simplicity and repeatability of system-oriented tests. Overall, we aim to achieve a clearer connection between user-oriented studies and system-oriented tests, allowing us to better transfer insights and outcomes from one to the other.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*

General Terms

Experimentation, Human Factors, Measurement

Keywords

Information retrieval, search evaluation

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.

1. INTRODUCTION

System-oriented tests and user-oriented studies represent competing approaches to search engine evaluation. A system-oriented test typically develops a set of relevance judgments to determine the quality of ranked document lists returned in response to a query [39]. Effectiveness measures computed over these ranked lists provide a method for tuning search engines and comparing one engine to another. In contrast, a user-oriented study examines actual user behavior during interactive retrieval sessions. These studies span a spectrum from intensive laboratory investigations involving a relatively small number of subjects [18] to large-scale analyses of interaction logs capturing the search activities of millions of people [9, 42, 43].

The value of a user-oriented study lies in its ability to reflect many aspects of the search process, including interaction and presentation. Unfortunately, user studies can be complex and costly. Running subjects through a laboratory experiment can require substantial time and money, and we may need to repeat experiments as elements of the interface and engine change. On the other hand, interaction logs are noisy, uncontrolled, and capture only a portion of the user experience.

The value of a system-oriented test lies in its simplicity and repeatability. Effectiveness measures isolate and evaluate a single element of the search engine, i.e., the quality of its ranked list. Once we develop a set of relevance judgments, evaluation measures can be re-computed with little additional expense or effort. As software developers modify and train ranking algorithms, effectiveness measures may be applied over and over again, as often as needed.

Various researchers [7, 10, 43] suggest a generic framework that encapsulates many of the effectiveness measures in current use, including Normalized Discounted Cumulative Gain (nDCG) [17], Rank Biased Precision (RBP) [23], and Expected Reciprocal Rank (ERR) [9]. All of these measures can be expressed as a sum over documents in a ranked list,

$$\frac{1}{\mathcal{N}} \sum_{k=1}^{\infty} g_k d_k, \tag{1}$$

where g_k represents the *gain* associated with the document appearing at rank k, and d_k represents a *discount* associated with rank k, which is independent of the document. In practice, we either compute the sum to some fixed but arbitrary depth K or until we know its value within some ϵ. The normalization factor $\mathcal{N}$ is optional, but if it is present, it serves to map the value of the sum into the range $[0:1]$, which may be required for averaging over multiple queries.

RBP uses the discount formula $d_k = \beta^{k-1}$, where β is a constant in the range $[0 : 1]$. Gain values for RBP are binary, with $g_k = 1$ if the document at rank k is judged relevant, and $g_k = 0$ if it is not judged relevant. ERR uses a reciprocal rank discount formula of $d_k = 1/k$, and nDCG uses the discount formula $d_k = 1/\left(\log_2(1+k)\right)$. For both of these measures, a gain value may be interpreted as a relevance probability, i.e., the probability that a user viewing the document would judge it to be relevant. When applying these measures, the probabilities are estimated from graded relevance judgments, e.g., "definitive", "excellent", "good", etc. [24]. In the case of ERR, the gain value of a document at rank k depends on the gain values of the documents at ranks 1 to $k-1$, with gain values increasingly attenuated when more relevant documents appear at higher ranks.

All of these measures may be interpreted in terms of a simple user model: The user starts at the first document and works their way down the list, eventually stopping when they become tired or bored. The discount d_k indicates the probability that the user continues to rank k, and the gain g_k represents the benefit (i.e., utility) to the user of viewing the document at rank k. Thus, the sum in Equation 1 can be understood as the expected total benefit experienced by the user, with the various gain values and discount formulae corresponding to different assumptions about user needs and behaviors. In particular, the attenuated gain values in ERR correspond to the idea that, after viewing each relevant document, the user derives less benefit from viewing additional relevant documents.

Underlying this user model is an implicit assumption that the user views documents at a constant rate, spending the same time assessing each one. Because of this assumption, these effectiveness measures can fail to accommodate aspects of the search process that directly impact the quality of the ranked list as experienced by the user. For example, most modern search engines provide captions, or summaries, that describe each of the documents [36]. A user scanning these summaries might quickly skip those results that are obviously not relevant to their needs, without ever viewing the contents of the associated documents. Similarly, we might expect shorter documents to take less time to assess than longer documents, since a user may spend less time skimming them. In addition, when a document is similar in content to a document a user has already viewed, it may be easy to assess quickly, regardless of its length. Once a user recognizes the duplicated content, they can immediately return to the result list and continue to the next result.

In this paper, we propose a framework for evaluation measures based on time, allowing us to directly accommodate aspects of the search process ignored by standard effectiveness measures. We retain the simple model of a user working their way down a result list, but with both gain and discount now expressed in terms of time. As the user works their way down the list, let $G(t)$ be the cumulative gain experienced by the user at time t. In other words, if the user invests time t in assessing the result list, they will derive a total benefit of $G(t)$. We model the possibility that the user stops at some point by a *decay* function $D(t)$, which indicates the probability that the user continues until time t. Thus, $D(0) = 1$ and decreases monotonically to 0 as $t \to \infty$. Under this model, an equivalent of Equation 1 may be expressed as

$$\frac{1}{\mathcal{N}} \int_0^\infty \frac{dG}{dt} D(t) dt. \tag{2}$$

Of course, measuring instantaneous gain would prove difficult in practice, and we quickly return to considering gain values one document at a time, but retaining decay as a function of time. By smearing gain equally across each document and converting $D(t)$ into a step function, we may approximate Equation 2 with

$$\frac{1}{\mathcal{N}} \sum_{k=1}^{\infty} g_k D(T(k)), \tag{3}$$

where $T(k)$ is the expected time it would takes for a user to reach rank k and begin to assess the document. Although we have returned to a summation over ranks, we now explicitly consider time as a part of the measure.

We refer to Equation 3 as *time-biased gain*, and we spend the remainder of this paper developing and exploring this effectiveness measure. In its development, we face the potentially controversial and complex problem of providing a realistic estimate for $T(k)$. Ideally, for use in an effectiveness measure, the estimate should be straightforward both to compute and to comprehend. Competing with this requirement is the need for a clear connection back to actual user behavior, so that the effectiveness measure provides a meaningful guide for training and tuning ranking algorithms.

