Simulating Ideal and Average Users

Matthias Hagen, Maximilian Michel, and Benno Stein

Bauhaus-Universität Weimar, Germany
<first name>.<last name>@uni-weimar.de

Abstract We propose a framework for deterministic simulation of user behavior that allows to analyze the cost-gain-based performance on single result lists or whole search sessions. The ideal user representing optimal behavior (i.e., most gain with lowest effort) is contrasted with more "average" users that employ the spreading activation model from cognitive theory. On TREC Session Track data, the ideal user achieves about double the gain of real users at the same costs while the average gain of our different simulated users correlates well with the session-DCG metric—another argument for that metric in session-based evaluation.

1 Introduction

Analyzing search logs is a common way to study users and their information needs and also for evaluating search systems in for instance A/B tests—assuming that users more likely click on relevant documents. However, such evaluations require huge user populations that the commercial web search engines certainly have but that are lacking in many other settings (e.g., enterprise search or academic research). To overcome this problem of scarce user data, simulating user behavior got more prominent over the last years [14,15]. We propose a framework to deterministically simulate user behavior over search sessions in cost-gain-based scenarios. Our focus is on the click and result list switching behavior leaving the integration of simulated query formulation for future work. One contribution is the ideal user with optimal behavior (e.g., clicking on only those results that lead to some gain). In contrast, we also contribute more "average" users who employ a cognitive model to base click decisions on the shown result snippets. Furthermore, given pre-defined queries of a search session, the user models also decide when to switch to the next query. Each session is restricted by a predefined cost budget (e.g., time-based), every action (clicking, querying, reading) comes with some costs. Therefore, the simulated users assess each decision not only by its potential benefits in form of information gain, but also according to the accompanying costs. We compare the simulated users to real users on TREC session track data and show that the average information gain of our models highly correlates with the session-DCG measure often used in evaluation. Interestingly, the ideal user achieves about double the performance of real users at the same costs.

2 Related Work

We briefly review the literature on search evaluation and user modeling; more references follow in the sections detailing our approach.

Search Evaluation Over time, the measures for evaluating search results have changed from precision and recall to more rank-oriented metrics. One first example is MAP (mean average precision): the precision is measured at the ranks of the relevant results. The underlying assumption of MAP in form of a user model would be that the user clicks on only the relevant results and stops when all relevant documents have been visited—a scheme we will use in our simulated ideal user. Alternatives to MAP are normalized discounted cumulative gain (nDCG) [24] where results have different relevance levels (i.e., information gain) and

lower ranked results are less likely to be seen (i.e., discounted gain) or expected reciprocal rank (ERR) [16] following a cascading model where the probability that a user views a result depends on its rank position and a stopping criterion. In order to evaluate whole search sessions, Järvelin et al. also introduced a session-variant of nDCG [25] with the results of later queries having discounted gains. In our simulation framework we employ a cascading scheme with cost-based stopping criteria but instead of discounting gain for lower ranks—except that we assume no gain from showing the same or similar results again—, we take the higher costs for viewing lower-ranked results into account.

Over the last years, several user studies found that MAP has a weak correlation with real user performance [41], that the information gain of real users correlates with the precision overall [37], and that the preference for some ranking strongly correlates with its nDCG and ERR score [34]. Although the experimentation setup usually does not resemble the process of a real web search, many studies agree that evaluation metrics like ERR resemble the users' performance in general, but they also claim that Cranfield-style evaluation metrics lack realism and sound user models [36]. As a more realistic metric, simulation-based time-biased gain (TBG) was recently proposed [36]. Each user action (view summary or document, save document) comes with a time-based cost in a semi-Markov model (initialized with data from 48 real users who solved some pre-defined tasks within 10 minutes). The simulation is then used to estimate the information gain for different time limits and rankings and the performance variance. This idea very much inspired our scheme but instead of non-deterministic users we simulate more "general" deterministic user types reflecting the ideas of existing standard evaluation metrics. Our framework allows to compare an optimal or average deterministic user (i.e., perfect or average decisions) to a real user and to measure the spread of the gain differences of optimal and average behavior.

User Modeling User modeling deals with predicting and explaining user behavior and intentions. For instance, O'Brien and Keane [31] compare clicks predicted by the SNIF-ACT spreading activation model of information scent [21] to real users. They show that a cascading threshold strategy (top-down assessment of search results, clicking if result is above some threshold) is more common among users and that it is favorable to a comparative strategy (first assessing all snippets, then clicking on the most relevant). We will employ both, thresholding and spreading activation, in two of our user models. But in addition to O'Brien and Keane's model we also take switching to another query into account. User click models describe the click behavior while interacting with a search engine. Such models can be used to infer document preferences from the click-through rates in query logs [17]. In contrast, Zhang et. al claim that user behavior is related to the information task as a whole and therefore, the click behavior depends on previous queries and clicks for the same information task [42]. Consequently, task-centric click models use the complete search session in order to infer the relevance of results (e.g., duplicate results are less likely to be clicked again)—an idea we adopt for our simulation. Still, probabilistic click models are not really applicable in our scarce-user scenario since they typically rely on the availability of huge search logs and we aim for deterministic models instead.

