Adaptive Persistence for Search Effectiveness Measures

Jiepu Jiang
Center for Intelligence Information Retrieval,
College of Information and Computer Sciences,
University of Massachusetts Amherst
jpjiang@cs.umass.edu

James Allan
Center for Intelligence Information Retrieval,
College of Information and Computer Sciences,
University of Massachusetts Amherst
allan@cs.umass.edu

ABSTRACT

Many search effectiveness evaluation measures penalize the importance of results at lower ranks. This is usually explained as an attempt to model users' persistence when sequentially examining results—lower ranked results are less important because users are less likely persistent enough to read them. The persistence parameters are usually set to cope with the target cohort and tasks. But during a particular evaluation round, the same parameters are applied to evaluate different ranked lists. In contrast, we present work that adapts the persistence factor according to the ranking and relevance of the ranked lists being evaluated. This is to model that rational users change their browsing behavior according to the search result page, e.g., users avoid wasting time (a low persistence level) if the results look apparently off-topic. Experimental results show that this approach better fits observed user behavior and correlates with users' ratings on their search performance.

KEYWORDS

Search effectiveness evaluation measure; persistence; user model.

1 INTRODUCTION

Accurately measuring the effectiveness of a search system needs to take into account not only the quality of retrieved results but also the possible ways that users may interact with the results. For example, many search effectiveness evaluation measures penalize the contribution of relevant results at lower ranks. This is because users are more likely to view top-ranked results on a search result page (SERP). Eye-tracking studies [30] observed less visual attention of users on lower-ranked results. Search log analysis [15, 30] also showed that higher-ranked results attract more clicks, although they are not necessarily more relevant.

Recent evaluation measures interpret such discounting components as models for users' browsing behavior. As users go to deeper ranks, they are less likely persistent enough to examine the results [27, 35]. Many measures include parameters for the degree of persistence. For example, discounted cumulated gain (DCG) [27] applies a discount factor $\frac{1}{\log_b(b+k-1)}$ to the kth result. A greater value of b penalizes results at lower ranks by a smaller extent, which stands for more persistent SERP browsing. Other examples include the persistence parameter p in rank-biased precision (RBP) [35], the half-life parameter h in time-biased gain (TBG) [40], and so on.

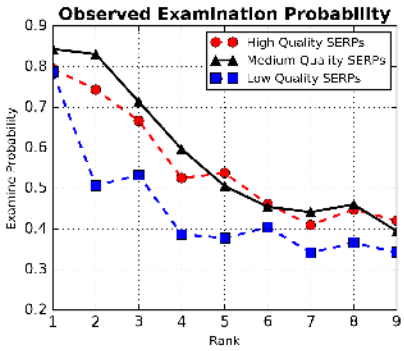

Figure 1: The probability of examining results at different ranks on SERPs with *high*, *medium*, and *low* "quality". We sort the SERPs by their DCG scores and divide them into five bins. *High*, *medium*, and *low* quality SERPs refer to those in the first, third, and fifth bins, respectively.

Previous work suggested to set these parameters according to the target users and tasks [27, 35]. For example, Moffat and Zobel [35] used $p = 0.95$ and 0.8 for "patient" users, and 0.5 for "impatient" ones in RBP. When user behavioral data (such as click logs) are available, one can tune these parameters according to the observed search behavior [40, 45]. Users may also have different levels of persistence in different scenarios (e.g., navigational and information queries [7] and tasks with different levels of complexity [3]). Despite these variabilities, these parameters are usually predetermined before evaluation—predetermined values are applied to evaluate different SERPs. This implicitly assumes that a user has the same persistence on various SERPs, which conflicts with our observations.

Figure 1 shows an example based on a laboratory user study's search log [29]. It plots users' probability of examining results at different ranks on SERPs with *high*, *medium*, and *low* "quality". We determine the user examined a result if we observed an eye fixation (captured by an eye-tracking device). Section 5 introduced details of the dataset. Figure 1 shows that the users have similar browsing patterns on the *high* and *medium* "quality" SERPs, but they are less likely to examine results on the *low* "quality" SERPs. Here we simply determine the quality of a SERP by their DCG scores. But we examined and found similar patterns using other measures as well. This suggests that users may adapt their browsing patterns according to the SERP (and particularly the quality of the SERP in this example)—for example, if the results look apparently low quality and off-topic, searchers quickly abandon rather than keep on examining more items.

CIKM'17, November 6–10, 2017, Singapore.
 ISBN 978-1-4503-4918-5/17/11…$15.00
DOI: https://doi.org/10.1145/3132847.3133033

Existing search effectiveness measures cannot explain such variability. For example, DCG and RBP's discount factors only depend on the rank of a result—following these models, SERPs with different "quality" should not vary in examination probability. Another popular measure, expected reciprocal rank (ERR) [9], sets the discount for a result adaptively according to the results at higher ranks. But it assumes that after examining a relevant result, users are less likely to continue to examine the next one due to the satisfaction of their information needs. Following ERR's model, searchers should have a higher chance to examine lower-ranked results on the *low* "quality" SERPs compared with on the *high* and *medium* "quality" ones, because the *low* "quality" SERPs have fewer relevant results at the top ranks—this is contradictory to our observations. Readers may refer to Figure 2 (plots labeled with "static") to the examination probability of existing measures on SERPs with different quality, where none of the plots fits with the observation in Figure 1.

In this paper, we adapt users' persistence based on the relevance and ranking of results on the SERPs (ranked lists) being evaluated (such that the evaluation measures and their browsing models are also adaptive to the SERPs). When evaluating different ranked lists, we compute different persistence values adaptively according to the results of the SERPs. Experimental results show that:

- Our approach helps existing evaluation measures, including DCG, RBP, ERR, TBG, and U-measure [37], to better fit with users' search behavior, including both the observed browsing behavior in an eye-tracking user study's log, and the clicking behavior in a commercial search engine's log.
- With more accurate user models, our approach also helps existing measures to better correlate with users' ratings on their search performance.

2 RELATED WORK

Ever since DCG [27], many search effectiveness measures included models for how users examine the ranked list (examination models). Our work is closely related to previous studies on this topic. Chapelle et al. [9] categorized the examination models in search effectiveness measures into position-based models (such as DCG and RBP [35]) and cascade models (such as ERR [9] and expected browsing utility [44]). We discuss both types of models and also a third type—cost-based models, including time-biased gain (TBG) [40] and U-measure [37]. Both measures consider users' examination behavior as dependent on the cost of examining results, which is usually measured by time spent or texts read by users. Another example of cost-based models is the Twist measure [20]. Section 3 analyzed current search effective measures in a deeper detail.

Our work adjusts the browsing models of search effectiveness measures according to the SERP being evaluated, which is closely related to many previous studies. For example, Kraft and Lee [31] introduced two stopping rules to the expected search length (ESL) measure: the satiation rule assumes persistent examination until enough relevant results have been found, while the disgust rule assumes users would stop after examining too many irrelevant results. de Vries, Kazai, and Lalmas [16] modeled that continuous examination of low quality (non-relevant) content leads to abandonment. Dupret and Piwowarski [17] modeled the chances of abandonment in click models based on both the rank of the result and its distance to the last clicked results. Effectiveness measures using cascade browsing models (such as ERR [10] and EBU [44]) all believe that examining a relevant result reduces the chances to examine the follow-up results. The INSQ family of measures [3, 33, 34] adapt the stopping probability based on user expectation and the current unmet information needs. Ferrante, Ferro, and Maistro [19] modeled users' stopping criteria based on the whole history of visited documents.