We note that setting $T(k) = c(k-1)$, for some constant c, reduces Equation 3 back to Equation 1, which corresponds to a scenario in which the user views documents at a constant rate. To move beyond this simple scenario, we need to examine user behavior in a realistic context, calibrating our estimate of $T(k)$ against the actions of these users. As a result, time-biased gain is more closely tied to a usage scenario than traditional evaluation measures. Moving to a different scenario requires a different calibration of the evaluation measure, a price we pay for the closer connection to user behavior.

After a review of related work, we discuss the calibration and validation of time-biased gain against the test collection employed in the TREC 2005 Robust Track experiments [39]. As part of this calibration, we report a user study in which we presented subjects with an interface styled after a modern web search engine, including a result page with ten query-biased summaries, and instructed them to find and save as many relevant documents as possible in the time allowed, while making as few mistakes as possible. The associated scenario is of a careful and determined user, with a topical information need, intending to identify as much relevant material as they can in the time they have available.

We believe this scenario is appropriate for the TREC Robust Track, and similar experiments, where a document is considered relevant if it contains any relevant material at all, and evaluation measures traditionally consider documents that are relatively deep in the result list. For other scenarios, a different calibration might be appropriate. For example, in the context of web search, where users may interact with fewer results and rarely dig as deeply in the result list, calibration might be based on interaction logs taken from the search engine itself.

After developing a $T(k)$ function appropriate to our usage scenario, we consider other components of our time-biased gain measure, including the choice of a decay function. We explore properties of the measure and compare it to other evaluation measures. Finally, we end the paper with a discussion of extensions and future work.

2. RELATED WORK

We may interpret many current effectiveness measures in terms of the simple user model outlined in the introduction [7, 9, 17, 23, 43]. While all of these measures implicitly assume that a user progresses down the ranked list at a constant rate, they incorporate different assumptions about when and why the user will stop, and how much benefit or utility they can be expected to gain. All of these measures may be expressed in form of Equation 1 [7, 43]. Even precision at k may be interpreted in terms of a user who always views exactly the top k documents, and no more. Along the same lines, Robertson [26] retrofits a similar user model into the venerable Mean Average Precision (MAP) effectiveness measure, which may also be extended to incorporate graded relevance values similar to those employed by nDCG [15, 27, 30].

Turpin et al. [37], Yilmaz et al. [42], and Dupret [14] have gone beyond this simple model by explicitly incorporating summaries into their user models and thus into their metrics. An important difference between their work and ours is that we explicitly model time in addition to the click decisions. Summaries are designed to, and do speed the rate at which users find relevant documents [20, 36]. We believe it is important that metrics reflect the value of summaries by having a user model that incorporates the time required to separately evaluate summaries and documents. Time is important to users. Su [35] found from interviews with 40 users that the success category of "Efficiency - Time" was ranked first with the highest frequency of mention.

Turpin and Hersh [38] hypothesized that the time to read documents was a reason why batch-style and user evaluations produce different results. In their study, Turpin and Hersh found that the baseline system, which had a lower MAP, ranked shorter documents higher than the *improved* system. The *improved* system had a pivoted length normalization [31] component that preferred longer documents — longer TREC documents have a higher prior probability of being relevant. Turpin and Hersh found that users of the baseline system were able to read more documents than the users of the *improved* system. As a result, users of the baseline system were able to compensate for its lower precision and find equal amounts of relevant material in the same amount of time. Our time-biased gain metric explicitly takes document length into consideration, for we have also found that users take longer to judge longer documents. Yang and Lad [41] and Arvola et al. [2] also consider the cost of reading in the evaluation of IR systems.

Our time-biased gain metric aligns well with Fuhr's probability ranking principle for interactive IR [16]. Fuhr carefully outlines that users of interactive IR systems have to make a series of decisions and that these decisions each have their own cost. For our metric, cost is measured in time, and both cost and gain are directly tied to the series of probabilistic decisions made by the user.

Much of this work can be viewed as following a framework for evaluation laid out by Cooper [11, 12] and Dunlop [13]. Cooper believed that effectiveness measures should be based on the utility a user personally gains from a retrieval system. A critical part of Cooper's plan was the use of *validation experiments* to measure the ability of an evaluation measure to predict utility. Dunlop built on and refined the ideas of Cooper. Rather than subjectively measure a user's utility, Dunlop established the use of HCI methods to predict user performance in terms of the number of relevant documents found in a given amount of time.

Efforts to relate evaluation measures to models of user behavior can be seen as an attempt to bridge the gap between system-oriented tests and user-oriented studies. In these cases, the goal is to move system-oriented measures closer to user-oriented studies. Working from the other direction, efforts have been made to simulate the use of interactive retrieval systems in order to determine their quality [1, 4, 21, 40]. Efforts related to this approach includes the use of simulation to better understand interactive retrieval behavior [3, 19]. We see these two directions of research merging as system-oriented measures continue to increase the resolution of their user models [8].

3. CALIBRATION

At the core of our new metric, time-biased gain, is the computation of the time to reach rank k of a ranked list, which we will refer to as the function $T(k)$. Given that we are retaining the traditional model of a user working down a ranked list in order, we need to estimate the amount of time it takes to process a given document. As such, we need to consider what is a reasonable process that a user would go through to determine if a document is relevant.

The majority of search interfaces now present a ranked list in two parts. The first part is the display of a series of document summaries that are intended to help the user decide if the corresponding full documents should be viewed or skipped. The second part is the full document itself.

Faced with this interface, the user must first examine the document summary and decide whether to click and view the full document or not. While it is certainly possible that a user may realize benefit from viewing the summary alone, we believe that for TREC-style topics and information needs, that a user needs to view the full document in order to realize gain from finding relevant material in the document.

Thus, we need to model the decision of whether to click or not and produce both an estimate of the time the decision takes to be made as well as an estimate of what the decision will be. If the user decides to click on a summary, then the user must study the document and make a decision about the document's relevance. Only if the user clicks on a relevant document's summary and then decides the document is relevant will the user realize any gain. Gain cannot be obtained from non-relevant documents nor from skipped relevant documents. As with the summary, we need to estimate time spent on the full document and the final relevance decision of the user.

There are nearly endless variables that could be taken into consideration in the models of decision making for summaries and documents. In this paper, we choose to focus on a few variables that are most likely to affect user decisions.

The most obvious variables affecting these decisions are the user and the topic. While users vary greatly and contribute large amounts of variance to user studies, we do not attempt to model different user strategies for processing ranked lists. Instead, our user model is an idealized individual representing the population as a whole. As for the topic, at this time we do not see an easy way to identify a priori how a topic will affect users' behavior, and we are forced to treat all topics the same.