3 Our General User Model

An information-seeking user approaches a search engine to satisfy an information need. For non-trivial tasks, the user typically submits several queries, scans their results and clicks on the ones whose snippets appear to be relevant—forming a search session. In this section, we propose a general user model that represents the space of all interaction sequences (we call them *paths*) a user might follow in a search session. Typically, search sessions are characterized by the respective query reformulations [22]. Note however, that we will concentrate on how users navigate through the result lists of a search session and we will not simulate query (re-)formulation.

3.1 The Framework

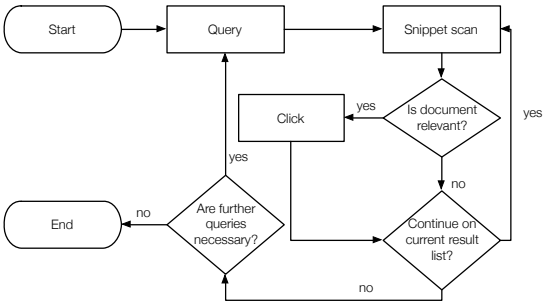

Figure 1. Flowchart of our general user model.

Basic assumption of our general user model is that a user wants to gain information in order to satisfy an information need against a retrieval system. The respecitve interactions come with certain costs (usually time but it could also be monetary charges for API querying etc.). The user has to find a trade-off between costs and benefits since the total "budget" for a search session typically is limited; leading to cost-driven behavior [3,4,7,8]. Our set of possible actions is similar to the elementary action types of Baskaya et al. [10]. Each session S consists of at least an initial query q_1, and a potentially empty list of subsequent queries q_2 to q_n. Each query q has an associated cost $cost_q(|q|)$ that depends on the length of the query (assumption: longer queries require more "effort"). After a query is submitted, the retrieval system returns a ranked result list with short snippets. The user starts scanning those snippets from top to bottom. Each scan of a snippet s has an associated cost $cost_{sc}$ that we assume to be a constant (assuming snippets of about equal length but non-constant length-dependence is also possible). In our model, at least one snippet is scanned following a query before another action can be performed. From scanning a snippet, the user estimates the result's relevance. If the result appears to be relevant, the user clicks on it. Each click c has some cost $cost_{cl}$ that we also assume to be constant (variable cost again is not difficult). A click leads to an information gain corresponding to the result's relevance level rel (i.e., the total gain is achieved with just one click assuming the whole document to be "read" at once) with one exception: no gain from a second click on the same or a similar result (cosine similarity). Consequently, relevance and thus click decisions not only depend on snippet relevance assessment but also on the previous clicks. After each snippet scan and after each click, the user decides if they proceed with scanning the next snippet or if they submit a new query. A search session ends when there are no further queries necessary or a given cost budget is reached—of course, the budget should suffice for at least submitting all pre-defined queries. Following others [11,30,38], Figure 1 depicts an abstract flowchart of our general user model including three kinds of decisions: (1) whether to click on a result, (2) whether to submit a new query, and (3) whether to end the session. Our simulated users instantiate schemes for those three decisions.

3.2 Restrictions of the General Model

Our general user model forms an abstraction of complex cognitive processes that might differ from user to user; consequently, not all possible search behavior can be expressed within our general model. For instance, the only way to cumulate information gain in our model is to click on a new result after a snippet scan. However, the user may already find the desired information in the snippet—a case we do not include in the current abstraction. We also assume a top-down processing of the result lists, starting with the first item in the result list. Through eye-tracking studies, Klöckner et al. found that this depth-first strategy is used by a majority of users [28]. Still, our user model does not represent the other around 15% of users. Furthermore, in our general model the users assess a document's relevance right after scanning its snippet and click on it if the relevance exceeds some threshold. This is in line with studies of O'Brien et al. who show that thresholding is the most common user strategy [31] but the information foraging theory, for instance, states that users might

also first assess all results and then decide to click on the one with the most gain [32]—a strategy that we do not model. Finally, we assume a cascading scheme where the user does not go back to a previous result list. The only way to see the results again would be to submit the same query (at the same costs). Such back-and-forth switching at lower costs is an interesting future simulation direction—also for query suggestion evaluation.

4 The Ideal User

First, we propose to simulate an ideal user: accumulating the most information gain for a certain cost "budget." Given the interaction costs and a search session with result lists and relevance judgments, the task is to find an optimal sequence of interactions within our general model. We call an interaction sequence a *path* through the state space formed by a session. A state is characterized by the lowest click or snippet scan in the different result lists and by the result currently in focus. Possible interactions form the edges connecting such states. The path of the ideal user shows how deep in the individual result lists a perfect behavior would scan snippets and which results should be clicked.

According to our general user model, three kinds of decisions have to be instantiated: clicking, switching to the next query, and ending the search session. Remember that we do not model query formulation but require pre-defined queries. The knowledge of the query sequence is used for the stopping criterion. We assume that each query of the sequence is submitted such that the user can only finish a session on that last query. Since the ideal user only clicks on results that lead to some gain, the crucial point of modeling the ideal user is the decision of when to change to a new query result list—very recently, independent of our investigations, optimal switching has also been investigated by Smucker and Clarke in a slightly different context [35].

