User simulations for interactive search: evaluating personalized query suggestion

Suzan Verberne[1], Maya Sappelli[1,2], Kalervo Järvelin[3], and Wessel Kraaij[2,1]

1. Institute for Computing and Information Sciences, Radboud University Nijmegen
2. TNO, Delft
3. School of Information Sciences, University of Tampere

Abstract. In this paper, we address the question "what is the influence of user search behaviour on the effectiveness of personalized query suggestion?". We implemented a method for query suggestion that generates candidate follow-up queries from the documents clicked by the user. This is a potentially effective method for query suggestion, but it heavily depends on user behaviour. We set up a series of experiments in which we simulate a large range of user session behaviour to investigate its influence. We found that query suggestion is not profitable for all user types. We identified a number of significant effects of user behaviour on session effectiveness. In general, it appears that there is extensive interplay between the examination behaviour, the term selection behaviour, the clicking behaviour and the query modification strategy. The results suggest that query suggestion strategies need to be adapted to specific user behaviours.

Keywords: interactive search, academic search, user simulations, user interaction, query suggestion

1 Introduction

Effective search for information often needs more than one iteration: An initial query is modified multiple times to increase precision or recall. Query suggestion is a functionality of a search engine that suggests the user a list of queries to proceed the search session with. If the query suggestion algorithm works well, it reduces the cognitive load of users and makes them more efficient in their search for information [2]. For web search, query logs are a good source for query suggestion [11]. However, for search tasks addressing highly specialized topics, where there are no relevant queries from other users available, the only realistic option we have is to fall back to the user's own data (previous queries, clicked documents) [18]. In this paper, we evaluate query suggestion for interactive search tasks in a scientific domain. After the initial (user-formulated) query, query suggestion can assist the user in entering effective follow-up queries. Our system generates candidate follow-up queries from the documents clicked by the user and presents these candidate queries (extensions or adaptations of the previous query) in a ranked list. This is a potentially effective method for personalized query suggestion, as shown in previous work [20], but we hypothesize that its effectiveness heavily depends on user behaviour. We therefore address the following research question in this paper: what is the influence of user search behaviour on the effectiveness of personalized query suggestion?

We set up a series of simulation experiments in which we explore a range of possible user behaviours in order to find out what the important variables are. We are especially interested in what happens to the effectiveness of personalized query suggestion when user behaviour is not perfect. For example, a non-perfect (realistic) user formulates underspecified queries, clicks on irrelevant documents, selects suboptimal queries from the query suggester, and ends his search before he has reached full recall of relevant results. We investigate the following aspects of user behaviour: (1) Query modification strategies, as proposed by [3]; (2) Examination and click behaviour, using an adapted version of the Click Chain Model by [9]; (3) Query selection strategies (model proposed in this paper); and (4) Time-driven session stopping behaviour [3].

Simulation of user behaviour is a powerful tool to evaluate systems for a large range of user behaviours without bringing in hundreds of real users. We use simulations as *what-if* experiments: we observe how the effectiveness of our system changes with varying user behaviours [1]. It should be noted that even if a model cannot be fully validated with user data (because of the lack of sufficient suitable data), the model can still be very useful to see the relation between user behaviour and system effectiveness.

The contributions of this paper are: (a) Session simulations of interactive search, based on the combination of four user models: a click model, a model for time-based stopping behaviour, a model for query formulation strategies and a new model for query selection strategies; (b) An adaptation of the Click Chain Model that accounts for lower examination probabilities for lower ranked results; (c) Large-scale simulations to measure the effectiveness of query suggestion under influence of diverse user behaviours.

2 Related work

Query suggestion. The most used source for query suggestion are query logs. These are especially useful when the queries of other users can be reused by the current user, for example because the queries occur in similar sessions [11]. For personalization purposes, the user's own previous queries are sometimes used as a source for query suggestion, but this data is sparse and topic-dependent [7]. When there are no relevant query logs available, documents in the retrieval collection can be used as an alterative source for query suggestion. The idea is to extract query terms from the documents in the collection that are most relevant to the user's current query. Relevance can either be defined by the search engine itself, using the top-n highest ranked documents ('pseudo-relevance feedback'), or by the user's clicks, using the documents that are clicked by the user ('relevance feedback') [5]. One advantage of using clicked documents as source for query suggestion, is that the suggested queries are geared towards the user's current information need, since he will click more often on documents that seem relevant to him [18]. This aspect makes the use of clicked documents suitable for search tasks addressing highly specialized topics. For personalized query suggestion in a scientific topic domain, we therefore implemented the recent successful approach by [20], which extracts terms from documents clicked in the current session and uses these terms as suggestions for follow-up queries.