After the user and topic, the document itself is likely the most important variable to consider. We consider two at-

tributes of the document to use in our model: the relevance of the document and the document's length. Relevance should be a strong indicator of how likely summaries will be clicked and whether or not the document will be recognized by users to be relevant [37]. The time to make decisions based on textual items is likely to be dependent on their length.

Given that summaries tend to be short and of nearly equal length, we will assume they are all assessed in the same amount time, T_S seconds. For documents, let $T_D(l) = al+b$, be the time in seconds to assess a document of length l, where l is measured in words. $T_D(l)$ models an evaluation process of scanning the document at a rate of a seconds per word and then a constant amount of time, b seconds, to account for carefully assessing a small amount of text and the overhead of making a decision about relevance. As we will justify, if a user views a duplicate of a document later, then we treat that duplicate as if it had zero length ($l = 0$).

We use a document's relevance to predict the probability that a summary will be clicked and to predict the probability that when viewed, a full document is recognized as relevant. Formally, let $P(C|R)$ be the probability of a click given NIST relevance, where C and R are binary random variables. Also, let $P(S|R)$ be the probability of saving a viewed document as relevant given NIST relevance, where S and R are binary random variables. Saving a document is the equivalent of recognizing a document to be relevant.

Putting both the time to make decisions and the probabilities of the decisions together, we have a model of the time it takes to process a document in the ranked list and can now estimate the time to reach rank k, i.e. $T(k)$.

Let l_i be the length of the document at rank i. Let r_i be the NIST binary relevance judgment associated with the document at rank i, where $r_i = 1$ if the document is relevant, $r_i = 0$ otherwise. Then, we estimate the expected time for the user to reach rank k as:

$$T(k) = \sum_{i=1}^{k-1} T_S + T_D(l_i)P(C = 1|R = r_i) \quad (4)$$

Not present in $T(k)$ are the probabilities of the user saving a document as relevant, which are only needed for computation of the gain.

With Equation 4 in hand, we need to calibrate it to produce estimates in line with actual human behavior. We next explain how we calibrated $T(k)$ and estimated $P(S|R)$ based on a user study.

3.1 User Study

To calibrate our metric, we need actual user data that involves the processing of ranked lists. We utilize data collected as part of the user study conducted by Smucker and Jethani [34].

The user study presented participants with a user interface styled after modern web search engines. The interface had two types of web pages. One page looked like a search engine results page with 10 query-biased document summaries. Clicking on a summary took a participant to a page with the full document. On the full document page, participants could decide to save the document as relevant or simply use the web browser's back button to return to the summaries page. At the bottom of the summaries page, participants could click a link to obtain the next 10 results. The system recorded all clicks and times spent on the pages as well the decisions to save documents as relevant. The user interface only allowed participants to search the given ranked list of documents. No query reformulation was possible.

The study used 8 topics from the TREC 2005 Robust track, which used the AQUAINT newswire document collection. The topics (310, 336, 362, 367, 383, 426, 427, 436) were selected to be of possible interest to study participants.

48 participants worked on 4 search topics for 10 minutes each. Topics were balanced across participants, list precision, and task order. The study instructed participants to find and save as many relevant documents as possible in the 10 minutes while making as few mistakes as possible. In cases of technical issues affecting data collection, or when it was apparent that participants did not follow instructions, these participants were removed and new participants were recruited until the study was a fully balanced design with 48 participants.

The user study had dual purposes. One purpose was to investigate the effect of precision on user performance, and the other was to provide behavioral data for calibration of our time-biased gain metric. As a result, the ranked lists of documents had a uniform precision of either 0.6 or 0.3, which were representative of the precision at rank 10 performance of the best and lower performing systems in the TREC 2005 Robust track. By uniform, we mean that for every 10 documents shown, either 6 or 3 were relevant. The construction of the result lists placed both relevant and non-relevant documents at ranks reflective of their overall likelihood of being ranked highly by retrieval systems. The result was that we were able to control the precision, and the study participants still had a realistic task of distinguishing between highly ranked relevant and non-relevant documents. Full details of the result lists construction, screenshots of the user interface, and other study details, can be found in [34], which also reports on the results of the precision component of the study.

3.2 Calibration Values

Given the user study, we can estimate the time it takes users to process a document summary and a full document. At first glance, computing these times would appear to be as simple as taking the average of each, for each user, and reporting the user average. Instead, since we are interested in computing gain versus time for a population, we need to compute harmonic means, or as we do in this paper, weighted averages where each weight reflects the amount of a user's activity in the study.

From the user study, we have measurements of the time participants spent on the summaries page before they clicked on a summary or the next link. Likewise, we have measured the time spent viewing documents before leaving the page.

To compute the population average for the time spent on a summary, we need to consider that we can only measure the time spent on the summaries page before clicking on a summary. Without eyetracking we cannot know for certain how much time is spent on each individual summary. Thus, we compute the amount of time per summary for a participant to be the sum of all time spent on the summaries page divided by the maximum rank of a clicked summary. In this way, we spread the time out over both the summaries clicked and not clicked. To compute the population average time on a single summary, we weight the individual partici-

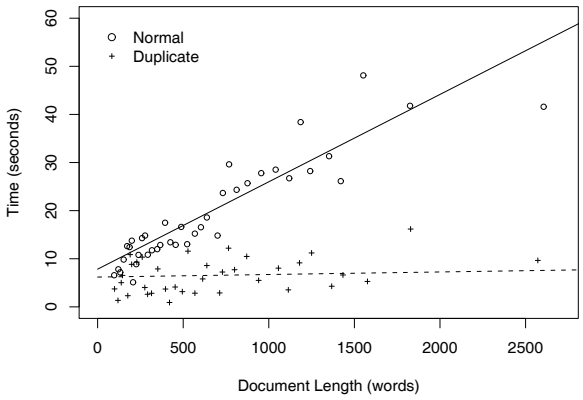

Figure 1: Time to judge document relevance with linear fits to 4212 data points that have been binned and averaged for visualization purposes.

Parameter	Description	Value
P(C=1|R=1)	Probability of click on relevant summary	0.64
P(C=1|R=0)	Probability of click on non-rel. summary	0.39
P(S=1|R=1)	Prob. of save/judge relevant document as rel.	0.77
P(S=1|R=0)	Prob. of save/judge non-relevant doc. as rel.	0.27
T_S	Time to evaluate a summary (seconds)	4.4 s
$T_D(l)$	Time in seconds to judge doc. of l words. Duplicates treated as $l = 0$.	$0.018 \cdot l + 7.8$

Table 1: Calibration values. P(S=1|R=0) is not used by the metric, but is shown for completeness.

pant's time by the maximum rank. As such, the population average time per summary is:

$$T_S = \sum_p \frac{\sum_i S_{ip}}{M_p} \frac{M_p}{\sum_p M_p} = \frac{1}{\sum_p M_p} \sum_p \sum_i S_{ip} \qquad (5)$$

where S_{ip} is the i-th time spent by participant p on the summary page and M_p is the maximum click rank by participant p. For our data, $T_S = 4.4$ seconds. If we did not compute a weighted average, the average would be 5.3 seconds.