However, our work also differs from previous adaptive evaluation measures from two aspects. First, previous measures (such as ESL, ERR, UBM, INSQ and so on) only adapt browsing models based on the examined search results, while our method further adjusts the overall persistence level of users based on the whole SERP. This helps to model many observed search behavior, especially user abandonment when browsing a low-quality SERP (as shown in Figure 1). Second, our method learns adaptive models from observed user behavioral data (such as clicks or eye fixations) rather than relying on any particular assumptions on how browsing behaviors are adapted. Section 4.4 discusses the differences between our method and previous measures in better detail.

The way we train parameters for our adaptive persistence model is also similar to much previous work that calibrates parameters of search effectiveness measures based on click log or other user behavior data [7, 40, 45]. Another approach to help search effectiveness measures to better fit with users is the click model-based metrics [14]. In contrast to these work, our approach is different in that, 1) we model the adaptiveness of the browsing behavior, and 2) we focus on the persistence parameters in evaluation measures.

3 EXISTING BROWSING MODELS

Most current search effectiveness measures, either implicitly or explicitly, included a browsing model for how users interact with a ranked list of results [5]. This section reviews some typical models and their persistence factors.

When discussing a measure M, we focus on $P_{\mathrm{M}}(k)$, the probability of examining the kth result, as determined by M's browsing model. Here to *examine* a result means to look at its snippet on the SERP and to click on its link and read details if the user believes it is worthwhile. We use the following notations: r_k is the relevance grade for the kth result, and b_k is the binary version ($b_k = 1$ if relevant, otherwise 0); r_{max} is the highest relevance grade.

3.1 Position-based Models

Position-based models determine the probability to examine a result only based on its position (rank) on the SERP. Discounted cumulated gain (DCG) [27] and rank-biased precision (RBP) [35] are typical examples of position-based models.

A popular version of DCG [4] applies a discounting factor $\frac{1}{\log_b(b+k-1)}$ to the kth result in the ranked list, as in Equation 1. Most studies set $b = 2$, and in such a case the discount is $\frac{1}{\log_2(k+1)}$. DCG did not introduce any explicit model for how users browse the list of results, but we can consider the discounting factor as examination probability (as its value ranges from 0 to 1 when $b > 1$). b is the persistence parameter in DCG. A smaller value of b penalizes lower-ranked

results by a greater extent.

$$\text{DCG} = \sum_{k=1}^{n} \frac{2^{r_k} - 1}{\log_b(b + k - 1)} \ , \ P_{\text{DCG}}(k) = \frac{1}{\log_b(b + k - 1)} \quad (1)$$

RBP [35] explicitly introduced a browsing model. It assumes that users examine results on the SERP sequentially from top to bottom. Users always examine the first result. After examining each result, users have the chance p to examine the next one, and $1 - p$ to stop. Following this model, users have the probability p^{k-1} to examine the kth result, as in Equation 2. p controls users' persistence in browsing. A smaller p yields a greater discount effect.

$$\text{RBP} = (1 - p) \cdot \sum_{k=1}^{n} b_k \cdot p^{k-1} \ , \ P_{\text{RBP}}(k) = p^{k-1} \quad (2)$$

In Equation 2, b_k is the gain of the kth result, which was set to a binary function in the original RBP measure [35]. In our experiment, we set $b_k = 2^{r_k} - 1$ to consider graded relevance. This improves the measure's correlation with users' ratings on search performance.

3.2 Cascade Models

Cascade models in search effectiveness measures were motivated by the cascade click models [10, 15]. They model the chances of examining a result as dependent on previously examined results. More specifically, all existing cascade models [9, 44] believe that after examining a relevant result, users are more likely to stop browsing due to the satisfaction of their information needs, compared with the case of examining a non-relevant result.

Expected reciprocal rank (ERR) [9] uses a typical cascade model. ERR's browsing model is similar to RBP, but it models that after examining the kth result, users have the probability $s_k = \frac{2^{r_k} - 1}{2^{r_{max}}}$ to stop browsing due to the satisfaction of their information need. The chance of continuing to examine the next result is $1 - s_k$, which depends on the relevance of the examined result. Results with a higher level of relevance are more likely to satisfy users (a greater s_k), and thus penalize follow-up results by a greater extent.

The most popular form of ERR does not include a persistence factor, but Chapelle et al. [9] introduced an extended version of ERR that takes into account a similar factor: users stop examining (abandon) due to dissatisfaction. Equation 3 describes this variant[1]. γ is the chance to continue, and $1 - \gamma$ is the chance to abandon after examining a result. To examine the kth result, users should have neither stopped due to satisfaction nor abandoned at higher ranks. We consider γ as the persistence parameter in ERR. A smaller value of γ penalizes lower-ranked results by a greater extent.

$$\text{ERR} = \sum_{k=1}^{n} \frac{1}{k} \cdot s_k \cdot \gamma^{k-1} \cdot \prod_{m=1}^{k-1} (1 - s_m) \quad (3)$$

$$P_{\text{ERR}}(k) = \gamma^{k-1} \cdot \prod_{m=1}^{k-1} (1 - s_m)$$

[1] Chapelle et al. [9] did not include $1/k$ into this variant; we include $1/k$ in Equation 3 because this yields a better correlation with user experience ratings in our dataset.

3.3 Cost-based Models

Cost-based models discount a result by the expected cost spent by the users. The cost is usually measured in terms of time [40] or the length of examined texts [37]. These models penalize a result by a greater extent if the user has spent more effort when examining the result. Time-biased gain (TBG) [40] and U-measure [37] are typical examples of cost-based models.

TBG [40] penalizes a result based on the expected time spent to arrive at the result (before examining the result). The longer it takes to reach a result, the less likely users are persistent enough to examine it. Equation 4 computes TBG. g_k is the gain of the kth result. t_k is the expected time spent before examining the kth result. We can consider h as a persistence parameter. A greater h penalizes lower-ranked results by a smaller extent.

$$\text{TBG} = \sum_{k=1}^{n} g_k \cdot e^{-t_k \cdot \frac{\log 2}{h}} \ , \ P_{\text{TBG}}(k) = e^{-t_k \cdot \frac{\log 2}{h}} \quad (4)$$

U-measure [37] discounts a result based on the total length of texts users need to read to finish examining the result (including the texts for both the result itself and those at higher ranks). The more texts it takes to read to finish examining a result, the less likely users are persistent enough to examine it. Equation 5 computes U-measure. l_k is the cumulative length of examined texts starting from the first result to the kth result (inclusive). g_k is the gain of the kth result. We consider L as a persistence parameter. A greater L penalizes lower ranked results by a smaller extent.

$$\text{U} = \sum_{k=1}^{n} g_k \cdot \max(0, 1 - \frac{l_k}{L}) \ , \ P_{\text{U}}(k) = \max(0, 1 - \frac{l_k}{L}) \quad (5)$$

4 ADAPTIVE PERSISTENCE MODELS

4.1 Adaptive Persistence

As the last section summarized, many existing measures included parameters for users' persistence in SERP browsing, such as b in DCG, p in RBP, γ in ERR, h in TBG, and L in U-measure. Most existing methods use the same parameter to evaluate different SERPs during an experiment. In contrast, we set these parameters adaptively according to the ranking and relevance of results on the SERPs being evaluated. This is to model that users may have different persistence and browsing behavior on various SERPs.