User simulations. Most previous work on user simulations for information retrieval focuses on models for result examination (snippet scanning) and clicking behaviour. In the current paper, we use the Dependent click model by [10] and the Click chain model by [9] for simulating examination and clicking behaviour. Examination and click models describe the user behaviour for one query; less attention has been paid to simulation of *session behaviour*. For simulating complete sessions, query modification strategies [4] need to be defined, as well as session stopping behaviour. For both, we use the models proposed by [3]. For the evaluation of query suggestion methods, we also need a model for query selection behaviour. Previous works on query suggestion either assume that the user always selects the first-ranked query (fully trusting the query suggester) [16] or uses expert assessments to determine which queries are selected [20, 6] . The main drawback of the latter approach is that each newly implemented query suggestion method will generate new terms that need to be judged. Therefore, we propose a model for query selection behaviour in this paper that allows query selection to be part of user simulations.

3 Methodology

3.1 Data

The iSearch collection of academic information seeking behaviour [17] consists of 65 natural search tasks (topics) from 23 researchers and students from university physics departments. The topic owners were given a task description form with five fields: (a) What are you looking for? (information need); (b) Why are you looking for this? (work task context); (c) What is your background knowledge of this topic? (knowledge state) (d) What should an ideal answer contain to solve your problem or task? (ideal answer); (e) Which central search terms would you use to express your situation and information need? (search terms). A collection of 18K book records, 144K full text articles and 291K metadata records from the physics field is distributed together with the topics. For each topic, 200 documents were manually assessed on their relevance using a 4-point scale.

3.2 Retrieval set-up

We indexed the iSearch collection with the Indri search engine[1]. We used the Indri API to set up a query interface to the combined index of Metadata, Book and Article records. All characters that are not alphanumeric, no hyphen or whitespace are removed from the query terms. Multiple query terms are concatenated and combined using the `combine` function in the Indri query language. For example, the two terms 'ZNO' and 'Transparent conductive oxides' together form the Indri query `#combine(zno transparent conductive oxides)`. Thus, we convert all queries to bag-of-words representations. As ranking model, we use the Indri LM with default Dirichlet smoothing ($\mu = 50$). Per query, we retrieve 100 results from the combined index.

[1] http://www.lemurproject.org/indri/

3.3 Simulation of query modification strategies

We implemented query modification strategies S1–S5 from [3], based on physicians' information seeking behaviour [15]: S1 creates queries of one term[2] where each follow-up query is a different term; S2 creates queries of two terms of which the first term is kept and the last term is varied; S3 creates queries of three terms of which the first two are kept and the last one is varied; S4 creates incrementally growing queries starting with one term and adding one term to each follow-up query; S5 creates incrementally growing queries starting with two terms. For a given topic, the first query of the session is always the first term (or first and second term) from field e 'search terms' in the iSearch data. When adding more terms from the iSearch data, we maintain the original order as created by the topic owner. For example, consider the search terms field "ZnO, transparent conductive oxides, magnetron sputtering, doping". With query modification strategy S4 , the initial query is 'zno'; the second query (without query suggestion) is 'zno transparent conductive oxides'; the third query 'zno transparent conductive oxides magnetron sputtering' and the fourth query 'zno transparent conductive oxides magnetron sputtering doping'.

A search session is defined by a pre-defined time limit; all actions in the session (query formulation, result examination) are associated with *costs* in terms of number of seconds. The user continues formulating queries as long as he has time left in the session and search terms left in his task description in the iSearch data. For query modification, we adopt the costs from [3]: Formulating the initial query costs 3 seconds in S1 and S4, 6 seconds in S2 and S5, and 9 seconds in S3. Each subsequent query costs 3 seconds.