For the time spent viewing a full document before saving it as relevant or clicking the web browser's back button to return to the summaries page, we take all of the recorded times and fit a linear model to the time in terms of the document's length. By using all of our participants' data, we will obtain a model of the time to judge a document's relevance that is weighted based on the level of activity of that participant.

Figure 1 show a plot of time to judge document relevance vs. document length. We measured document length as the number of word occurrences in a document as parsed by Indri [32]. For the purposes of the plot, we binned the individual data points such that approximately 15 documents were in each bin. Each single document was judged by an average of 9.1 participants, and thus each plotted data point represents the average of many document lengths and times. Averaging the data points in the plot hides the considerable variation in time to judge documents.

In Figure 1, we have separately plotted the times for documents when first viewed by a participant, and the times for a participant's later views of document duplicates. We identified near duplicates in the AQUAINT collection with a variant of Broder's duplicate detection algorithm [5], and only considered documents with the highest possible similarity to be duplicates of each other, i.e. all shared shingles are identical. We refer to these near duplicates as duplicates. We found 10.1% of the documents to be duplicates.

The linear fits to all of the first-viewed documents (3614 data points) and the duplicate views (598 points) are shown in Figure 1. The linear model for first-viewed documents is given by: $T_D(l) = 0.018 \cdot l + 7.8$, and has an adjusted R-squared of 0.12, which means that 12% of the variance is explained by the model. We looked at including the relevance of the documents and/or the precision of the lists in the linear regression, but neither increased the adjusted R-squared.

The linear fit to the *duplicates* does not explain any of the variance in the time to judge a duplicate, and as such the time to judge a duplicate can be estimated equally well by taking the mean, which gives a result of 6.8 ± 0.5 seconds. The constant in the fit to the first viewed documents is 7.8 ± 0.7. Since the standard errors of the mean duplicate time and the constant term of the linear fit overlap, there is no significant difference between the two, and we simply treat duplicates as zero length documents.

Lorigo et al. [22] found that on information style tasks, participants spent 40% of their time looking at the search results and the remainder elsewhere (presumably at the full web pages). Our participants spent 33% of their time on the summaries and 67% on the full documents.

Of note, user studies typically report user averages. For example, with our data, the average participant took 26.5 seconds per viewed document. In contrast, for the same set of document viewings, the mean of $T_D(l)$ is 18.9 seconds. The difference comes about because the participants who worked faster, took less time per document and also viewed more documents in the same amount of time. If we used the participant averages, we would overestimate the amount of time it takes our population to reach rank k. This difference also means that most statistics reported by user studies cannot be directly used in metrics like ours.

We obtained the parameter settings for the probabilities of clicking on summaries and documents in a straightforward fashion. For both the probability to click on a summary given the relevance, $P(C|R)$, and the probability to save a document as relevant given given relevance, $P(S|R)$, we calculate them as weighted averages. We know the relevance of each document as given by the NIST assessors and can calculate the fraction of views on relevant documents that result in a document being saved, $P(S = 1|R = 1)$, as well as the fraction of views on non-relevant documents that result in a save, $P(S = 1|R = 0)$. For calculating the probability of a click on a summary given the document's relevance, $P(C|R)$, we make the assumption that all summaries up to the maximum clicked rank are viewed. Given this assumption, we can calculate the probabilities of clicking on sum-

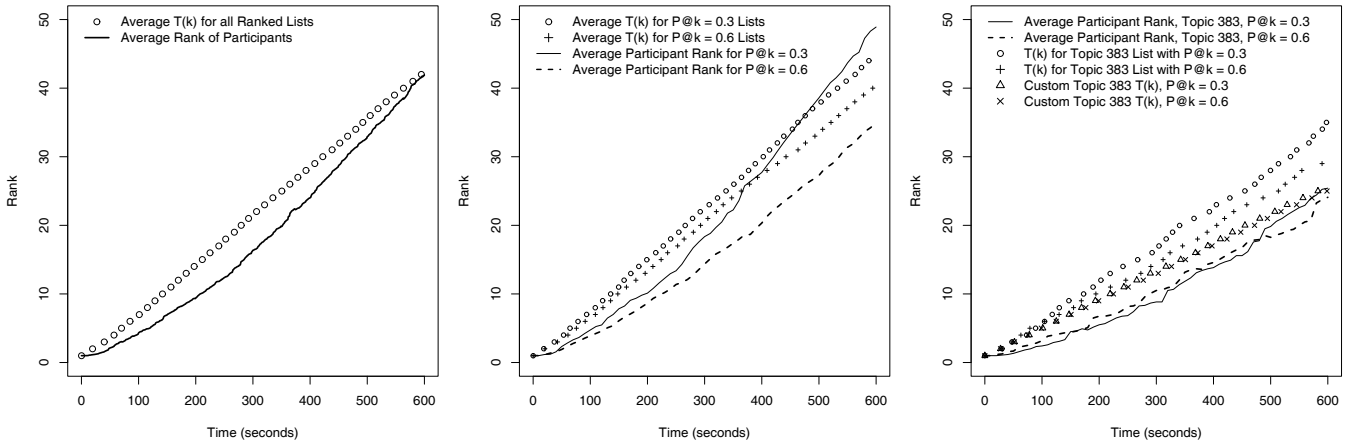

Figure 2: These plots show the predicted time to rank k, $T(k)$, vs. the measured time to rank k for the 48 participants in the user study. Please see the text for a detailed description of each plot.

maries in the same fashion as for documents. In the case of computing the weighted average for documents, we weight a participant's $P(S|R)$ by the number of documents viewed during a search task. In the case of summaries, we weight a participant's $P(C|R)$ by the participant's maximum clicked rank. Table 1 shows the computed probabilities.

Yilmaz et al. [42] reported for a commercial web search engine the probabilities of users clicking on documents of varying relevance: Bad: 0.49, Fair: 0.45, Good: 0.55, Excellent: 0.71, and Perfect: 0.94. In comparison, our probability of clicking on a relevant document is 0.64 and falls in-between "Good" and "Excellent", which seems appropriate. Our probability of clicking on a non-relevant document is 0.39 and falls below "Bad", and shows that our summaries on average provided the study participants with an ability to discriminate between relevant and non-relevant documents. Like Yilmaz et al., we find that a significant fraction of low-quality or non-relevant documents are clicked on.