Let s be a persistence parameter (e.g., s can be b in DCG, p in RBP, and etc.). We model s as a linear model based on the relevance of results at different ranks as in Equation 6: w_0 is a fixed term; w_{ij} is the weight for "the ith result has relevance grade j"; $[r_i = j]$ is a binary variable that takes the value 1 if $r_i = j$ (the ith result has relevance grade j), otherwise it is 0.

$$s = w_0 + \sum_{i=1}^{n} \sum_{j=0}^{r_{max}} w_{ij} \cdot [r_i = j] \quad (6)$$

When evaluating a SERP, we first compute s according to the results on the SERP and the parameters w_0 and w_{ij}. Then, we apply the calculated SERP-dependent persistence value s to the effectiveness measure to evaluate the SERP. Different SERPs may yield different persistence. Therefore, measures using such a SERP-dependent persistence are also adapted to the SERPs being evaluated. Note that s is only meant to be a computational model of persistence—we

do not intend to suggest that users will first scan all results on a SERP and then determine a persistence level for browsing.

The full model in Equation 6 has $n \cdot r_{max} + n + 1$ parameters in total. For a regular SERP design (10 results per page) and an evaluation protocol using five levels of relevance, s has 51 parameters. We can reduce the number of parameters by considering only a few top-ranked results (assuming that top-ranked results are more important for users' persistence). Another option is to consider only binary relevance rather than all relevance levels—for each rank k, s only includes two parameters for $[r_k = 0]$ and $[r_k > 0]$. These reduced models may help when we only have limited training data. If we only include a fixed term w_0, s is identical to the persistence parameters in existing measures.

In this paper, we model user's persistence (s) as only dependent on the ranking and relevance of results on a SERP. This simplifies the problem. Here we do not intend to suggest users' browsing behavior and persistence are only dependent on these factors. But such a model requires nothing more than the ranked list and relevance labels as input when evaluating a SERP. This makes it applicable to the Cranfield-style automatic evaluation approaches. Of course, we still need observed user interaction data to train parameters of s (w_0 and w_{ij}). But once the model has been trained, it can be applied to any unseen ranked lists as long as we have relevance judgments.

4.2 Parameter Estimation

4.2.1 Using Eye Tracking Data. A straightforward option for parameter estimation is to fit with observed browsing behavior. For example, when eye-tracking data is available, we can learn the parameters of s by maximizing the likelihood of the observed eye fixations on the SERP. Eye fixation refers to users' stably gaze at an area of the screen, which is widely used as a surrogate for users attention [30]. Many previous studies equate observing an eye fixation on a result's area to that the user examined the result.

Let v_k be a binary variable for whether or not we observed the user's eye fixation on the kth result. We use V_k for the chances of observing users' eye fixation on the kth result, as in Equation 7. n_v is a normalization factor between $P_{\mathrm{M}}(k)$ (examination probability) and V_k (the chances of observing an eye fixation). This is to take into account the fact that we do not always observe users' eye fixations on the first result, but most examination models assume that users always view the first result on the SERP. We estimate n_v as the chances to observe eye fixations on the top ranked result.

$$V_k = n_v \cdot P_{\mathrm{M}}(k) \tag{7}$$

Equation 8 computes the log likelihood (LL) of the observed eye fixations for a single SERP. The LL for multiple SERPs simply sums up the LLs for each SERP. For simplicity, we use the LL for an individual SERP in all following discussions.

$$\begin{aligned} LL_{view} &= \sum_{k=1}^{n} \log(V_k v_k + (1 - V_k)(1 - v_k)) \\ &= \sum_{k=1}^{n} \log((2v_k - 1)V_k - v_k + 1) \end{aligned} \tag{8}$$

Equation 8 is straightforward to maximize using approaches such as gradient ascent. Equation 9 computes the gradient. One can further derive $\frac{\partial P_k}{\partial w}$ for a specific measure according to its examination model. For example, Equation 10 derives the gradient for RBP. We do not further derive the gradients for other measures due to limited space.

$$\frac{\partial LL_{view}}{\partial w} = \sum_{k=1}^{n} \frac{(2v_k - 1) \cdot n_v}{(2v_k - 1)V_k - v_k + 1} \cdot \frac{\partial P_{\mathrm{M}}(k)}{\partial w} \tag{9}$$

$$\frac{P_{\mathrm{RBP}}(k)}{\partial w_0} = (k-1)p^{k-2}\,,\ \frac{P_{\mathrm{RBP}}(k)}{\partial w_{ij}} = (k-1)p^{k-2} \cdot [r_i = j] \tag{10}$$

4.2.2 Using Click Log. Collecting eye-tracking data is expensive, which makes it difficult to scale up. Therefore, a more practical option is to estimate the parameters using click log.

Let a_k be the "attractiveness" of the kth result (the chances of clicking on the result after examining its snippet). We can predict the likelihood of clicking on the kth result based on the examination model, as in Equation 11. This is often referred to as *examination hypothesis* [15, 17] in click models—click depends on both examination and the attractiveness of the result. C_k is the chances of clicking on the kth result, and c_k is the binary event that whether or not we observed any clicks on the kth result.

$$C_k = a_k \cdot P_{\mathrm{M}}(k) \tag{11}$$

Equation 12 computes the log likelihood of the observed clicks for an individual SERP. Similarly, Equation 13 derives the gradient, which is similar to Equation 9.

$$LL_{click} = \sum_{k=1}^{n} \log((2c_k - 1)C_k - c_k + 1) \tag{12}$$

$$\frac{\partial LL_{click}}{\partial w} = \sum_{k=1}^{n} \frac{(2c_k - 1) \cdot a_k}{(2c_k - 1)C_k - c_k + 1} \cdot \frac{\partial P_{\mathrm{M}}(k)}{\partial w} \tag{13}$$

Note that although Equation 11 looks similar to click models, our purpose here is not to achieve better click prediction or to compete with existing click models [10, 12, 17, 21]. Our purpose is only to set the parameters' values (w_0 and w_{ij}) appropriately through the process of click prediction. Also, the setting is also very different from those for training click models—our training process requires both clicks and relevance labels as input, while click models can be trained without relevance labels (and one of their primary purposes is to predict results' relevance labels).

We set a_k (attractiveness) only based on result relevance, i.e., $a_k = a(r_k)$. Based on the assumption that users always view the first result on a SERP, we estimate $a(r)$ as the click-through rate of results with the relevance grade r at the top rank.

4.3 Example

To better illustrate the proposed approach, we present an example of applying the adaptive persistence model to RBP. In the following example, persistence is modeled by considering graded relevance (0, 1, or 2) and the top 5 results. The following table shows the parameters' values estimated from a dataset.

We consider three example ranked lists (SERPs) L_1, L_2, and L_3. Their relevance vectors are as follows:

$$L_1 = [0, 0, 0, 0, 0]$$
$$L_2 = [1, 1, 1, 1, 1]$$
$$L_3 = [2, 2, 2, 2, 2]$$

$w_0 = 0.544$			
w_{ij}	$j = 0$	$j = 1$	$j = 2$
$i = 1$	0.047	0.088	0.059
$i = 2$	0.049	0.084	0.061
$i = 3$	0.048	0.096	0.050
$i = 4$	0.042	0.054	0.098
$i = 5$	0.052	0.072	0.070

The evaluation procedure is similar to those using regular RBP, except that the persistence p in RBP vary for different SERPs. When evaluating L_1, we first compute persistence based on L_1's relevance vector—$p = w_0 + w_{10} + w_{20} + w_{30} + w_{40} + w_{50} = 0.782$. Thus we apply $p = 0.782$ to evaluate L_1. Similarly, for L_2, we have $p = w_0 + w_{11} + w_{21} + w_{31} + w_{41} + w_{51} = 0.938$. For L_3, the persistence is $p = w_0 + w_{12} + w_{22} + w_{32} + w_{42} + w_{52} = 0.882$.