3.4 Simulation of examination behaviour and clicks

We use the Click Chain Model (CCM) by [9] to simulate examination and clicking behaviour on the result list. Like all cascade models, CCM assumes that the user examines the result list from top to bottom.

Click probabilities. The conditional probability that a document is clicked, given that its snipped is scanned/examined ($P(C_i = 1|E_i = 1)$, where E_i means: "the snipped of the ith result is examined"), is determined by R_i, the perceived relevance of the examined document. For estimating R_i, we use a model that gives the probability that a document is *perceived* relevant given the *actual* relevance of the document (which is given by the relevance assessments in the iSearch data). This probabilistic model is adopted from the dependent click model by [10], who defined probabilities for three different user/query types: perfect (the user never clicks an irrelevant document), informational and navigational (see Table 1). Furthermore, in order to make evaluation straightforward, we assume in the simulation that the user remembers his/her clicks for the short duration of a session and therefore never clicks on a document he has clicked on before in the same session.

Examination probabilities. In CCM, the probability of examining the next result ($E_{i+1} = 1$) is zero if the current result is not examined (cascade assumption: if the user does not scan the ith snippet, he will also not scan the $i+1$th snippet). If the current result is examined ($E_i = 1$) and not clicked ($C_i = 0$), the probability of

[2] A term can consist of more than one word.

Table 1. The click probabilities that we use for user simulation. The model has been adapted from [10], converting a 3-level relevance scale to a 4-level relevance scale.

relevance grade	0	1	2	3
perfect	0.00	0.33	0.67	1.00
informational	0.40	0.60	0.75	0.90
navigational	0.05	0.33	0.67	0.95

examining the next result is a constant α_1 . The higher α_1, the more persevering the examination behaviour. We make one adaptation to the model: We argue that the examination probability should not only depend on user perseverance but also on the rank of the current result i: the further down in the result list a result is (the higher i), the lower the probability that the user examines the next result [14, 8]. Even a highly persevering user will at some point stop examination of a (long) result list. Therefore, we adapt the examination probability as follows, using a sigmoid function to model the decreasing examination probability with higher ranks:

$$P(E_{i+1} = 1 | E_i = 1, C_i = 0) = \frac{1}{1 + e^{k(i-\gamma)}} \qquad (1)$$

where i is the rank of the current result, k is a parameter representing the steepness of the slope (a higher k makes the sigmoid less linear and more threshold-like) and γ is a parameter that defines the center of the sigmoid; the rank at which $P(E_{i+1} = 1) = 0.5$. We use a sigmoid function because the examination probabilities that are reported in the literature (based on eye-tracking fixations) can be fitted using a sigmoid: With the parameters $k = 0.5$ and $\gamma = 5$, we can fit equation (1) to the distribution of fixations reported by [8] with a Mean Squared Error of 1.7% and the distribution of fixations reported by [14] with a Mean Squared Error of 3.0%. Both distributions are for web search.

In the situation where the current result was clicked ($C_i = 1$), the probability that the next result is clicked ($E_{i+1} = 1$) depends on the perceived relevance of the current result R_i, and two parameters α_2 and α_3 . In order to make the examination probability for $C_i = 1$ also rank-dependent, we use the same sigmoid function as eq. (1), but we set k to: $k = \alpha_2 * (1 - R_i) + \alpha_3 * R_i$, using α_2 and α_3 as in the original CCM. A larger difference between α_2 and α_3 leads to a larger influence of R_i.

Like query formulation, the examination of the result list is associated with costs. We also adopt these from [3]: the scanning of a snippet costs 3 seconds.

3.5 Query suggestion method

We implemented the method for personalized query suggestion from [20]. The simulated user gets 10 suggestions for query terms to be added to the next query. These terms have automatically been extracted from all the documents that the user clicked on in the current search session, including the clicked documents from earlier queries.[3] All word n-grams with $n = \{1, 2, 3\}$ in these documents are consid-

[3] In the case of metadata and book records, the terms are extracted from the fields 'title' and 'description'; in the case of articles in PDF, for which no metadata is available, the terms are extracted from the first 200 words of the document.

ered candidate terms. The terms are ranked by scoring them with Kullback-Leibler divergence [19] between the probability distributions for a term in two collections: the collection of documents clicked by the user and a background collection of general English (the Corpus of Contemporary American English, COCA). The output score denotes the informativeness of the term for the collection of clicked documents. The terms are ranked by this score and presented to the (simulated) user as suggested query terms, either to expand the previous query or replace the final term of the previous query, depending on the query modification strategy.