The probabilities for saving a document show that given a full document, participants are better able to discriminate between relevant and non-relevant as compared to summaries, but that they still have a true positive rate, $P(S = 1|R = 1) = 0.77$ less than one. In other words, not all relevant documents are recognized as such, and this makes sense given a population of users as opposed to modeling the single NIST assessor. Given constraints on time and abilities, not all users will detect when a document is relevant.

4. VALIDATION

In Section 3 we proposed a method to estimate the time it takes a user population to reach rank k, i.e. $T(k)$. Our $T(k)$ consists of Equation 4 and the calibration values for $T(k)$ estimated from a 48 participant user study. Table 1 shows a summary of the calibration values.

Our $T(k)$ is a simple model of user behavior for what is a task of considerable complexity. In processing a ranked list of documents, users must make complex decisions regarding the time they devote to summaries and full documents and the accuracy with which they make decisions. In the user study data that we used to calibrate time-biased gain, there is considerable variation in observed user strategies for the searching of ranked lists [33]. For example, some users quickly select summaries to click while others take 2-3 times longer to make their selection. Another example is that while the study participants primarily moved forward (down) the ranked lists with 94% of the average participant's clicks being on a summary lower ranked than the previously clicked summary, not all movement is forward.

In this section we discuss the ability of our $T(k)$ to make reasonable estimates of the time it takes the population to reach rank k. In the user study, participants searched for relevant documents in 16 ranked lists of documents. Each of 8 topics had two lists. One of the two lists had a precision of 0.3 and the other a precision of 0.6. The design of the study meant that 12 participants worked on each list. Participants worked for 10 minutes on each list.

Figure 2 shows three plots comparing $T(k)$ and actual participant behavior. The far left plot averages $T(k)$ for all 16 of the ranked lists and 192 traces of participant behavior. On average, $T(k)$ produces a linear rate of progress down a ranked list. Given how we produced the calibration values, this average behavior for the 16 ranked lists is expected. As we will show briefly, $T(k)$ produces non-linear predictions for individual ranked lists. At 600 seconds, the average $T(k)$ matches the average rank of the participants and shows that our calibration on average works. If we had not used weighted averages to produce the calibration values, $T(k)$ would have underestimated the rank reached at 600 seconds.

We also see in the far left plot of Figure 2 that the study participants increased their rate as they progressed down the ranked lists. This speedup makes sense given that participants will take some time to learn a model of what is and is not relevant. We also noticed that participants working down some of the 0.3 precision lists dramatically increased their rate of work once getting past the first page of 10 results. Our $T(k)$ is an average of the slower initial behavior and the faster behavior found later in the search process. We leave for future work the possibility of having the parameters of $T(k)$ vary with time.

IR user interfaces employ document summaries as a mechanism to allow users to find relevant material faster. The better the document summaries are, the easier it is for users

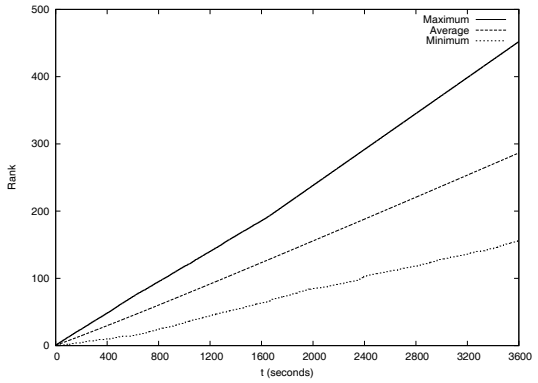

Figure 3: Maximum, average, and minimum rank at time t over all queries and runs from the TREC 2005 Robust Track.

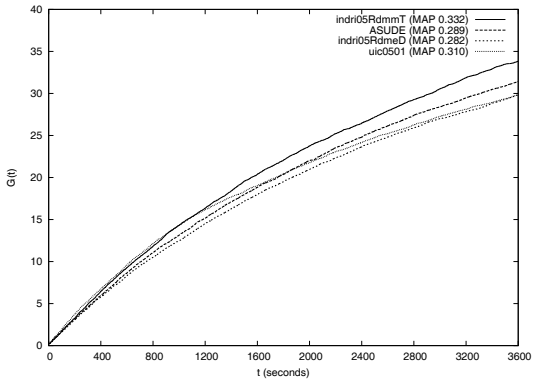

Figure 4: $G(t)$, cumulative gain at time t, for top scoring runs from the TREC 2005 Robust Track.

to save time by skipping non-relevant results. In the center plot of Figure 2, we see that the study participants processing the 0.3 precision list do process more of the lists compared to the 0.6 precision lists. $T(k)$ only models precision in the form of different summary click probabilities given the relevance of the document, but it too predicts that the population will reach a larger rank k with a lower precision list as compared to a higher precision list in the same amount of time. At 600 seconds, the predictions of $T(k)$ fall between the actual ranks reached by the participants.

As mentioned in Section 3, our study design precluded us from using the search topic as part of our model even though it is well-known that topics vary considerably. In the far right plot of Figure 2, we show the participants' behavior for the 0.3 and 0.6 precision lists of Topic 383 (mental illness drugs). Topic 383 requires that a relevant document name an actual drug. The documents in the result list tend to be long documents. As the far right plot shows, the participants work down the ranked list at a much slower rate than on average (far left plot). In addition, the participants show little difference in rank reached for the 0.3 and 0.6 lists. Also shown in the plot are the predictions of $T(k)$ for these two ranked lists. As can be seen here, $T(k)$ is not simply a linear function of rank. The different documents with their differing ranks and relevance judgments produce unique $T(k)$ values for each ranked list.

While our $T(k)$ correctly estimates a slower rate for Topic 383, it does overestimate the rate of progress, and it also predicts a gap between the ranks reached for the 0.3 and 0.6 lists. To illustrate the degree to which modeling of the topic could help future versions of $T(k)$, we calibrated $T(k)$ using only Topic 383. In the far right plot, we label this version of $T(k)$ the "Custom Topic 383 T(k)." As shown, the custom calibrated $T(k)$ much better reflects the actual participant behavior on Topic 383.