Let l denote the rank in the result list R at which the ideal user stops scanning and switches to the next query (e.g., $l = 10$ means scanning the first 10 snippets). Whenever the ideal user encounters a result $r \in R$ not similar to a previously clicked document with a relevance level $rel(r)$ above a relevance threshold τ_{rel}, a click on the result is performed at the click cost $cost_{cl}$. The document is then added to the list $Clicked$ of clicked documents. The accumulated cost $Cost(l, q, R_q)$ and gain $Gain(l, q, R_q)$ for a query q and its result list R_q with limit l is

$$Cost(l, q, R_q) = cost_q(|q|) + \sum_{i=1}^{l} cost(r_i), \text{ where}$$

$$cost(r_i) = \begin{cases} cost_{sc} + cost_{cl}, & \text{if } rel(r_i) \geq \tau_{rel} \text{ and } r_i \text{ not similar to sth. in } Clicked, \\ cost_{sc}, & \text{otherwise.} \end{cases}$$

$$Gain(l, q, R_q) = \sum_{i=1}^{l} gain(r_i), \text{ where}$$

$$gain(r_i) = \begin{cases} rel(r_i), & \text{if } rel(r_i) \geq \tau_{rel} \text{ and } r_i \text{ not similar to sth. in } Clicked, \\ 0 & \text{otherwise.} \end{cases}$$

Determining the ideal search behavior forms a multiple-choice knapsack problem. For each result list R_q of each query q in the session S, we have to choose a limit l_q such that the total cumulated information gain is maximized and a given cost budget $cost_{max}$ is not exceeded.

$$\text{maximize} \sum^{q, R_q \in S} Gain(l, q, R_q) \qquad \text{while} \sum^{q, R_q \in S} Cost(l, q, R_q) \leq cost_{max}$$

Multiple-choice knapsack is NP-hard [27]. In order to prune the problem space, we omit *dominated* states that can never be part of an optimal solution: For result list R_q of query q, a limit l is dominated by a limit $l' \neq l$ iff either $Cost(l, q, R_q) > Cost(l', q, R_q)$ and $Gain(l, q, R_q) \leq Gain(l', q, R_q)$ or $Cost(l, q, R_q) \geq Cost(l', q, R_q)$ and $Gain(l, q, R_q) < Gain(l', q, R_q)$. For a sample result list with the relevant entries at ranks 1, 3, and 6, these ranks form the dominating limits. A limit at rank 2 is dominated by the limit at rank 1 since both lead to the same information gain but the limit at rank 2 has higher costs. Limits at ranks 4 or 5 are dominated by the limit at rank 3, etc. For determining the click behavior of the ideal user, each relevant result not similar to something clicked before represents a dominating limit.

In order to derive an optimal interaction sequence (i.e., ideal behavior), we have to choose from each result list in the session the limit that leads to an optimal gain for the whole session (i.e., the highest information gain possible for a given cost budget). There are several algorithmic solutions for such a multiple-choice knapsack problem like a dynamic programming approach [33] or a branch-and-bound strategy [20]. However, we cannot apply these approaches since we do not allow for clicking a relevant document if something similar has been clicked before. Hence, each click has a potential influence on the information gain of later results. If the user clicks on a relevant result in the current list, similar entries are no longer relevant in the next lists. In other words, we cannot treat the result lists independently but every combination of dominating states has to be checked for finding an optimal sequence. Let a path $P =< l_1, \ldots l_n >$ through a search session S be a list of limits for every result list. We call P a d-path, if only dominating limits are included. Let $\mathcal{P}$ be the family of all possible d-paths. In order to find a d-path that represents ideal user behavior, we derive the total cost $Cost(P, S)$ and gain $Gain(P, S)$ for every d-path $P \in \mathcal{P}$ as

$$Cost(P, S) = \sum_{\substack{l_q \in P, \\ q, R_q \in S}} Cost(l_q, q, R_q) \quad \text{and} \quad Gain(P, S) = \sum_{\substack{l_q \in P, \\ q, R_q \in S}} Gain(l_q, q, R_q).$$

From the d-path family we algorithmically choose a d-path that does not exceed the cost limit and that has the highest gain as follows. The dominating limits in every result list in the session are set to the ranks of the relevant results. All the combinations of all dominating limits of every result list then form the family $\mathcal{P}$ of possible d-paths. From this family, an ideal d-path P_{ideal} for a cost budget $cost_{max}$ is derived by first removing from $\mathcal{P}$ all d-paths that exceed the cost limit and then choosing one d-path with the highest gain. Note that clicks on similar results will not be part of such a path as long as the budget is not too high (since they do not yield any gain in our scenario) and that he resulting path is an optimal sequence of interactions given the cost budget—the ideal user behavior.

5 Spreading Activation Users

To simulate ideal click behavior, relevance judgments have to be "known" to the user. When no relevance information is available, we need another strategy for deterministic click decisions. We propose a cognitive approach employing the task description and shown snippets to this end.