3.6 Simulation of query selection behaviour

We propose the following model for simulating the selection of a query in a query suggestion scenario. The input for the model is the output of the query suggestion method: an ordered list of suggested query terms $L = t_1, t_2, ...t_k$. We simulate the user's decision with four variables S_{ts}, S_{rel}, S_{in}, S_{st}. Each term in L takes a value for each of these four variables:

- S_{ts}: The term suggester score. This is the output of the query suggestion method (See Section 3.5).
- S_{rel}: The output of a term scorer that determines the informativeness of the term for the subcollection of documents that are relevant for the current topic, using Kullback-Leibler divergence between this subcollection and a background corpus of general English (COCA). If a term from L does not occur in the subcollection, $S_{rel} = 0$
- S_{in}: The output of a term scorer that determines the informativeness of the term for the user's explicit information need (a concatenation of the fields a, b and d for the current topic in the iSearch data), using Kullback-Leibler divergence between the collection and a background corpus of general English (COCA). If a term from L does not occur in the information need, $S_{in} = 0$.
- S_{st}: The output of a binary scorer that determines whether or not the term is in the set of the user-formulated search terms (from the iSearch data). If the term (normalized for case, whitespace and hyphenation) is in the list of search terms then $S_{st} = 1$, otherwise $S_{st} = 0$.

These four variables are justified as follows: S_{in} and S_{st} were given in the iSearch data by the searchers whose query selection behaviour we aim to simulate. S_{ts} is the score given by the term suggestion algorithm, the evaluation of which is central to our simulations. And S_{rel} is higher for terms that come from documents that are judged as relevant by the searcher; a competent searcher will be more likely to select one of these terms than a term from an irrelevant document. The term selection simulator is a tuple of integer weights $W = (W_{ts}, W_{rel}, W_{in}, W_{st})$. The simulated user selects a term by solving:

$$\arg\max_{t \in L} \frac{W_{ts} * S_{ts} + W_{rel} * S_{rel} + W_{in} * S_{in} + W_{st} * S_{st}}{\sum_{x \in \{ts, rel, in, st\}} W_x} \tag{2}$$

The higher the weight for S_{in}, S_{rel} and S_{st}, the more informed the simulated user is. A higher weight for S_{ts} implies more trust in the query suggester; a simulated user with a 0-weight for S_{in}, S_{rel} and S_{st} fully trusts the query suggester and will always take the top-ranked term. A simulated user with a 0-weight for S_{ts} and S_{rel} is very critical and will only select terms that are in his explicit information need or list of search terms.

It is yet unknown what a realistic time cost is for selecting a query from a drop-down list. We assume that it takes less time to select a query than to fomulate one: we set the time-cost of query term selection to 1 second.

3.7 Evaluation

We evaluate the effectiveness of the session using Cumulated Gain (CG) [13], following the arguments by [3] for no discounting and no normalization: discounting has value in a one-query evaluation setting but is not sensible over a complete session, and normalization may leed to counterintuitive results when user behaviour is based on time costs instead of result ranks. CG is the sum of the relevance scores of all seen documents in the session. Thus, the goal for the simulated user was to collect as much gain as possible in a 5 minute session. For each query in the session, we evaluate the result list up to the last examined result, keeping track of the relevance of the seen documents. Documents that have been seen by the user previously in the same session are disregarded. In all experiments, we set the session time limit (a bit arbitrarily) to 300 seconds (5 minutes). We leave it for later papers to examine the effect of session length.

4 Experiments and results

4.1 The effect of examination behaviour

We measured the effect of the examination parameters (k, γ, $\alpha_{2,3}$) in two settings: the setting where the user selects his own queries using terms from field e in the iSearch data, and the setting where the user gets query suggestions. In both settings, the query modification strategy was S4 (see Section 3.3).