We apply adaptive persistence to the browsing models in existing measures and call them adaptive persistence browsing models and measures. Adaptive persistence browsing models and measures are variants for existing browsing models and measures where the persistence parameters are replaced with adaptive persistence, which varies adaptively according to the SERPs being evaluated.

4.4 Relation to Existing Measures

The measures we examined all discount the contribution of results at lower ranks. But the discount depends on different factors in various measures.

The discount components in position-based models are SERP independent. Position-based models determine the discount on the kth result only based on its rank k. For different SERPs, they set the same discount on the kth result without considering the results on the SERPs, which is oversimplified.

Cascade models and cost-based models determine the discount based on the results at higher ranks (e.g., the chances of stopping after examining results at higher ranks in the case of ERR, and the time to examine results at higher ranks in the case TBG). Therefore, their discount components are SERP dependent—for different SERPs, the discount for the kth result can be different depending on the results at higher ranks. But the dependency is *local*—they only take into account results at higher ranks than k.

The adaptive persistence model introduces a *global* dependency between SERP results and the browsing models. The discount factor depends on all the results on the SERP because we compute persistence based on all the results' relevance and their rankings. Adaptive persistence does not conflict with existing models such as cascade models and cost-based models but complements them. After applying adaptive persistence, all the three types of models are SERP dependent. The cascade models and position-based models, with the help of adaptive persistence, discount the contribution of a result based on both previously examined results (local dependency) and all the results on the SERP (global dependency). As later sections examined, such a global dependency between SERP results and the browsing models is helpful for evaluation measures.

Again, we note that our method is only a computational model of persistence—we assume that the persistence level of a user who is going to browse a SERP can somehow be inferred from the quality of the SERP. We leave the verification of this assumption and the explanation of the detailed mechanism for future work. Nevertheless, applying our method does not introduce additional risks because the parameters will be learned from user behavioral data—if users' persistence levels do not vary by SERPs, the learned model should come to similar persistence values for different SERPs.

5 DATASETS

We use two different datasets in our experiments:

- **J&A**[2]. This dataset was released by Jiang and Allan [28] based on a user study's search log [29]. It provides eye tracking data and users' ratings on their search performance in a session. We use the eye tracking data to verify how well the adaptive persistence models fit with users' browsing behavior. Also, we also examine how well search effectiveness measures applying the adaptive persistence models correlate with users' ratings on their search performance. Figure 1 was plotted based on this dataset.
- **Yandex**. This dataset is a subset of the Yandex relevance prediction challenge dataset[3]. The original purpose of the dataset was to evaluate click models regarding predicting results' binary relevance labels. We use this dataset to verify whether or not the adaptive persistence models better fit with observed clicking behavior compared with existing search effectiveness measures. Training the proposed adaptive persistence models requires both click and relevance labels. Thus we only select a subset of the dataset where each SERP was fully judged. 1,029,427 SERPs from 1,027,613 different sessions were selected in total.

Note that both datasets have some limitations. However, to the best of our knowledge, they are the most suitable open, accessible options for our purpose. The J&A dataset was collected in a laboratory user study setting. It is small in size (only 388 SERPs from 80 sessions). Also, the adopted search tasks came from the TREC session track [6], which included relatively more complex information needs than regular web search. In contrast, the Yandex dataset is more realistic because it comes from real commercial web search engines and is large enough for training robust models. But it does not offer eye tracking data and user experience ratings. Also, the log has been anonymized, which makes it impossible to assess the underlying search scenarios. Later sections discussed the implications of these limitations to the results. The following table shows some basic statistics of the two datasets.

	J&A	**Yandex**
# sessions	80	1,027,613
# SERPs	388	1,029,427
# results per SERP	9	10
Relevance levels	0–2	binary
Click	Yes	Yes
Eye tracking	Yes	No
User experience ratings	Yes	No
Experiment setting	Lab	Web search engine
Search system	Google	Yandex
Search task	Complex	Unknown

[2] https://github.com/jiepujiang/ir_metrics
[3] https://academy.yandex.ru/events/data_analysis/relpred2011/

6 EXPERIMENTS

6.1 Implementation

In the original TBG measure, Smucker and Clarke [40] estimated the time to examine a result based on the length of the document because the two correlate with each other in their dataset [41]. However, the Yandex dataset does not include document length. Besides, we did not find a significant correlation between the two in the J&A dataset ($r = 0.02$), but we observed a significant correlation between result relevance and dwell time ($r = 0.27$, $p < 0.001$). Similarly, in the Yandex dataset, users also spent significantly longer dwell time on relevant results compared with non-relevant ones (1104 vs. 774, $p < 0.001$). Therefore, we estimate the expected time to examine a result based on result relevance.

Equation 14 computes $t(r)$, the expected time to examine a result with the relevance r. It takes into account the time to read a result snippet (t_{snippet}), and the possible time spent on the result document if the user clicks on it. t_{snippet} is assumed a constant for all results. $P_{\text{click}}(r)$ is the chances of clicking on a result with relevance grade r, and $t_{\text{click}}(r)$ is the time spent on a result document after clicking.

$$t(r) = t_{\text{snippet}} + P_{\text{click}}(r) \cdot t_{\text{click}}(r) \tag{14}$$

The following table shows the time estimation in the two datasets (the Yandex dataset normalized time using an unknown unit). We estimate t_{snippet} based on the rank of the first clicked result on a SERP and the time spent from submitting the query to the first click [40]. $t_{\text{click}}(r)$ is estimated as the time spent from clicking on the result to the next recorded action in the search log (either submitting a query or clicking on a result). When computing TBG, we compute t_k based on $t(r)$, i.e., $t_k = \sum_{i=1}^{k-1} t(r_i)$ (note that t_k excludes the time to examine the kth result). We set $g_k = 2^{r_k} - 1$ in TBG, and we ignore the optional normalization component.

Time estimation in the J&A dataset.				
	t_{summary}	$P_{\text{click}}(r)$	$t_{\text{click}}(r)$	$t(r)$
$r = 0$	3.6 s	0.26	17.2 s	8.1 s
$r = 1$	3.6 s	0.50	30.7 s	19.0 s
$r = 2$	3.6 s	0.54	52.2 s	31.8 s

Time estimation in the Yandex dataset.				
	t_{summary}	$P_{\text{click}}(r)$	$t_{\text{click}}(r)$	$t(r)$
$r = 0$	74	0.51	774	471
$r = 1$	74	0.63	1104	765

Similarly, we compute a time-based variant for U-measure due to the lack of document length information in the Yandex dataset. Equation 15 computes this variant. Here t_{k+1} stands for the expected total time to reach the $(k+1)$th result (to be consistent with the t_k in TBG), which is computationally equivalent to the expected total time spent until the user finishes examining the kth result. The parameter T is similar to L in the original U-measure, except that it is measured in time. T is the persistence parameter in this variant. We set $g(k) = \frac{2^{r_k}-1}{2^{r_{max}}}$ as Sakai [37] did, and we also ignore the optional normalization factor in U-measure.

$$\text{U} = \sum_{k=1}^{n} g_k \cdot \max(0, 1 - \frac{t_{k+1}}{T}) \tag{15}$$

Note that for all the five measures we examined, their persistence parameters' values should stay within certain "reasonable" range, e.g., $b > 1$ in the case of DCG. But Equation 6 cannot guarantee this property. Thus, when computing persistence, we normalize the computed value to the closest valid value if it is not within the reasonable range. For DCG, we set $b = 1.01$ if the computed value ≤ 1. For RBP, we set $p = 0$ if the computed $p < 0$, and set $p = 1$ if the computed $p > 1$. For TBG and U-measure, we set h and T to 1 if the computed values < 1. For ERR, we set $\gamma = 0$ if the computed $\gamma < 0$. One exception is that we allow $\gamma > 1$ in ERR. This conflicts with the original notation of γ (the probability of continuing to examine the next result when the user was not satisfied), but yields better results. Section 6.3 discussed this issue in detail.