5.1 Cognitive Modeling and Spreading Activation

Cognitive models explain basic cognitive processes (e.g., learning and decision making) and their interactions in more complex processes. Their big advantage over statistical models is that instead of inferring a posterior description from generated data, explanations for cognitive processes can be found in an inductive way [13]. One example of cognitive modeling is Pirolli and Card's information foraging theory [32] stating that users searching for information are faced with traces of navigational cues (e.g., links) emitting *information scent*

and that the cue with the most information scent will be followed. This rational behavior aims for an effective trade-off between cost and benefit and matches our general user model. However, we will not employ the costly comparison strategy of the original model but only use the cognitive SNIF-ACT architecture [21]; calculating information scent with the help of the spreading activation model.

Fu and Pirolli use the spreading activation model to estimate the utility of navigational choices [21]. The neuronal structure of the brain is modeled as an associative network consisting of interconnected concepts with different association strengths as in Anderson et al.'s cognitive architecture ACT-R [1]. When the user reads a document or a snippet, some of the concepts in the associative network are *activated*. This activation then spreads through the network and may activate other concepts depending on the associative strength. In our context, two regions in the associative network are important for the snippet relevance assessment: the region that is activated by reading the snippet (the perception), and the region that represents the user's focus and intention (i.e., the topic description in TREC scenarios). While scanning a snippet, the user model encounters the concepts in the snippet and these network nodes are activated and spread through the network to eventually activate topic description concepts. The relevance is then assessed according to the total activation level of the description concepts; if the activation is above a certain threshold, the document is perceived as relevant and a click is performed.

Concept Extraction The head-noun phrase extractor [9] is used to identify concepts in task descriptions and snippets. On average, document snippets contain fewer terms than a TREC task description (34 vs. 42) but both have similar number of concepts (8 vs. 10). We also removed some more "instructional" concepts like `find information` contained in many TREC descriptions.

Spreading Activation Calculation The concepts extracted from the topic descriptions and the document snippets form the nodes of a network. As for the edges (i.e., the activation strength), we simplify the relevance assessment situation to a bipartite directed graph. The concepts extracted from a scanned snippet form one node subset (the perception) and the concepts from the task description form the other (the focus). We assume that all snippet concepts are connected to all description concepts and omit any activations that may spread between concepts of one side. Based on this simplified network, we compute the total activation level A of the task concepts C_T that spread from the snippet concepts C_S. The activation level of a snippet is modeled as the sum of the attentional weighted association strength of every concept in C_T and every concept in C_S as $A(C_S, C_T) = \sum_{i \in C_T} \sum_{j \in C_S} association(i,j) \cdot attention(j)$ [21]. The formula includes a length normalization preventing unbounded activations and includes a temporal decay of activation following the assumption that a user spends more attention on the first concepts of a snippet. We follow Fu and Pirolli [21] using the exponential decay function $attention(j) = a \cdot e^{b \cdot j}$ and setting the scaling parameter $a = 1$ and the decay parameter $b = -0.1$. As for the association strength $association(i,j)$ between two concepts, we use the pointwise mutual information (PMI) [18] of $\log \frac{p(i,j)}{p(i)p(j)}$ approximating the probabilities $p(i,j)$, $p(i)$ and $p(j)$ with the normalized document frequencies df/N from the English Wikipedia, where N is the total number of Wikipedia articles. In a study comparing PMI to (generalized) latent semantic analysis as measures for association strength, Budiu et al. found that PMI is the most efficient method for identifying semantic similarities [12]. Following their suggestion, we use a window of 16 terms to derive the document frequencies $df(i,j)$.

Relevance Thresholding The total activation level A indicates how relevant a result appears to the user after the snippet scan. To distinguish between relevant results that should be clicked and non-relevant results that should not be clicked, an activation threshold τ_{act} is

part of the spreading activation model. We set the binary relevance of a snippet S and a task description T to

$$rel(C_S, C_T) = \begin{cases} 1 & \text{if } A(C_S, C_T) \geq \tau_{act}, \\ 0 & \text{otherwise,} \end{cases}$$

and propose two ways for setting the threshold τ_{act}: a static constant extracted from user interaction logs and a dynamic variant adapted to the rank bias favoring clicks on the first ranks.

Static Threshold To determine a static threshold, we use the TREC 2012 Session track logs. We compute the activation level of every result snippet and let the relevance judgments ≥ 2 form the relevant class. The mean activations of relevant and non-relevant results then are significantly different (22.8 vs. 12.8, $p \ll 0.01$ for a t-test). To choose a thresholding strategy, we compare the F-scores of a maximum a posteriori estimation (MAP) threshold, a likelihood comparison variant of MAP ignoring the prior probabilities, and an oracle threshold chosen to yield the best possible F-score. Due to the big difference of the prior probabilities (only 20% of the results are relevant), the conservative MAP estimation had a lot of false negatives (F-score of 0.19) such that we choose the likelihood estimation as our static threshold that comes pretty close to the artificial best possible F-score method in our pilots (F-score of 0.47 vs. 0.48).