Given the importance of modeling the topic, a possibility for producing per-topic, custom calibrated versions of $T(k)$ would be to assign multiple assessors to a topic during test collection construction. The behavior of the assessors could be recorded and packaged alongside the relevance judgments, and then each topic could have its own $T(k)$.

To what extent does $T(k)$ vary on actual runs? Figure 3 shows the maximum, average, and minimum rank at time t for all of the submitted runs to the TREC 2005 Robust Track. As shown, there is a considerable range of ranks at a given time. In terms of Equation 3, this result means that at a given rank k, different ranked lists will vary in their predicted time to reach that rank and will have different amounts of decay, $D(t)$, applied at that rank.

5. TIME-BIASED GAIN

In this section, we pull together the pieces of the time-biased gain measure defined in Equation 3, suggesting specific formulae for the remaining components, including gain, decay and normalization.

5.1 Gain

For the remainder of the paper, we define

$$g_k = \begin{cases} P(C=1|R=1)P(S=1|R=1) & \text{if } r_i = 1 \\ 0 & \text{if } r_i = 0 \end{cases}$$

Under this definition, if the document at rank k is relevant, the gain is equivalent to the probability of viewing it and saving it. If the document is not relevant, the gain is zero. This definition is consistent with the binary relevance values typical of TREC tasks and adopted for our calibration process. In principle, we could adopt the graded gain values employed by nDCG, the attenuated gain values employed by ERR, or even gain values that take document length and other features into account. We leave the exploration of these ideas for future work.

Ignoring decay for now, we combine gain values with $T(k)$, our formula for estimated rank at time t, allowing us to compute $G(t)$ for individual runs. Figure 4 shows $G(t)$ for four runs taken from the TREC 2005 Robust Track, along with their official MAP values [39]. All four runs performed well, with `indri05RdmmT` and `uic0501` being the best and second-best title-only runs, and `ASUDE` and `indri05RdmeD` being the best and second-best description-only runs. For three of these runs, their relative cumulative gain remains consistent over the full time period. The story for `uic0501` is different. It leads the other runs for the first twenty minutes, but begins to fall behind after that, providing some insight into its behavior. While similar insight might be gained from a standard recall-precision plot, calibrating gain against time may provide a better sense of the performance experienced by a user.

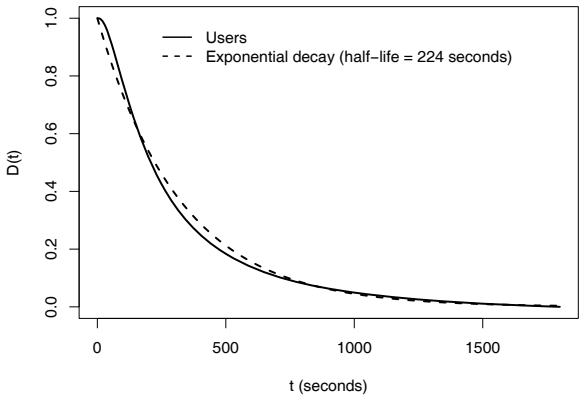

Figure 5: User data and fitted decay curve for $D(t)$.

5.2 Decay

Our calibration process provides no guidance regarding the form of the decay function $D(t)$, but standard exponential decay provides one possibility:

$$D(t) = e^{-t\frac{\ln 2}{h}}, \tag{6}$$

where h is the "half-life" of users, i.e., the time at which half of the initial users have stopped scanning the result list.

To support this choice of decay function, we turn to an interaction log taken from a commercial search engine, in this case from the MSN search engine. This anonymized log was made available to selected researchers across the information retrieval community during 2006 and 2007. For example, Zhang et al. [43] employed it to validate the discount function appearing in RBP. The log contains user interaction data for 5 million searches during May 2006.

We treat the time between the query and its last click as a proxy for the time spent scanning the result list and viewing documents, allowing us to estimate $D(t)$. Since our scenario assumes a careful and determined user, we try to filter out users with navigational needs, etc., by eliminating searches with less than five clicks. We also filter out searches that take longer than 30 minutes.

Figure 5 plots the $D(t)$ estimated from this log. As shown in the figure, exponential decay with a half-life of 224 seconds provides a good fit to this data. We use this half-life in the remainder of the paper.

We note that it should be possible to employ log data to estimate $T(k)$, as well as $D(t)$, although we would need document length and other data not present in this log. This approach would be particularly appropriate for calibration in the context of web search. We leave this idea for future work.

5.3 Interpretation

Before we move on to the normalization factor, we pause in our development of time-biased gain to consider one way in which the measure may be interpreted. Recall that g_k represents the *expected* gain at rank k and $T(k)$ represents the *expected* time to reach rank k. Let G_k be a random variable indicating the *actual* gain experienced by a particular user from the document at rank k, and let T_k be a random variable indicating the *actual* time taken by a user to reach rank k. Thus, we have

$$\begin{aligned}
\sum_{k=1}^{\infty} g_k D(T(k)) &= \sum_{k=1}^{\infty} E[G_k] D(E[T_k]) && (7)\\
&\leq \sum_{k=1}^{\infty} E[G_k] E[D(T_k)] && (8)\\
&= \sum_{k=1}^{\infty} E[G_k D(T_k)] && (9)\\
&= E\left[\sum_{k=1}^{\infty} G_k D(T_k)\right]. && (10)
\end{aligned}$$

Equation 8 holds by Jensen's inequality, since exponential decay is convex. Equation 9 holds since G_k and T_k are independent. Equation 10 is the number of documents a user is expected to save as relevant. Hence, if we ignore normalization by setting $\mathcal{N} = 1$, we may interpret the measure as a lower bound on this value.

5.4 Normalization

Evaluation measures are typically normalized so that their values fall into the range [0:1]. Depending on the measure, this normalization may be required for the measure to be reasonably averaged over multiple queries, at least if an arithmetic mean is used. For example, the raw summation appearing in MAP is normalized by dividing it by the number of known relevant documents, so that it treats "all queries equally" [25].

The nDCG measure bases its normalization on the set of known relevant documents. It computes a maximum value for the summation in Equation 1 over a ranked list of these documents, which becomes the value for $\mathcal{N}$. If we know all relevant documents in the collection, we may follow a similar approach for time-biased gain, ranking these relevant documents in the order that maximizes Equation 2. Under this approach, the value for $\mathcal{N}$ varies from topic to topic, according to the set of relevant documents known for that topic. Unfortunately, as it does for nDCG, this approach suffers from the need to know all relevant documents, and the value of the measure will change if new relevant documents surface.