6.2 Experiment Condition

We apply adaptive persistence to the five measures' browsing models and compare with current ones where the persistence parameters are constant when evaluating different SERPs. We refer to the later *static persistence* models or measures. The purpose of the experiments is to examine:

- **RQ1**: how well the adaptive persistence browsing models explain observed browsing behavior compared with the static persistence ones (Section 6.3)
- **RQ2**: how well the adaptive persistence browsing models fit with observed clicking behavior compared with the static persistence ones (Section 6.4)
- **RQ3**: how well search effectiveness measures applying the adaptive persistence models correlate with users' ratings on their search performance compared with the static persistence ones (Section 6.5)

More specifically, we compare with two baselines:

- **Baseline 1** is the measures using "default" static persistence parameters. We set $b = 2$ in DCG, and $\gamma = 1$ in ERR. We set $p = 0.8$ and 0.5 in RBP, which were usually adopted for "patient" and "impatient" users [5, 35]. We set h to the "half life" of the users when they examine a SERP [40]. Sakai [37] set L to the largest maximal trail text length across all possible search sessions in the original U-measure. Similarly, we set T to the longest examine time for a SERP.
- **Baseline 2** is the browsing models using $s = w_0$ (only a fixed term). It is essentially the same as using static persistence parameters, but the values are trained using observed eye fixations or clicks.

6.3 Fitting Observed Browsing Behavior

To study RQ1, we first examine how well $V_k = n_v \cdot P_{\text{M}}(k)$ interpret the observed eye fixations in the J&A dataset. We use a cross-validation setting in experiments. We produce ten random partitions of the dataset. On each partition, we perform a 10-fold cross validation, using nine folds for training and one fold for testing. This produces results on 100 test folds in total. We report the mean negative log likelihood on these 100 test folds (smaller values are better).

Table 1 reports the results. The adaptive persistence models were trained using observed eye fixations. "top$k$" refers to adaptive persistence models considering only the top k results on the SERP. "Grade Relevance" and "Binary Relevance" stand for whether

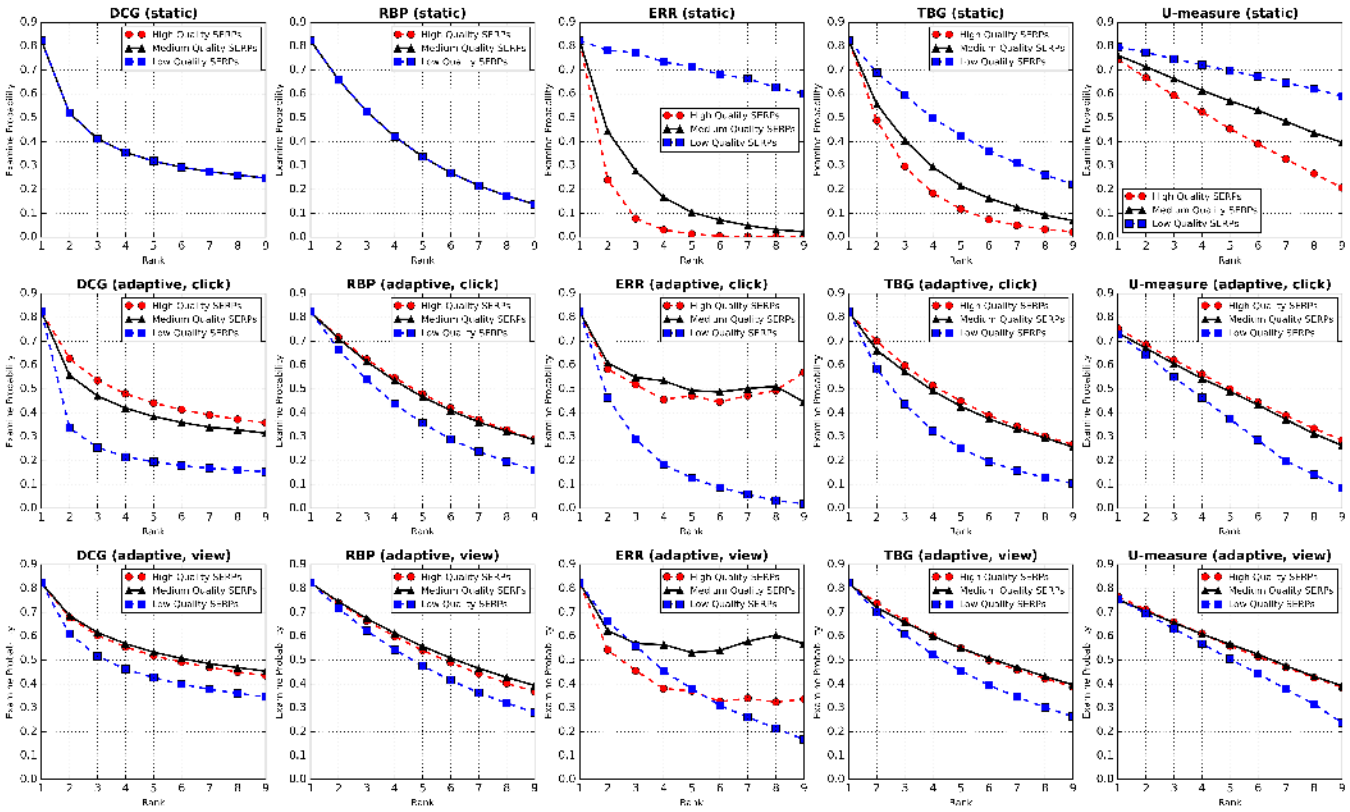

Figure 2: Predicted examination probability at different ranks on SERPs with *high*, *medium*, and *low* quality. (static) refers to measures using constant persistence parameter values on all SERPs. (adaptive, click) and (adaptive, view) refer to the adaptive measures trained using observed clicks and eye fixations, respectively.

Table 1: Fixed vs. adaptive persistence browsing models in predicting eye fixations, using the J&A dataset (negative log likelihood, smaller values are better).

	Baselines		Adaptive Persistence Measures (Graded Relevance)				Adaptive Persistence Measures (Binary Relevance)			
	1	2	top3	top5	top7	top9	top3	top5	top7	top9
DCG	247.5	223.7[1]	221.4[12]	222.2[12]	222.8[1]	222.6[1]	222.2[12]	222.1[12]	222.9[1]	222.6[1]
RBP, $p = 0.8$	252.7	224.3[1]	222.4[12]	223.6[1]	223.0[1]	223.6[1]	223.5[12]	223.9[1]	224.5[1]	224.6[1]
RBP, $p = 0.5$	514.6	224.3[1]	222.4[12]	223.6[1]	223.0[1]	223.6[1]	223.5[12]	223.9[1]	224.5[1]	224.6[1]
ERR	494.5	495.1[1]	338.6[12]	288.7[12]	299.9[12]	326.2[12]	356.3[12]	333.7[12]	337.2[12]	341.2[12]
TBG	269.7	231.7[1]	224.2[12]	221.9[12]	222.8[12]	221.7[12]	223.9[12]	221.4[12]	221.4[12]	220.8[12]
U-measure	235.4	235.3	228.9[12]	227.9[12]	226.9[12]	232.5	228.3[12]	226.6[12]	226.6[12]	223.6[12]

[1] and [2] indicate statistical significant differences at least at 0.05 level compared with baseline 1 and 2 by two-tail paired t-test.

or not the adaptive persistence models consider graded relevance. We compare with Baseline 1 and 2. Baseline 1 is a weak baseline here because the default persistence values are not trained to maximize the chances of observing the eye fixations. Baseline 2 is a solid baseline, which helps examine whether the SERP-dependent adaptive persistence outperforms trained SERP-independent static persistence.