In [9], the following values for the examination parameters are suggested for informational queries: $\alpha_1 = 1$ (the user inspects all results; queries without clicks were disregarded), $\alpha_2 = 0.40$ and $\alpha_2/\alpha_3 = 1.5$. Since we aim to investigate the influence of user aspects on the effectiveness of personalized query suggestion, we inspect a *range* of parameter values instead of fixing them for a given user type. We experimented with the following grid of parameter values around the values suggested in [9]: α_2 in the range 0.1–1.0 with steps of 0.1 and the proportion α_2/α_3 in the range 1.0–3.0 with steps of 0.5. We fix $k = \alpha_2$. For γ (the center of the sigmoid), we had found that $\gamma = 5$ gave the best fit when fitting the sigmoid to examination probabilities in web search (see section 3.4). For interactive search on scientific topics, we extend the range of this parameter to higher values, to represent a more recall-oriented user [16]: we use a grid in the range of 5–50 with steps of 5.

For the setting without query suggestions, we obtained CG values (averaged over queries) ranging from 0 to 5.1. The parameter that has the largest influence on the effectiveness of the session is γ: there is a strong positive relationship between γ and CG (Kendall's $\tau = 0.78$, $P < 0.0001$), while the relationship between α_2 (or k) and CG is weak (Kendall's $\tau = 0.20$, $P < 0.001$). The effect of γ on the effectiveness of

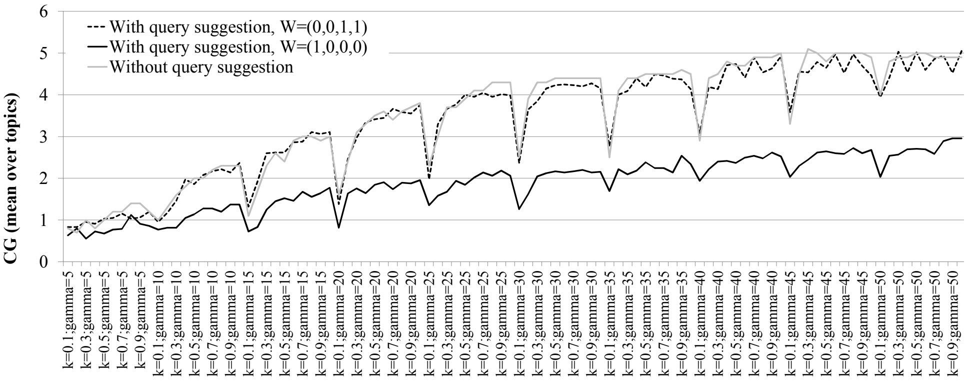

Fig. 1. Mean CG ($N = 65$ topics) for the grid of inspection parameter values (k and γ), with and without query suggestion. $W = (1, 0, 0, 0)$ represents 'lazy' query suggestion behaviour; $W = (0, 0, 1, 1)$ represents critical query selection behaviour.

the session is a consequence of the number of documents examined per query: in the sessions with $\gamma = 5$, the average number of results examined per query is 3.7, while in the sessions with $\gamma = 50$, the average number of results examined per query is 33.8. In other words, a higher γ represents more persevering user behaviour.

In the setting *with* query suggestions, we used the perfect click model (see Table 1) and we evaluated two extreme configurations of the term selection model W (See Section 3.6):

- $W = (1, 0, 0, 0)$: the user fully relies on the term suggester.
- $W = (0, 0, 1, 1)$: the user only selects terms that are in his explicit information need or his list of search terms. If none of the terms is, the user formulates his own query using the terms in the field e from the iSearch data (like in the setting without query suggestion).

Figure 1 shows the results. We see that a user who fully trusts the query suggester ($W = (1, 0, 0, 0)$) ends up with lower CG than the user who formulates his own query. The difference between the two is the smallest for the lowest values of γ and k. This suggests that the more persevering the user is in examining results (higher k and γ), the larger the importance of formulating or selecting the right query. On average, the lazy user who fully trusts the query suggester is faster: he can enter more follow-up queries because he spends less time formulating each query. For $\gamma = 10$, the user who formulates his own queries enters 5.1 queries per 300 second-session, while the user who picks the first suggestion from the query suggester enters 11.8 queries per session on average. In addition, the user who does not get query suggestions often runs out of query terms before the session time is up, while the user who picks the first suggestion from the query suggester mostly spends the complete 300 second-session formulating queries and stops when the 300 seconds are up.