RBP bases its normalization on an idealized collection containing an unlimited number of relevant documents. We may follow a similar approach for time-biased gain by assuming an unlimited number of zero-length relevant documents. If we set $T_x = T_S + T_D(0)P(C = 1|R = 1)$, then over this idealized collection we have $T(k) = T_x \cdot (k - 1)$, so that

$$\begin{aligned}
\mathcal{N} &= \sum_{k=1}^{\infty} g_k D(T(k)) && (11)\\
&= P(C = 1|R = 1)P(S = 1|R = 1) \sum_{k=0}^{\infty} e^{-T_x k \frac{\ln 2}{h}}\\
&= \frac{P(C = 1|R = 1)P(S = 1|R = 1)}{1 - e^{-T_x \frac{\ln 2}{h}}}
\end{aligned}$$

Under this approach, $\mathcal{N}$ is now constant across all queries. For our calibration values, $\mathcal{N} \approx 17.1$.

Given our interpretation of time-biased gain as a lower bound on the number of documents a user is expected to save as relevant, normalization may not be necessary, and in this paper, we set $\mathcal{N} = 1$. The creators of ERR, who

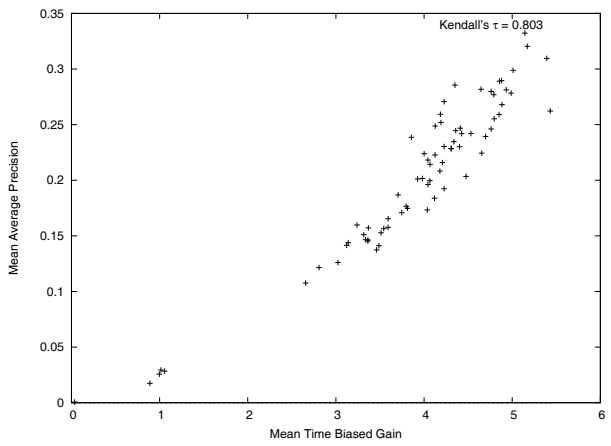

Figure 6: Time-biased gain compared against MAP over runs from the TREC 2005 Robust Task.

emphasize a similar interpretation, do not normalize that measure. In addition, normalization is not required if we average over topics using a geometric mean [25, 39]. We leave further exploration of normalization for future work.

6. COMPARISON

In this section, we provide a brief comparison between time-biased gain and other effectiveness measures used at TREC and elsewhere. Apart from time-biased gain, these measures are computed by the standard TREC evaluation program (`trec_eval`). All measures are computed over the runs submitted to the TREC 2005 Robust Track.

Figure 6 presents a scatter plot of MAP vs. time-biased gain. The measures are correlated, with a Kendall's τ of 0.794. Due to exponential decay, time-biased gain tends to emphasize early gain, a point illustrated by the run `uic0501`. Reflecting the behavior seen in Figure 4, it achieves the second-best performance, a time-biased gain value of 5.39 vs. a MAP value of 0.310, outperforming runs with higher MAP values.

With the best time-biased gain value of 5.43, and a MAP value of only 0.262, the run `sab05ror1` forms an interesting case study [6]. This run was trained on TREC vol. 4 & 5 qrels and documents and had very mixed performance across topics as measured by average precision. While the attempt appeared to fail when measured by MAP, the run's status as an obvious outlier suggests that the attempt may have succeeded after all (or at least warrants further investigation).

Sakai [29] proposes *discriminative power* as one method for assessing the behavior of effectiveness measures. While high discriminative power does not necessarily indicate a good measure, low discriminative power would raise questions about the measure. To calculate discriminative power, we compute a significance test between each pair of runs submitted to an experimental task, such as the TREC Robust Track. Discriminative power is the number of pairs that are significant at some fixed level. Table 2 compares the discriminative power of time-biased gain against several standard measures. MAP is known to have high discriminative power, due to the depth at which relevant document can influence its value. However, time-biased gain provides good discriminative power when compared to simple precision@10 and nDCG@K, for typical values of K.

Measure	t-test	Randomization	Bootstrap
MAP	66.0%	66.4%	67.2%
precision@10	51.6%	51.4%	52.7%
nDCG@10	49.5%	49.6%	51.0%
nDCG@20	54.5%	54.6%	55.6%
time-biased gain	57.5%	57.5%	58.8%

Table 2: Discriminative power of measures according a two-tailed paired t-test, a randomization test, and a bootstrap test (significance level 0.05).

7. CONCLUDING DISCUSSION

Equation 2 lies at the heart of our work. The remainder of the paper represents only one possible route for developing this equation. In following this route, we have adhered as closely as possible to the typical assumptions underlying a TREC task. We adopt the scenario of a careful and determined user, with a topical information need, intending to identify as much relevant material as they can in the time they have available. In another context, such as web search, another scenario will be more appropriate, requiring different calibration of the measure.

In following this route, we have also adhered as closely as possible to the typical assumptions underlying the creation of evaluation measures. Similar to the generic framework of Equation 1, time-biased gain is structured as a normalized sum over discounted gain values. Our primary innovation is to discount by time instead of rank.

However, a better numeric approximation to Equation 2 might be achieved through simulation. The user model described in Section 3 could easily be adapted for this purpose. By simulating thousands of users interacting with a result list, we may approximate both expected gain and variance. We leave this idea for future work.

Time-based calibration allows us to accommodate aspects of the search process that are ignored by traditional effectiveness measures. For example, in TREC collections longer documents are more likely to be relevant than shorter documents [31]. During the early years of TREC, considerable effort was made to adjust ranking functions according to document length [28, 31]. In light of the greater time required to judge longer documents, it may be that adjustment of the evaluation measures was also required.

We recognize that our approach to duplicates rewards behavior that might be viewed as undesirable by many users. Since we are following typical TREC guidelines for relevance, our calibration of time-biased gain rewards a system for returning duplicate documents, since they are faster to assess. In other contexts, it is likely that duplicates should provide no gain, thus penalizing runs through the extra time required to assess them. We considered following this approach in this paper, but decided to stay as true as possible to the TREC guidelines.

In this paper we focused on the population level behavior of determined, informational searchers. In future work, it would be good to study different classes of user behavior, and produce for each class a calibrated time-biased gain.

8. ACKNOWLEDGMENTS

We thank all of the anonymous reviewers for their comments. In particular, we thank Reviewer 3 for providing several pages of thoughtful feedback. Like Reviewer 3, we recognize the limitations of this paper's version of time-biased gain, and we plan to address these limitations in future work. Thanks to Ben Carterette and Falk Scholer for their helpful feedback on an early version of this paper. David Hu wrote the software to compute the sets of near-duplicate documents.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), in part by GRAND NCE, in part by Google, in part by an Amazon Web Services in Education Research Grant, in part by the facilities of SHARCNET, and in part by the University of Waterloo.