Dynamic Threshold The underlying assumption of our dynamic threshold is a rank bias on the user side meaning that the users get more and more "skeptical" at lower ranks requiring a higher activation for a click. We model this assumption as follows. The user starts with a fixed activation threshold for the first rank that may very well represent rank bias by setting the initial $\tau_{act} = 0$ resulting in a blindfold click on the first rank. Every further result on a lower rank must have a higher activation level than the last clicked result; hence, the activation τ_{act} is monotonically growing. This dynamic thresholding is inspired by findings of Kean and O'Brien on users' rank bias [26] but in our cost-based model also resembles the fact that a mediocre result accessible at low costs may still be more appealing than a result with high relevance at a low rank. Hence, dynamic thresholding also models the *satisficing* behavior, meaning that the user prefers a fast and sufficient decision over evaluating all possible actions in order to find the optimum [29].

6 Our Analyzed User Models

Our general user model requires two components: (1) the *click behavior* of when to click on a result, and (2) the *stopping strategy* of when to switch to the next query and when to end the session.

6.1 Click Behavior

We propose three kinds of click behavior. First, the *optimal* click behavior of users who only click on relevant results—like the ideal user introduced in Section 4. Second, *activation-based* click behavior inspired by the spreading activation model introduced in Section 5—potentially leading to non-optimal clicks on non-relevant results. Third, a simple *click all* approach whose click decisions are independent of the relevance of a result: every result that is scanned is also clicked. This click behavior probably is the least cost efficient one, since clicking every result means also clicking every non-relevant result among the scanned results.

6.2 Stopping Strategies

We propose four simple stopping strategies following previous research. Zhang et al. observed that users tend to click more at the end of a session [42]. Their explanation is that

with every query reformulation the user improves the quality of the query and eventually ends up with a "best" query. The user probably scans some of the results in earlier queries but invests most of their budget for the last results. Our respective *prefer-last* stopping strategy is formally defined as follows. Let a path P consist of a list of limits $l_1 \dots l_n$ that represent the lowest rank the user views in each result list. A path P is a prefer-last path iff $l_i \leq l_{i+1}, i = 1, \dots, n-1$. In contrast to the findings of Zhang et al., the user model of the session-nDCG metric is based on the assumption that results of reformulated queries are less valuable since the user has to invest more effort [25]. According to this model, the user would prefer results of the first queries—yielding a *prefer-first* stopping strategy. A path P is a prefer-first path iff $l_i \geq l_{i+1}, i = 1, \dots, n-1$. To model the stopping strategy of the ideal user, we propose the highest-gain strategy. A user following this strategy views as many documents that appear to be relevant as possible for a given cost budget. A user model with optimal clicking behavior and highest gain strategy represents the ideal user. Let $\mathcal{P}$ be the family of all possible paths for a given cost limit and search session and let $gain(P)$ the accumulated information gain of a path P. A path P is a highest-gain path iff $gain(P) = max\{gain(P') : P' \in \mathcal{P}\}$. Similarly, to model more "average" users, we also propose a *median-gain* strategy where the user accumulates an information gain that represents the median of all information gains of all possible paths through a search session for a given cost limit. A path P is a median-gain path iff $gain(P) = median\{gain(P') : P' \in \mathcal{P}\}$.

6.3 Combining Clicking and Stopping

In order to simulate a certain click behavior and stopping strategy for a given search session, we identify paths through the search session that do not exceed the cost budget and that represent the stopping strategy. Finding such a path involves three steps. (1) For each result, determine whether it is clicked based on the click behavior. (2) Determine the family of all paths that do not exceed the cost budget. (3) From the path family, choose a path that matches the stopping strategy and has the highest information gain. From the 16 possible combinations, we further investigate all combinations with the highest-gain strategy (the ideal user with optimal clicks, the dynamic/static activation clicks, and the click-all user), the median user with optimal click behavior and median-gain strategy, and the prefer-first/-last users with clicking-all behavior and prefer-first/-last strategies. While the prefer-first/-last users represent assumptions from the literature, the median user somewhat represents an "average" user and the ideal user represents experts with perfect judgments from reading a snippet. The activation user models represent users without perfect click decisions and they can even be simulated in scenarios without relevance judgments. Although the click-all user seems very trivial, we include it in our considerations since it somewhat represents the envisioned user of a perfect retrieval systems. If the click-all user achieves the same information gain at the same costs as the ideal user, the ranking of the result list is perfect.

6.4 The TREC Session Track User

In the course of the TREC Session Track, logged interactions of real users were provided for several topics. We compare our simulated models to these users by modeling the *TREC user* whose behavior follows the originally logged data. In general, we expect the TREC user's performance to differ a lot from the ideal user in terms of information gain since a human user will not be able to optimally assess relevance from snippets, will have a rank bias, and will not make perfect stopping decisions. We instantiate the TREC user model for each search session in the TREC Session Track data as if they were produced following our general user model framework. This assumes top-down scanning, at least one snippet scan per result list, scans of all snippets of ranks above clicked results, etc.

Table 1. Average accumulated gain on the TREC Session Track 2011–2013 data.