We also see that the line for the setting without query suggestion and the line for the setting with query suggestion but critical query selection behaviour ($W = (0, 0, 1, 1)$) are strongly related to each other. This is because the user with critical query selection behaviour only selects a query term from the suggester if it is in his own list of query terms, or his explicit information need. Analysis of the query selection behaviour shows that for $k = 1$ and $\gamma = 50$, the user with $W = (0, 0, 1, 1)$ selects a suggested query for only 11.6% of his queries; in all other cases, he formulates his own query using a term from the iSearch data. For lower values of k and γ this percentage is even lower. In the next subsection, we more precisely investigate the effect of the term selection model W.

4.2 The effect of query selection behaviour ($\boldsymbol{W}$)

For measuring the effect of the term selection model W, we experiment with a grid of weight integer values $\{0, 1, 10\}$ for all four weights. Thus, we get configurations such as $(0, 10, 0, 1)$, $(1, 0, 10, 1)$, etc. We compared two values of the most important examination parameter: $\gamma = \{10, 50\}$. We fix k to 0.5, $\alpha_2 = k$ and $\alpha_2/\alpha_3 = 1.5$ because their effect on the effectiveness of the session is much smaller than of γ and having too many variables makes analysis of the results complex. In all cases, the query modification strategy was S4 and we used the perfect click model.

We found that the only W-parameter that has a significant relationship with CG is W_{ts}; this relationship is moderately negative (Kendall's $\tau = -0.37$, $P < 0.0001$ and $\tau = -0.34$, $P < 0.001$ respectively for $\gamma = 10$ and $\gamma = 50$). This means that the higher the weight for the term suggester score (thus the more the user trusts the term suggester), the lower the CG. Overall, the best-scoring parameter settings for query selection behaviour are $W = \{0, 0, 1, 0\}$ (CG for $\gamma = 50$ is 5.2, CG for $\gamma = 10$ is 2.3) and $W = \{0, 0, 1, 1\}$.

4.3 The effect of clicking behaviour

For investigating the effect of the click behaviour on the effectiveness of query suggestion, we evaluated the perfect, informational and navigational click models (see Section 3.4). We again compared two values of the most important examination parameter: $\gamma = \{10, 50\}$ and we fix k to 0.5, $\alpha_2 = k$ and $\alpha_2/\alpha_3 = 1.5$. For the query selection behaviour, we compared two parameter settings: $W = \{(1, 0, 0, 0), (0, 0, 1, 1)\}$. In all cases, the query modification strategy was S4. The results are in Figure 2.

The results show that a bigger effect comes from the examination parameter γ than from the click model and the query selection model W. Besides that, we see that with critical query selection behaviour ($W = (0, 0, 1, 1)$), the perfect click model seems to give the highest results for the persevering user ($\gamma = 50$), while the navigational model gives the highest results for the 'lazy' examination behaviour ($\gamma = 10$).[4] This suggests that for lazy examination behaviour, it can be profitable to click on more documents, even if not all of them are relevant. This is not the case when the user always selects the highest ranked suggested query ($W = (1, 0, 0, 0)$); then the three models perform almost equally.

4.4 The effect of the query modification strategy

We evaluated the five query modification strategies (see Section 3.3). We again compared two values of the most important examination parameter: $\gamma = \{10, 50\}$

[4] A paired t-test (with the CG scores for individual topics paired) shows that the difference between the navigational and the perfect click model for $\gamma = 10$ is significant with $P < 0.01$. For $\gamma = 50$, the difference is not significant.