Table 1 shows that for all the five measures' browsing models, using adaptive persistence consistently interprets observed eye fixations better than using the static baselines in the J&A dataset. This verifies that the proposed approach better fits with users' browsing behavior compared with the existing ones. Also, we noticed that the magnitudes of improvements seem larger for ERR, TBG, and U-measure compared with those for DCG and RBP.

Table 1 also shows that using more results and graded relevance in the adaptive persistence models do not necessarily have better performance in the J&A dataset. One possible reason is that considering many results and graded relevance levels increases the number of parameters, which requires larger datasets to train robust models. The reduced models are reasonable and effective alternatives to the full model in the case of limited training data.

To further understand why the adaptive persistence models perform better than the static ones, Figure 2 plots the estimated examination probabilities at different ranks on SERPs with *high*, *medium*,

Table 2: Static vs. adaptive persistence browsing models in click prediction, using the Yandex dataset ($\times 10^5$ negative log likelihood, smaller is better).

	Baselines		Adaptive Persistence		
	1	2	top3	top5	top10
DCG	3.660	2.640^{1}	2.548^{12}	$\mathbf{2.545}^{12}$	2.547^{12}
RBP, $p = 0.8$	3.651	2.819^{1}	2.738^{12}	2.733^{12}	$\mathbf{2.728}^{12}$
RBP, $p = 0.5$	2.864	2.819^{1}	2.738^{12}	2.733^{12}	$\mathbf{2.728}^{12}$
ERR	3.264	3.065^{1}	2.799^{12}	$\mathbf{2.798}^{12}$	2.927^{12}
TBG	3.214	2.866^{1}	2.692^{12}	2.681^{12}	$\mathbf{2.679}^{12}$
U-measure	4.319	4.248^{1}	4.167^{12}	3.987^{12}	$\mathbf{3.874}^{12}$

[1] and [2] indicate statistical significant differences at least at 0.05 level compared with baseline 1 and 2, respectively.

and *low* "quality". Here the setting is the same as that in Figure 1—we sort the SERPs by their DCG scores and refer to those in the first, third, and fifth bins as *high*, *medium*, and *low* "quality" SERPs. This is arbitrary, but we examined and found that using other measures would also produce similar results to Figure 2.

The first row shows the predicted examination probabilities by existing browsing models (using static persistence). None of the five models explain the differences in observed examination probabilities on SERPs with different "quality" (as in Figure 1). As we discussed, DCG and RBP have the same examination probabilities on various SERPs. In contrast, ERR assigns higher examination probability to lower-ranked results on the *low* "quality" SERPs compared with on the *high* and *medium* "quality" ones. The variants of TBG and U-measure are similar to ERR because relevant results have greater costs (require a longer time to examine) and discount lower-ranked results by a greater extent. This conflicts with the observed examination probabilities in Figure 1, where users are less likely to examine lower-ranked results on the *low* "quality" SERPs.

The second and the third rows show examination probabilities for browsing models with adaptive persistence trained using click and eye fixation data. All these figures better interpret the differences in observed examination probabilities on SERPs with different "quality". The adaptive persistence models were learned to correctly reduce the examination probabilities on the *low* "quality" SERPs. This further confirms that, as we expected, by modeling persistence based on the results on the SERP, the adaptive persistence models can help existing browsing models better fit with real users' browsing behavior. This also explains why the magnitudes of improvements are larger for ERR, TBG, and U-measure in the J&A dataset, because their browsing models diverge from the observed examination probabilities by greater extents compared with DCG and RBP in the J&A dataset.

Note that allowing $\gamma > 1$ is the key to make the technique work for ERR. As the first row of Figure 2 shows, the default $s_k = \frac{2^{r_k}-1}{2^{r_{max}}}$ sets a radical discount to the examination probability, which may not work well in scenarios other than navigational search. In such a case, allowing $\gamma > 1$ helps the model to better fit with the actual browsing behavior.

6.4 Fitting Observed Clicking Behavior

To study RQ2, we examine how well $C_k = a_k \cdot P_{\mathrm{M}}(k)$ interpret the observed clicks. We use a similar cross-validation setting as the last section. A limitation of the previous section is that the experiments are based on a small dataset (J&A), where the included search tasks may not be representative of typical web searches. Thus, we examine RQ2 using both the J&A and the Yandex datasets. Note that the purpose of the experiment is not to achieve better click prediction or compete with existing click models. Our goal is to evaluate the browsing models in search effectiveness measures. Since we use $C_k = a_k \cdot P_{\mathrm{M}}(k)$ to predict clicks for all browsing models, we expect the performance of click prediction can indicate the effectiveness of the browsing models.

Table 2 reports the click prediction performance of the browsing models in the Yandex dataset. Similar to the findings in the last section, Table 2 shows that browsing models with adaptive persistence explain users' clicking behavior significantly better than the two baselines. This further verifies the effectiveness of our approach in a larger, more representative, and robust dataset. This suggests that the variability of browsing behavior to SERPs with different quality is not restricted to the J&A dataset, but is generalizable to regular web search scenario as well.

Table 3 further reports the click prediction performance of the models in the J&A dataset. The findings are similar to Table 2 and that for predicting eye fixations. The adaptive persistence models have significantly better click prediction performance than both baseline 1 and baseline 2 in most cases except RBP. Also, reduced models help maintain high effectiveness in this small dataset.

Note that here we do not hope to suggest findings such as "DCG has a better browsing model than ERR". This is because the results only suggest the overall effectiveness of the models in predicting click behavior at all ranks. Practically, a better fit with users' behaviors at the top-ranked results may be more valuable due to the importance of top-ranked results.

6.5 Correlating with Users' Ratings

A major goal of search effectiveness measures is to serve as indicators for potential users' experience after they interact with the SERPs. With better browsing models, we expect the search effectiveness measures can better model and correlate with users' search experience. The J&A dataset offers users' ratings to their search performance in a session. After finishing a search session, users answered the question "*how well do you think you performed in this task*" using a five-point Likert scale from *very well* (5) to *very badly* (1).

To study RQ3, we examine the correlation between search effectiveness measures and users' ratings in a session. Note that the dataset only provides users' ratings for a session as a whole, while all the examined measures are for individual SERPs. Therefore, when examining a measure, we compute the measure's values on different SERPs for a session and use the mean value as an indicator of the session's quality. We examine how well the average value of the measure for different SERPs in a session correlate with user's rating for that session.

We generate 25 random partitions of the sessions and perform a 4-fold cross-validation on each partition. We use three folds (60 sessions' SERPs) to train the adaptive persistence models, and measure the correlation (Pearson's r) on the test fold (20 sessions). The setting is different from previous sections because we noticed that Pearson's r becomes less stable for a small number of test

Table 3: Fixed vs. adaptive persistence browsing models in click prediction, using the J & A dataset (negative log likelihood, smaller values are better).