	Ideal	Median	Act. St.	Click all	TREC	Act. Dyn.	Pref. First	Pref. Last
mean	2.7	1.9	1.5	1.5	1.4	1.2	1.2	1.0
med	2.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
std	2.5	1.4	1.5	1.9	1.8	1.3	1.2	1.2
max	18.0	9.0	9.0	16.0	15.0	5.0	4.0	6.0

7 Evaluation

We conduct experiments on data from the TREC Session Track 2011–2013 comparing our models with respect to information gain and cost usage, and analyzing the relation to traditional effectiveness metrics.

7.1 Accumulated Information Gain

The budget for a session is set to the time the original user's interactions would need in our general model setup. Following observations of Tran and Fuhr [39] we assume 2 seconds for a snippet scan and 15 seconds for a click, and following observations of Arif and Stuerzlinger [2] a query costs 1 second per term. The original TREC data consists of 288 search sessions for 160 topics. However, for 188 sessions none of the models (including the original TREC user) can achieve any information gain given the budget (i.e., no relevant results at all or too low in the lists). For our evaluation, we use the remaining 110 sessions and assume the gain per clicked result to correspond to the relevance score in the TREC Session Track judgments.

Table 1 shows the characteristics of the accumulated information gain distribution. The ideal user performs best, followed by the median user. The click-all user, the static activation user and the TREC user have about the same average performance. Interestingly, the ideal user almost doubles the performance of the original TREC user at the same cost. The prefer-first user is significantly better than the prefer-last user: the TREC search sessions seem to have more relevant documents in the first result lists.

To identify correlating user models, we compute the Spearman's rank correlation coefficient for each of the 136 possible pairs among the TREC user and the 16 different user models possible from our four click behaviors and four stopping strategies. The user models with the same click behavior correlate more than user models with the same stopping strategy and the choice of the click behavior has a higher impact on the user model's performance than the choice of the stopping strategy. The user models with the highest correlation to the TREC user are the model with dynamic activation clicks and prefer-first stopping strategy (Spearman's rank correlation test $\rho = 0.65$, $p < 0.01$) and the dynamic activation user model with highest gain strategy ($\rho = 0.62$, $p < 0.01$). This again reflects the rank bias of real users (dynamic thresholds) and supports the model underlying the session-nDCG metric (prefer-first).

7.2 Cost Usage

Figure 2 shows the distribution of the cost spent by the TREC user as a portion of the "maximum cost," the cost needed to click on all relevant documents in a session (including scanning all previous snippets). On average, the TREC user used 71% of the maximum cost; for half of the sessions the user invested 61% of the maximum cost reflecting the satisficing theory we briefly discussed in the thresholding part. However, in 19% of the sessions, the TREC user invests even more effort than necessary; mostly in sessions where few relevant results are found but more are clicked.

In order to compare how the user models use the cost budget, we also analyze the interactions for which some cost is spent. All user models spend the most cost on clicking but the ideal user and the median user invest approximately equal amounts for the different interactions; they scan way more results than they click. He and Wang [23] and Tran and Fuhr [40]

Table 2. Transition probabilities between query q, click c, snippet scan s, and end e.

	TREC	Ideal	Median	Act. St.	Act. Dyn.	Click all	Pref. First	Pref. Last
$q \to s$	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
$s \to q$	0.03	0.09	0.12	0.07	0.01	0.02	0.03	0.03
$s \to s$	0.56	0.52	0.51	0.30	0.35	0.03	0.00	0.02
$s \to c$	0.39	0.34	0.31	0.59	0.61	0.93	0.93	0.93
$c \to s$	0.55	0.47	0.34	0.57	0.45	0.58	0.50	0.47
$c \to q$	0.25	0.28	0.35	0.25	0.31	0.23	0.28	0.28
$s \to e$	0.02	0.05	0.07	0.05	0.02	0.02	0.04	0.02
$c \to e$	0.20	0.26	0.32	0.19	0.24	0.19	0.22	0.25

also suggest Markov models to investigate search behavior. A Markov model consists of a set of states and transition probabilities with the assumption that the probability of transitioning to the next state is only dependent on the current state. The transition probability between a state a and a state b is $p(a \to b)$ given by the relative occurrence frequency.

Table 2 shows the transition probabilities of our user models and the TREC user. The user models differ the most in the probability $p(s \to s')$ of transitioning from one snippet scan to the next snippet scan and the probability $p(s \to c)$ of transitioning from a snippet scan to a click. For the ideal user, the median user, and the TREC user it is more likely to continue with the next snippet scan, for the other user models it is more likely that they will click.

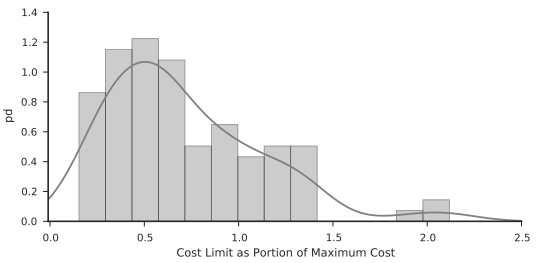

Figure 2. Cost limits of the logged users in the TREC data.

7.3 Simulated Users and Evaluation

We compare the average estimated information gain of our simulated users to the "traditional" metrics session-discounted cumulative gain (sDCG), expected reciprocal rank (ERR) and MAP on the sessions of the TREC Session Track.