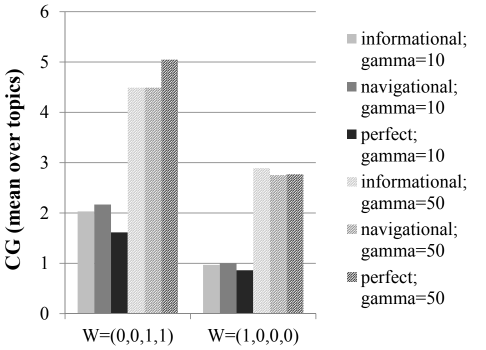

Fig. 2. Mean CG ($N = 65$ topics) for the 3 click models, 2 different examination behaviour types γ, and two different query selection behaviours W. In all cases, the query modification strategy is S4.

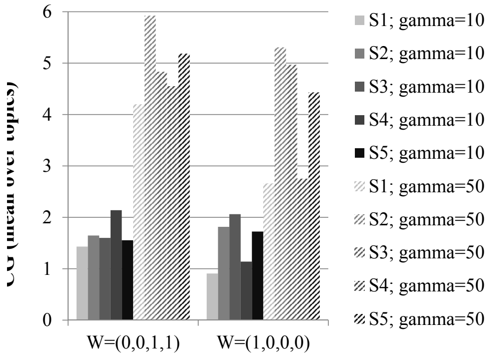

Fig. 3. Mean CG ($N = 65$ topics) for the five query strategies, two different examination behaviour types γ, and two different query selection behaviours W. In all cases, we used the perfect click model.

and we fix k to 0.5, $\alpha_2 = k$ and $\alpha_2/\alpha_3 = 1.5$. We used the perfect click model and for the query selection behaviour, we compared two parameter settings: $W = \{(1,0,0,0),(0,0,1,1)\}$. The results are in Figure 3.

Again, the biggest effect comes from the examination parameter γ. However, query strategy does have a big influence. The effect of query strategy is bigger for the user who trusts the query suggester ($W = (1,0,0,0)$) than for the user with critical query selection behaviour ($W = (0,0,1,1)$). Surprisingly, the best performing query strategy is S2 (subsequent queries of two terms of which the first term is kept and the last term is varied). The poorest performing query strategy is S1 (issueing one term at the time), followed by S4 (adding each new query term to the previous query) in most combinations of W and γ. An exception is the setting where $W = (0,0,1,1)$ and $\gamma = 10$ (lazy examination behaviour with critical query selection behaviour); in that case S4 is the most effective strategy. The differences between S1,4 and S2,3,5 are bigger for $W = (1,0,0,0)$ than for $W = (0,0,1,1)$. We think this is because picking the highest-ranked term from the query suggester for each follow-up query can lead to topic drift in the session. In S2, S3 and S5, the first query of the session consists of two user-formulated terms, whereas in S1 and S4, the first query only consists of one user-formulated term. The combination of two user-formulated search terms in the first query apparently ensures a better topical focus of the session.

5 Conclusions

We addressed the following research question in this paper: what is the influence of user behaviour on the effectiveness of personalized query suggestion? Query suggestion can make the user more efficient, because it takes less time to select a query than to formulate one. But query suggestion is not profitable for all user types. We found the following significant effects of user behaviour on the effectiveness of query suggestion: (1) The more persevering the user is in examining result lists, the larger the importance of selecting the right query from the query suggester. This might be because lower in the result list the documents are less relevant and therefore the suggested terms are of lower quality. The persevering user should therefore be more critical in where he clicks or more critical in selecting suggested queries; (2) The less critical the user is in selecting terms from the query suggester (the more trust he has in the suggestions), the lower the cumulated gain of the session; (3) If the user examines few results ('lazy examination behaviour'), clicking on more documents results in higher cumulated gain, even if not all of the clicked documents are relevant. This is in line with previous findings for pseudo-relevance feedback [12]. (4) Because of the risk of topic drift when the user adds suggested terms from clicked documents, it is profitable to start the session with a query consisting of more than one term.

It appears that there is extensive interplay between the examination behaviour and the term selection behaviour on the one hand, and the clicking behaviour or the query modification strategy on the other hand: both the choice of the most effective query modification strategy and the most effective click model depend on how persevering the examination behaviour and how critical the query selection is. This suggests that query suggestion strategies need to be adapted to specific user behaviours.