	Baselines		Adaptive Persistence Measures (Graded Relevance)				Adaptive Persistence Measures (Binary Relevance)			
	1	2	top3	top5	top7	top9	top3	top5	top7	top9
DCG	173.6	171.6^{1}	171.0^{1}	170.7^{12}	$\mathbf{170.4}^{12}$	171.3^{1}	171.1^{1}	$\mathbf{170.3}^{12}$	170.9^{1}	171.3^{1}
RBP, $p = 0.8$	175.0	171.9^{1}	$\mathbf{172.2}^{1}$	172.8^{12}	173.2^{12}	174.4^{2}	$\mathbf{171.8}^{1}$	172.1^{1}	172.8^{12}	173.3^{12}
RBP, $p = 0.5$	258.1	171.9^{1}	$\mathbf{172.2}^{1}$	172.8^{12}	173.2^{12}	174.4^{12}	$\mathbf{171.8}^{1}$	172.1^{1}	172.8^{12}	173.3^{12}
ERR	269.6	265.0^{1}	214.3^{12}	$\mathbf{194.9}^{12}$	205.5^{12}	215.1^{12}	223.3^{12}	$\mathbf{215.0}^{12}$	217.6^{12}	224.3^{12}
TBG	183.1	175.2^{1}	173.5^{12}	$\mathbf{170.8}^{12}$	177.7	174.0^{1}	172.7^{12}	$\mathbf{170.1}^{12}$	170.2^{12}	170.4^{12}
U-measure	176.7	177.7^{1}	176.8	175.2^{2}	$\mathbf{173.5}^{12}$	179.0	177.1	175.5^{2}	$\mathbf{174.0}^{12}$	176.7

[1] and [2] indicate statistical significant differences at least at 0.05 level compared with baseline 1 and 2 by two-tail paired t-test.

Table 4: Comparison between baselines and adaptive persistence measures in correlating with users' ratings on search performance (mean values of Pearson's r over 100 different test folds; greater values are better).

	Baselines		Adaptive Persistence Measures (Graded Relevance)				Adaptive Persistence Measures (Binary Relevance)			
	1	2	top3	top5	top7	top9	top3	top5	top7	top9
DCG	0.381	0.378^{1}	0.382^{2}	0.392^{12}	$\mathbf{0.396}^{12}$	0.394^{12}	0.384^{2}	0.391^{12}	$\mathbf{0.392}^{12}$	0.392^{12}
nDCG	0.340	0.332^{1}	0.337^{12}	0.346^{12}	$\mathbf{0.348}^{12}$	0.346^{2}	0.336^{12}	0.342^{2}	$\mathbf{0.343}^{2}$	0.343^{2}
RBP, $p = 0.8$	0.393	0.386^{1}	$\mathbf{0.411}^{12}$	0.392	0.361^{12}	0.354^{12}	$\mathbf{0.413}^{12}$	0.402^{2}	0.400^{2}	0.401^{2}
RBP, $p = 0.5$	0.376	0.386	$\mathbf{0.411}^{12}$	0.392	0.361^{2}	0.354^{2}	$\mathbf{0.413}^{12}$	0.402^{12}	0.400^{12}	0.401^{12}
ERR	0.364	0.367^{1}	0.420^{12}	$\mathbf{0.449}^{12}$	0.415^{12}	0.389^{12}	0.358	$\mathbf{0.391}^{12}$	0.376	0.385^{12}
TBG	0.379	0.375	0.387^{12}	0.400^{12}	0.409^{12}	$\mathbf{0.415}^{12}$	0.387^{12}	0.397^{12}	0.401^{12}	$\mathbf{0.407}^{12}$
U-measure	0.365	0.365	0.362	$\mathbf{0.378}^{12}$	0.372^{12}	0.362	0.358^{12}	$\mathbf{0.377}^{12}$	0.376^{12}	0.373^{12}

[1] and [2] indicate statistical significant differences at least at 0.05 level compared with baseline 1 and 2 by two-tail paired t-test.

instances (if we use the same setting as previous sections, each test fold includes only eight sessions). Table 4 reports the results, where the adaptive persistence models are trained using clicks (this stands for a more realistic choice compared with eye fixation). We also apply the approach to normalized DCG (nDCG). The process of training the persistence parameters is the same as that for DCG. However, when computing nDCG, the ideal DCG is computed using the ideal ranked list's persistence, which may be different from that for the ranked list being evaluated.

Table 4 shows that after applying adaptive persistence, all the five search effectiveness measures achieve significantly better correlations with users' ratings on their search performance compared with both baseline 1 and baseline 2. This confirms the usefulness of the proposed approach—with better user interaction models, our approach helps existing search effectiveness measures better correlate with users' perceptions on their search performance.

In addition, we noticed that, although baseline 2 unsurprisingly outperformed baseline 1 in interpreting observed user behavior, it does not necessarily lead to better correlations with users' ratings. For DCG, nDCG, and RBP ($p = 0.8$), baseline 2 yields slightly weaker correlations compared with baseline 1. This suggests that it requires further investigations on when and to what extents correlating with user behavior helps measures to model user experience.

7 DISCUSSION AND CONCLUSION

Accurately measuring the effectiveness of search systems is a key challenge to ensure consistent improvements of search quality—as search systems are usually trained to optimize some search quality indicators, they would fail if the quality indicators fail. However, many search effectiveness measures do not correlate with actual search quality well enough [2, 26, 38, 41–43]. This makes many search engine companies to rely on online evaluation techniques such as user experience prediction [1, 11, 18, 22, 23, 32] and interleaved experiments [8, 13, 24, 25, 36, 39] to determine whether or not to deploy a new ranking algorithm. Despite these issues, the Cranfield-style evaluation and search effectiveness measures are still important in IR evaluation and system design due to their automatic nature, which makes them suitable for automatically guiding system optimization.

This paper proposed and examined adaptive persistence model, a technique to improve many offline search effectiveness measures. This model deals with the issue of user behavior variability caused by SERP results. It adapts the browsing models in existing search effectiveness measures according to the SERPs being evaluated. Experiments show our approach is fruitful and helpful, concerning both fitting observed user behavior and correlating with users' ratings on their search experience. The technique is also generic, as it can be applied to different search effectiveness measures as long as they included such a persistence parameter. Our study also covers all the main user models in search effectiveness measures, including position-based, cascade, and cost-based ones.

A key difference between our work and current measures lies in that we take into account a global dependency between users' browsing behavior and the SERPs being evaluated. In contrast, position-based models (such as DCG and RBP) are independent of the SERPs. Cascade models (e.g., ERR) and cost-based models (e.g., TBG and U-measure) are also adaptive to the SERPs being evaluated, but they only consider a local dependency—where the discount on the kth result only depends on the previously exampled results (results at higher ranks). As we showed, our approach does not

conflict with existing measures but helps them better simulate user behavior, and consequently better correlate with search quality. This indicates that it is necessary to take into account such a global dependency in browsing models and search effectiveness measures.

It should be noted that, although motivated by the differences in examination probabilities on SERPs with different "quality", our approach is not restricted to the example we observed in the J&A dataset. The model does not rely on any specific assumptions about how different SERPs would differ in examination probabilities. It learns to adapt to such differences and thus can be generalized to different cases. As long as users' browsing behavior exist variability on various SERPs and such variability is related to the relevance of results, our model has the chance to learn the dependency. As the experimental results on the Yandex dataset show, our approach also explains click behavior significantly better in a very different dataset than the J&A dataset.