The behavior of our simulated user models is cost-driven such that we can describe the accumulated information gain on a search session as a function $Gain(cost_{max})$ of the cost budget. In order to give an estimate on how much information gain a user model will accumulate, we need to take into account how the users choose their cost limit. Let $f(cost_{max})$ be a probability density function that represents the likelihood of choosing a cost limit. This cost limit likelihood function is normalized such that the integral between the minimum and the maximum of the function equals 1. Smucker and Clarke [36] proposed to use such a function f in order to estimate the accumulated information gain E of a session S as $E(S) = \int_0^\infty Gain(S, cost_{max}) \cdot f(cost_{max})\, dcost_{max}$. The probability density function we obtain is the curve in Figure 2 approximated using a kernel density estimation. The cost budget is normalized with the maximum cost (i.e., the cost needed to click all relevant results in a session S): $maxcost(S) = cost_{scan} \cdot |D| + cost_{click} \cdot |D_{rel}| * \sum_{i=1}^{|S|} cost_{query} \cdot |q_i|$, where $|D|$ is the number of results in the session and $|D_{rel}|$ is the number of relevant results.

In order to calculate the estimated information gain for a user model and a session, we sum the gain and the likelihood of the cost budgets between 0 and an upper bound. We set this upper bound to $2.5 \cdot maxcost(S)$ since this is the highest cost limit any real user has spend in any session (cf. Figure 2). As an increment $incr$ for the budgets we use the cost it takes to scan one snippet and perform one click. The estimated gain E

of a session S then can be calculated as $E(S) = \sum_{i=0} Gain(S, incr(i)) \cdot F(i)$, where $F(i) = \int_{incr(i-1)}^{incr(i)} f(cost_{max})\, dcost_{max}$ and $incr(i) = i \cdot (cost_{click} + cost_{scan})$. The rectangle method can be used to calculate an efficient approximation of the integral of the cost limit likelihood function F in one incrementation step i. We derive the estimated information gain of each of our seven simulated user models for the TREC Session Track 2011–2013 data and compute the correlation with the sum of the individual ERR values of the result lists, the mean of the summed average precisions of the result lists (MAP), and the session-DCG. Among the individual pairs, the highest correlation of 0.91 is between the average estimated information gain of our deterministic user simulations and session-DCG. The MAP metric correlates the least with the other metrics and our simulations (0.73). These correlations show that based on user simulations, the session-DCG metric is very reasonable. An interesting future metric could be formed by the difference of the ideal user to the more average median or activation users. If system A has better ideal user gains than system B but lower average/activation user gains, real users behaving more "average" and probably using snippet activation of some kind in their click decisions would prefer system B—which also is another argument for working on highly informative snippets giving a clue on actual result relevance.

8 Conclusion

We propose a framework to simulate deterministic user models with different stopping strategies and click behaviors. The goal is to use the simulations to better understand and evaluate user behavior in search sessions or query suggestion scenarios without requiring a huge online user population. We measure the effort of a simulated user by assigning costs to every interaction and contrast that with the achieved information gain. One of models is the ideal user with optimal click behavior and a high-information gain stopping strategy representing the perfect trade-off between cost and gain (i.e., the highest information gain possible for a given cost budget). More "average" variants are the median user with decisions towards achieving a median possible gain or the more cognitive activation-based users whose click behavior employs the spreading activation model during snippet scans. Comparing the deterministic simulations to real TREC users (interaction logs of the TREC Session Track), the "real" TREC user achieved only about half of the gain the ideal user would manage with the same cost budget. The TREC user correlates the most with an activation user having a dynamic click threshold. Using Markov model analysis, we show that the TREC users and our user models with optimal click behavior click less than other models. The estimated average gain of the simulated users correlates very well with the session-DCG metric. Though all proposed models are deterministic, our framework allows to include probabilistic decisions as well. An interesting application could be estimating the information gain with the help of large populations of simulated users in scenarios where no huge logs of millions of users are available (e.g., enterprise search). A metric based on simulation would be very transparent since for every instance of a user the achieved information gain is reproducible. The effect of changes in the ranking or the UI (that also influences cost) can be directly tested on different instances of simulated users. Different cost models also form a promising future direction since costs heavily influence search behavior [6]. Scanning a list of ten results is more costly on a phone than on a desktop while talking to a device could make queries cheaper. With variable costs, different environments can be simulated. Finally, a very important addition would be the extension of our framework such that also query (re-)formulations are simulated. Possible steps could be simulating known-item queries (clicked documents as the known item) or query simulation based on anchor texts [5,19]. This would allow the simulation of complete sessions based on a search task description without relying on the queries of the TREC Session tracks or similar datasets.

Acknowledgment Working on simulated ideal and average users was very much inspired by many discussions the first author had with Leif Azzopardi, Charlie Clarke, Gianmaria Silvello, and Robert Villa in the "User simulation" working group of the Dagstuhl seminar 13441, organized by Maristella Agosti, Norbert Fuhr, Elaine Toms, and Pertti Vakkari.