9. REFERENCES

[1] I. J. Aalbersberg. Incremental relevance feedback. In *SIGIR*, pp. 11–22. 1992.

[2] P. Arvola, J. Kekäläinen, and M. Junkkari. Expected reading effort in focused retrieval evaluation. *Information Retrieval*, 13:460–484, 2010.

[3] L. Azzopardi. The economics in interactive information retrieval. In *SIGIR*, pp. 15–24, 2011.

[4] L. Azzopardi, K. Järvelin, J. Kamps, and M. D. Smucker. Report on the SIGIR 2010 workshop on the simulation of interaction. *SIGIR Forum*, 44:35–47, 2011.

[5] A. Broder. On the resemblance and containment of documents. In *Compression and Complexity of Sequences*, pp. 21–29, 1997.

[6] C. Buckley. Looking at limits and tradeoffs: Sabir Research at TREC 2005. In *TREC*. 2005.

[7] B. Carterette. System effectiveness, user models, and user utility: A conceptual framework for investigation. In *SIGIR*, pp. 903–912, 2011.

[8] B. Carterette, E. Kanoulas, and E. Yilmaz. Simulating simple user behavior for system effectiveness evaluation. In *CIKM*, pp. 611–620, 2011.

[9] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *CIKM*, pp. 621–630, 2009.

[10] C. L. Clarke, N. Craswell, I. Soboroff, and A. Ashkan. A comparative analysis of cascade measures for novelty and diversity. In *WSDM*, pp. 75–84, 2011.

[11] W. S. Cooper. On selecting a measure of retrieval effectiveness. *JASIS*, 24(2):87–100, Mar/Apr 1973.

[12] W. S. Cooper. On selecting a measure of retrieval effectiveness: Part II. implementation of the philosophy. *JASIS*, 24(6):413–424, Nov/Dec 1973.

[13] M. D. Dunlop. Time, relevance and interaction modelling for information retrieval. In *SIGIR*, pp. 206–213. 1997.

[14] G. Dupret. Discounted cumulative gain and user decision models. In *SPIRE*, pp. 2–13. 2011.

[15] G. Dupret and B. Piwowarski. A user behavior model for average precision and its generalization to graded judgments. In *SIGIR*, pp. 531–538, 2010.

[16] N. Fuhr. A probability ranking principle for interactive information retrieval. *Information Retrieval*, 11:251–265, 2008.

[17] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM TOIS*, 20(4):422–446, 2002.

[18] D. Kelly. *Methods for Evaluating Interactive Information Retrieval Systems with Users*, now Publishers, 2009.

[19] H. Keskustalo, K. Järvelin, T. Sharma, and M. L. Nielsen. Test collection-based IR evaluation needs extension toward sessions: A case of extremely short queries. In *AIRS*, pp. 63–74, 2009.

[20] R. Khan, D. Mease, and R. Patel. The impact of result abstracts on task completion time. In *Workshop on Web Search Result Summarization and Presentation, WWW'09*.

[21] J. Lin and M. D. Smucker. How do users find things with PubMed? Towards automatic utility evaluation with user simulations. In *SIGIR*, pp. 19–26. 2008.

[22] L. Lorigo, M. Haridasan, H. Brynjarsdóttir, L. Xia, T. Joachims, G. Gay, L. Granka, F. Pellacini, and B. Pan. Eye tracking and online search: Lessons learned and challenges ahead. *JASIS*, 59(7):1041–1052, 2008.

[23] A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM TOIS*, 27(1):1–27, 2008.

[24] M. A. Najork, H. Zaragoza, and M. J. Taylor. HITS on the Web: How does it compare? In *SIGIR*, pp. 471–478, 2007.

[25] S. Robertson. On GMAP: and other transformations. In *CIKM*, pp. 78–83, 2006.

[26] S. Robertson. A new interpretation of average precision. In *SIGIR*, pp. 689–690, 2008.

[27] S. E. Robertson, E. Kanoulas, and E. Yilmaz. Extending average precision to graded relevance judgments. In *SIGIR*, pp. 603–610, 2010.

[28] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *SIGIR*, pp. 232–241, 1994.

[29] T. Sakai. Evaluating evaluation metrics based on the bootstrap. In *SIGIR*, pp. 525–532, 2006.

[30] T. Sakai and S. Robertson. Modelling a user population for designing information retrieval metrics. In *EVIA*, pp. 30–41, 2008.

[31] A. Singhal, G. Salton, M. Mitra, and C. Buckley. Document length normalization. *IPM*, 32(5):619–633, 1996.

[32] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language-model based search engine for complex queries (extended version). Tech Report IR-407, CIIR, University of Massachusetts Amherst, 2005.

[33] M. D. Smucker. An analysis of user strategies for examining and processing ranked lists of documents. In *HCIR*, 2011.

[34] M. D. Smucker and C. Jethani. Human performance and retrieval precision revisited. In *SIGIR*, pp. 595–602, 2010.

[35] L. T. Su. Evaluation measures for interactive information retrieval. *IPM*, 28:503–516, March 1992.

[36] A. Tombros and M. Sanderson. Advantages of query biased summaries in information retrieval. In *SIGIR*, pp. 2–10, 1998.

[37] A. Turpin, F. Scholer, K. Järvelin, M. Wu, and J. S. Culpepper. Including summaries in system evaluation. In *SIGIR*, pp. 508–515. 2009.

[38] A. H. Turpin and W. Hersh. Why batch and user evaluations do not give the same results. In *SIGIR*, pp. 225–231, 2001.

[39] E. M. Voorhees. Overview of the TREC 2005 Robust Retrieval Track. In *TREC*, 2005.

[40] R. W. White, I. Ruthven, J. M. Jose, and C. J. van Rijsbergen. Evaluating implicit feedback models using searcher simulations. *ACM TOIS*, 23(3):325–361, 2005.

[41] Y. Yang and A. Lad. Modeling expected utility of multi-session information distillation. In *ICTIR*, pp. 164–175, 2009.

[42] E. Yilmaz, M. Shokouhi, N. Craswell, and S. Robertson. Expected browsing utility for web search evaluation. In *CIKM*, pp. 1561–1564, 2010.

[43] Y. Zhang, L. A. Park, and A. Moffat. Click-based evidence for decaying weight distributions in search effectiveness metrics. *Information Retrieval*, 13:46–69, 2010.