In future work, we plan to collect real user data for search tasks in a scientific domain. With these data, we will (1) model the query modification strategies that are typical for academic search sessions, including the associated time costs; (2) validate (and improve) our examination model with rank-dependent examination probabilities; and (3) optimize our query suggestion method and compare it to other methods.

References

1. Azzopardi, L., Järvelin, K., Kamps, J., Smucker, M.D.: Report on the sigir 2010 workshop on the simulation of interaction. SIGIR Forum **44**(2) (January 2011) 35–47
2. Azzopardi, L., Kelly, D., Brennan, K.: How query cost affects search behavior. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, ACM (2013) 23–32
3. Baskaya, F., Keskustalo, H., Järvelin, K.: Time drives interaction: simulating sessions in diverse searching environments. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, ACM (2012) 105–114
4. Bates, M.J.: Information search tactics. Journal of the American Society for information Science **30**(4) (1979) 205–214
5. Belkin, N.J., Cool, C., Kelly, D., Lin, S.J., Park, S., Perez-Carballo, J., Sikora, C.: Iterative exploration, design and evaluation of support for query reformulation in interactive information retrieval. Information Processing & Management **37**(3) (2001) 403–434
6. Bhatia, S., Majumdar, D., Mitra, P.: Query suggestions in the absence of query logs. In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, ACM (2011) 795–804

7. Feild, H., Allan, J.: Task-aware query recommendation. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. SIGIR '13, New York, NY, USA, ACM (2013) 83–92
8. Guan, Z., Cutrell, E.: An eye tracking study of the effect of target rank on web search. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM (2007) 417–420
9. Guo, F., Liu, C., Kannan, A., Minka, T., Taylor, M., Wang, Y.M., Faloutsos, C.: Click chain model in web search. In: Proceedings of the 18th international conference on World wide web, ACM (2009) 11–20
10. Hofmann, K., Schuth, A., Whiteson, S., de Rijke, M.: Reusing historical interaction data for faster online learning to rank for ir. In: Proceedings of the sixth ACM international conference on Web search and data mining. WSDM '13, New York, NY, USA, ACM (2013) 183–192
11. Huang, C.K., Chien, L.F., Oyang, Y.J.: Relevant term suggestion in interactive web search based on contextual information in query session logs. Journal of the American Society for Information Science and Technology **54**(7) (2003) 638–649
12. Järvelin, K.: Interactive relevance feedback with graded relevance and sentence extraction: simulated user experiments. In: Proceedings of the 18th ACM conference on Information and knowledge management, ACM (2009) 2053–2056
13. Järvelin, K., Kekäläinen, J.: IR evaluation methods for retrieving highly relevant documents. In: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, ACM (2000) 41–48
14. Joachims, T., Granka, L., Pan, B., Hembrooke, H., Gay, G.: Accurately interpreting clickthrough data as implicit feedback. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM (2005) 154–161
15. Keskustalo, H., Järvelin, K., Pirkola, A., Sharma, T., Lykke, M.: Test collection-based IR evaluation needs extension toward sessions–a case of extremely short queries. In: Information Retrieval Technology. Springer (2009) 63–74
16. Kim, Y., Seo, J., Croft, W.B.: Automatic boolean query suggestion for professional search. In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. SIGIR '11, New York, NY, USA, ACM (2011) 825–834
17. Lykke, M., Larsen, B., Lund, H., Ingwersen, P.: Developing a test collection for the evaluation of integrated search. In Gurrin, C., He, Y., Kazai, G., Kruschwitz, U., Little, S., Roelleke, T., Rüger, S., van Rijsbergen, K., eds.: Advances in Information Retrieval. Volume 5993 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2010) 627–630
18. Shen, X., Tan, B., Zhai, C.: Implicit user modeling for personalized search. In: Proceedings of the 14th ACM international conference on Information and knowledge management, ACM (2005) 824–831
19. Tomokiyo, T., Hurst, M.: A language model approach to keyphrase extraction. In: Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18, Association for Computational Linguistics (2003) 33–40
20. Verberne, S., Sappelli, M., Kraaij, W.: Query term suggestion in academic search. In: Proceedings of the 36th European Conference on IR Research, ECIR 2014. Volume 8416 of Lecture Notes in Computer Science. (2014) 560–566