References

1. Anderson, Matessa, Lebiere. ACT-R: A theory of higher level cognition and its relation to visual attention. *Hum.-Comput. Interact.*, 12(4):439–462, 1997.
2. Arif, Arif, Stuerzlinger. Analysis of text entry performance metrics. *IEEE TIC-STH 2009*, 100–105.
3. Azzopardi. Modelling interaction with economic models of search. *SIGIR 2014*, 3–12.
4. Azzopardi. The economics in interactive information retrieval. *SIGIR 2011*, 15–24.
5. Azzopardi, de Rijke, Balog. Building simulated queries for known-item topics: an analysis using six european languages. *SIGIR 2007*, 455–462.
6. Azzopardi, Kelly, Brennan. How query cost affects search behavior. *SIGIR 2013*, 23–32.
7. Azzopardi, Zuccon. An analysis of theories of search and search behavior. *ICTIR 2015*, 81–90.
8. Azzopardi, Zuccon. Two scrolls or one click: A cost model for browsing search results. *ECIR 2016*, 696–702.
9. Barker, Cornacchia. Using noun phrase heads to extract document keyphrases. *AI 2000*, 40–52.
10. Baskaya, Keskustalo, Järvelin. Time drives interaction: simulating sessions in diverse searching environments. *SIGIR 2012*, 105–114.
11. Baskaya, Keskustalo, Järvelin. Modeling behavioral factors in interactive information retrieval. *CIKM 2013*, 2297–2302.
12. Budiu, Royer, Pirolli. Modeling information scent: A comparison of LSA, PMI and GLSA similarity measures on common tests and corpora. *RIAO 2007*, 314–332.
13. Busemeyer, Diederich. *Cognitive Modeling*. SAGE Publications, 2009.
14. Carterette, Bah, Zengin. Dynamic test collections for retrieval evaluation. *ICTIR 2015*, 91–100.
15. Carterette, Kanoulas, Yilmaz. Simulating simple user behavior for system effectiveness evaluation. *CIKM 2011*, 611–620.
16. Chapelle, Metlzer, Zhang, Grinspan. Expected reciprocal rank for graded relevance. *CIKM 2009*, 621–630.
17. Chapelle, Zhang. A dynamic Bayesian network click model for web search ranking. *WWW 2009*, 1–10.
18. Church, Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, 1990.
19. Dang, Croft. Query reformulation using anchor text. *WSDM 2010*, 41–50.
20. Dyer, Kayal, Walker. A branch and bound algorithm for solving the multiple-choice knapsack problem. *J. Comp. Appl. Math.*, 11(2):231–249, 1984.
21. Fu, Pirolli. SNIF-ACT: A cognitive model of user navigation on the world wide web. *Hum.-Comput. Interact.*, 22(4):355–412, 2007.
22. Hagen, Gomoll, Beyer, Stein. From search session detection to search mission detection. *OAIR 2013*, 85–92.
23. He, Wang. Inferring search behaviors using partially observable markov model with duration (POMD). *WSDM 2011*, 415–424.
24. Järvelin, Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM TOIS*, 20(4):422–446, 2002.
25. Järvelin, Price, Delcambre, Nielsen. Discounted cumulated gain based evaluation of multiple-query IR sessions. *ECIR 2008*, 4–15.
26. Keane, O'Brien, Smyth. Are people biased in their use of search engines? *CACM*, 51(2):49–52, 2008.
27. Kellerer, Pferschy, Pisinger. *Knapsack problems*. Springer, 2004.
28. Klöckner, Wirschum, Jameson. Depth- and breadth-first processing of search result lists. *CHI 2004*, p. 1539.
29. Manktelow. *Reasoning and thinking*. Psychology Press, 1999.
30. Maxwell, Azzopardi, Järvelin, Keskustalo. Searching and stopping: An analysis of stopping rules and strategies. *CIKM 2015*, 313–322.
31. O'Brien, Keane. Modeling user behavior using a search-engine. *IUI 2007*, 357–360.
32. Pirolli, Card. Information foraging in information access environments. *CHI 1995*, 51–58.
33. Pisinger. A minimal algorithm for the multiple-choice knapsack problem. *Europ. J. Op. Res*, 83(2):394–410, 1995.
34. Sanderson, Paramita, Clough, Kanoulas. Do user preferences and evaluation measures line up? *SIGIR 2010*, 555–562.
35. Smucker, Clarke. Modeling Optimal Switching Behavior. *CHIIR 2016*, 317–320.
36. Smucker, Clarke. Modeling user variance in time-biased gain. *HCIR 2012*, paper 3.
37. Smucker, Jethani. Human performance and retrieval precision revisited. *SIGIR 2010*, 595–602.
38. Thomas, Moffat, Bailey, Scholer. Modeling decision points in user search behavior. *IIiX 2014*, 239–242.
39. Tran, Fuhr. Using eye-tracking with dynamic areas of interest for analyzing interactive information retrieval. *SIGIR 2012*, 1165–1166.
40. Tran, Fuhr. Markov modeling for user interaction in retrieval. *MUBE 2013*, 13–14.
41. Turpin, Scholer. User performance versus precision measures for simple search tasks. *SIGIR 2006*, 11–18.
42. Zhang, Chen, Wang, Yang. User-click modeling for understanding and predicting search-behavior. *KDD 2011*, 1388–1396.