However, we also acknowledge the limitation of our work. First, it remains unclear how to interpret the dependency between browsing behavior and SERP results. A possible interpretation for the low examination probabilities on the *low* "quality" SERPs is that users quickly abandon to avoid wasting time. However, this requires further verification. It is also unclear whether other reasons exist. Second, both the two datasets have certain limitations—the J&A dataset is small, and the Yandex dataset is anonymized and uses only binary relevance. Therefore, it requires experiments on other datasets to fully examine the effectiveness of our approach.

Resources related to this study can be accessed online[4].

ACKNOWLEDGMENT

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-0910884. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] M. Ageev, Q. Guo, D. Lagun, and E. Agichtein. Find it if you can: A game for modeling different types of web search success using interaction data. In *SIGIR '11*, pages 345–354, 2011.

[2] A. Al-Maskari, M. Sanderson, P. Clough, and E. Airio. The good and the bad system: Does the test collection predict users' effectiveness? In *SIGIR '08*, pages 59–66, 2008.

[3] P. Bailey, A. Moffat, F. Scholer, and P. Thomas. User variability and IR system evaluation. In *SIGIR '15*, pages 625–634, 2015.

[4] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML '05*, pages 89–96, 2005.

[5] B. Carterette. System effectiveness, user models, and user utility: A conceptual framework for investigation. In *SIGIR '11*, pages 903–912, 2011.

[6] B. Carterette, P. Clough, M. Hall, E. Kanoulas, and M. Sanderson. Evaluating retrieval over sessions: The TREC session track 2011–2014. In *SIGIR '16*, pages 685–688, 2016.

[7] B. Carterette, E. Kanoulas, and E. Yilmaz. Simulating simple user behavior for system effectiveness evaluation. In *CIKM '11*, pages 611–620, 2011.

[8] O. Chapelle, T. Joachims, F. Radlinski, and Y. Yue. Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems*, 30(1):6:1–6:41, 2012.

[9] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *CIKM '09*, pages 621–630, 2009.

[10] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *WWW '09*, pages 1–10, 2009.

[11] Y. Chen, Y. Liu, K. Zhou, M. Wang, M. Zhang, and S. Ma. Does vertical bring more satisfaction?: Predicting search satisfaction in a heterogeneous environment. In *CIKM '15*, pages 1581–1590, 2015.

[12] A. Chuklin, I. Markov, and M. de Rijke. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 7(3):1–115, 2015.

[13] A. Chuklin, A. Schuth, K. Hofmann, P. Serdyukov, and M. de Rijke. Evaluating aggregated search using interleaving. In *CIKM '13*, pages 669–678, 2013.

[14] A. Chuklin, P. Serdyukov, and M. de Rijke. Click model-based information retrieval metrics. In *SIGIR '13*, pages 493–502, 2013.

[15] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *WSDM '08*, pages 87–94, 2008.

[16] A. P. de Vries, G. Kazai, and M. Lalmas. Tolerance to irrelevance: A user-effort oriented evaluation of retrieval systems without predefined retrieval unit. In *RIAO '04*, pages 463–473, 2004.

[17] G. E. Dupret and B. Piwowarski. A user browsing model to predict search engine click data from past observations. In *SIGIR '08*, pages 331–338, 2008.

[18] H. A. Feild, J. Allan, and R. Jones. Predicting searcher frustration. In *SIGIR '10*, pages 34–41, 2010.

[19] M. Ferrante, N. Ferro, and M. Maistro. Injecting user models and time into precision via Markov chains. In *SIGIR '14*, pages 597–606, 2014.

[20] N. Ferro, G. Silvello, H. Keskustalo, A. Pirkola, and K. Järvelin. The twist measure for IR evaluation: Taking user's effort into account. *Journal of the Association for Information Science and Technology*, 67(3):620–648, 2016.

[21] F. Guo, C. Liu, A. Kannan, T. Minka, M. Taylor, Y.-M. Wang, and C. Faloutsos. Click chain model in web search. In *WWW '09*, pages 11–20, 2009.

[22] A. Hassan. A semi-supervised approach to modeling web search satisfaction. In *SIGIR '12*, pages 275–284, 2012.

[23] A. Hassan, R. Jones, and K. L. Klinkner. Beyond DCG: User behavior as a predictor of a successful search. In *WSDM '10*, pages 221–230, 2010.

[24] K. Hofmann, F. Behr, and F. Radlinski. On caption bias in interleaving experiments. In *CIKM '12*, pages 115–124, 2012.

[25] K. Hofmann, S. Whiteson, and M. D. Rijke. Fidelity, soundness, and efficiency of interleaved comparison methods. *ACM Transactions on Information Systems*, 31(4):17:1–17:43, 2013.

[26] S. B. Huffman and M. Hochster. How well does result relevance predict session satisfaction? In *SIGIR '07*, pages 567–574, 2007.

[27] K. Järvelin and J. Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In *SIGIR '00*, pages 41–48, 2000.

[28] J. Jiang and J. Allan. Adaptive effort for search evaluation metrics. In *ECIR '16*, pages 187–199, 2016.

[29] J. Jiang, D. He, and J. Allan. Searching, browsing, and clicking in a search session: Changes in user behavior by task and over time. In *SIGIR '14*, pages 607–616, 2014.

[30] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR '05*, pages 154–161, 2005.

[31] D. Kraft and T. Lee. Stopping rules and their effect on expected search length. *Information Processing & Management*, 15(1):47–58, 1979.

[32] Y. Liu, Y. Chen, J. Tang, J. Sun, M. Zhang, S. Ma, and X. Zhu. Different users, different opinions: Predicting search satisfaction with mouse movement information. In *SIGIR '15*, pages 493–502, 2015.

[33] A. Moffat, F. Scholer, and P. Thomas. Models and metrics: IR evaluation as a user process. In *ADCS '12*, pages 47–54, 2012.

[34] A. Moffat, P. Thomas, and F. Scholer. Users versus models: What observation tells us about effectiveness metrics. In *CIKM '13*, pages 659–668, 2013.

[35] A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, 27(1):2:1–2:27, 2008.

[36] F. Radlinski and N. Craswell. Optimized interleaving for online retrieval evaluation. In *WSDM '13*, pages 245–254, 2013.

[37] T. Sakai and Z. Dou. Summaries, ranked retrieval and sessions: A unified framework for information access evaluation. In *SIGIR '13*, pages 473–482, 2013.

[38] M. Sanderson, M. L. Paramita, P. Clough, and E. Kanoulas. Do user preferences and evaluation measures line up? In *SIGIR '10*, pages 555–562, 2010.

[39] A. Schuth, F. Sietsma, S. Whiteson, D. Lefortier, and M. de Rijke. Multileaved comparisons for fast online evaluation. In *CIKM '14*, pages 71–80, 2014.

[40] M. D. Smucker and C. L. Clarke. Time-based calibration of effectiveness measures. In *SIGIR '12*, pages 95–104, 2012.

[41] M. D. Smucker and C. P. Jethani. Human performance and retrieval precision revisited. In *SIGIR '10*, pages 595–602, 2010.

[42] A. Turpin and F. Scholer. User performance versus precision measures for simple search tasks. In *SIGIR '06*, pages 11–18, 2006.

[43] A. H. Turpin and W. Hersh. Why batch and user evaluations do not give the same results. In *SIGIR '01*, pages 225–231, 2001.

[44] E. Yilmaz, M. Shokouhi, N. Craswell, and S. Robertson. Expected browsing utility for web search evaluation. In *CIKM '10*, pages 1561–1564, 2010.

[45] Y. Zhang, L. A. Park, and A. Moffat. Click-based evidence for decaying weight distributions in search effectiveness metrics. *Information Retrieval*, 13(1):46–69, 2010.

[4] https://ciir.cs.umass.edu/downloads/adaptive_metric/