Incorporating Revisiting Behaviors into Click Models *

Danqing Xu, Yiqun Liu, Min Zhang, Shaoping Ma,Liyun Ru
State Key Lab of Intelligence Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China P.R.
xudanqing06@gmail.com, {yiqunliu,z-m, msp}@tsinghua.edu.cn, lyru@vip.sohu.com

ABSTRACT

Click-through behaviors are treated as invaluable sources of user feedback and they have been leveraged in several commercial search engines in recent years. However, estimating unbiased relevance is always a challenging task because of position bias. To solve this problem, many researchers have proposed a variety of assumptions to model click-through behaviors. Most of these models share the sequential examination hypothesis, which is that users examine search results from the top to the bottom. Nevertheless, this model cannot draw a complete picture of information-seeking behaviors. Many eye-tracking studies find that user interactions are not sequential but contain revisiting patterns. If a user clicks on a higher ranked document after having clicked on a lower-ranked one, we call this scenario a revisiting pattern, and we believe that the revisiting patterns are important signals regarding a user's click preferences. This paper incorporates revisiting behaviors into click models and introduces a novel click model named Temporal Hidden Click Model (THCM). This model dynamically models users' click behaviors with a temporal order. In our experiment, we collect over 115 million query sessions from a widely-used commercial search engine and then conduct a comparative analysis between our model and several state-of-the-art click models. The experimental results show that the THCM model achieves a significant improvement in the Normalized Discounted Cumulative Gain (NDCG), the click perplexity and click distributions metrics.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]:

General Terms

Experimentation, Algorithms, Performance

Keywords

Click-through Behaviors, Revisiting Patterns, Temporal Hidden Click Model, Document Relevance

1. INTRODUCTION

With the explosive growth of information on the Web, search engines have become indispensable information acquisition tools for users. How to obtain an ideal ranking is always a challenging task with respect to search engines. Previous studies [5, 10, 20] have developed a number of ranking optimization algorithms for manually labeled data. However, the manual labeling process is both expensive and time-consuming, especially for updating some labels. Updating these labels is a necessary yet difficult job because some queries require the up-to-date results. For example, if a user submits "WSDM" as a query, results related to WSDM2012 are more likely to be expected documents than results related to WSDM2011. Click-through logs record user interactions with search engines and can be collected at a low cost. In addition, click-through data is treated as an important signal of users' click preferences [1, 2] because click-through data can provide fresh and timely information. As a result, click-through logs are widely adopted in both sponsored searches [6, 21] and Web searches [5, 7, 8, 9, 12, 13, 18].

Previous studies [5, 7, 8, 9, 12, 13, 18] show that click logs are informative but biased. This phenomenon is represented as follows: a higher ranked document has a higher probability to be examined and clicked even if it is not as relevant as lower ones. Based on position bias problem, researchers have proposed many click models to obtain an unbiased estimation of document relevance. The cascade model [8] assumes that each user examines the results from top to bottom sequentially. A strong assumption is that a user will end the current search session as soon as he clicks a document; therefore, this model is only suitable for single-click situations and cannot draw a whole picture of multi-click sessions. To solve this problem, the DCM model [13] proposes the following assumption: a user who clicks a document has a λ probability of continuing and 1-λ of abandoning the search. Here λ depends on the rank of this document. Thus, the DCM model [13] extends the cascade model to multi-click situation. Guo et al. [12] introduce the skipping behavior

*This work was supported by Natural Science Foundation (60903107, 61073071), National High Technology Research and Development (863) Program (2011AA01A205) and Research Fund for the Doctoral Program of Higher Education of China (20090002120005)

WSDM'12, February 8–12,2012,Seattle,Washington,USA.

into the CCM model by assuming that user can choose to click or skip according to the current relevance.

Eye-tracking studies [11, 17, 19] are used as direct microcosmic evidences for user behaviors. According to [17], the eye-tracking processing is categorized into two parts: the depth-first strategy and the breadth-first strategy. The depth-first model assumes that the user examines the result list from top to bottom and decides immediately whether to click. The breadth-first strategy is described as follows: the user looks ahead at a series of results and then revisits the most relevant results to click on. Most of previous click models [7, 8, 9, 12, 13] are rooted in the depth-first model. Lorigo et al. [19] performed a series of eye-tracking experiments and used scan path to characterize users' browsing behaviors. They found that only 34 percent of the scan paths are linear while over 50 percent of sessions contain revisiting (also named as regression behaviors in [19]) or skipping behaviors which cannot be covered by the depth-first model. Their experimental results indicated that a majority of users may not, in general, follow the presentation order. According to their findings, empirical seeking behavior is very complex, and the depth-first model is a simplification. The revisiting behaviors are acceptable supplements to the depth-first model. However, little work has been done for incorporating revisiting behavior into recent click models.

According to our analysis of click-through data containing over 115 million user sessions (described in Section 3.1), we discover that 24.1% of multi-click sessions contain revisiting behaviors. This result coincides with the eye-tracking study performed by [19]. Based on these findings, we can see that the revisiting behaviors cover a large number of search sessions and should not be ignored in the construction of a practical click model. As a result, we will introduce revisiting behaviors to solve the position bias problem. To the best of our knowledge, this study is the first attempt to incorporate revisiting behaviors into click models. For each position on a search engine results page (SERP), a user may move down for lower results, may stop and abandon the search or may review the higher ranked results. Previous models only estimate the probability of going down for lower results and the probability of stopping or abandoning the search. In our model, we will determine the probability of revisiting higher results. Thus, our model provides a more complete simulation of a user's information-seeking behaviors. Given M results in a SERP, the i-th result is accessed from two perspectives: going down from the higher ranked results and revisiting from the lower ones. In other words, the probability of being examined is estimated based on both the higher ranked (from 1 to i-1) and the lower-ranked (from i+1 to M) results. To make our model generative, we build up a new model with a dynamic temporal order. Our model is named Temporal Hidden Click Model (THCM), and it differs from previous models in two respects: 1. revisiting behaviors are taken into consideration; 2. user interaction is organized in a temporal order instead of ranking order.

The main contributions of our work are as follows:

i. A novel click model THCM is proposed to incorporate revisiting behaviors in the whole search process and improves the performance of the click models.

ii. The users' interactions are organized in a temporal order. This order is more reasonable than the ranking order because users do not always follow the ranking order.

iii. A practical method based on the THCM model is also proposed to solve the relevance inference and parameter estimation problems with an acceptable scalability in both time and space.

The remainder of this paper is organized as follows. We first present some important hypotheses and existing click models in Section 2. After providing a detailed description of our model in Section 3, the process of relevance inference and parameter estimation are represented in Section 4. In Section 5, we conduct experimental studies and evaluation, and we discuss and conclude our work in Section 6.

2. PRELIMINARIES

Algichtein et al. [2] were among the first researchers to utilize click-through logs to improve Web search rankings. They aggregated useful implicit feedback from the "noisy" user behaviors and found that incorporating the implicit feedback can help improve Web search performance. Baeza-Yates et al. [3] analyzed into users' Web searching process and modeled users' behaviors on user clicks, query formulations and page visited and other related features. Their experimental results showed that the aggregation of these features provided a valuable indicator of relevance preference. Joachims et al. [16] conducted eye-tracking experiments to track users' information-seeking behaviors. Their studies showed that clicks are informative but biased. In addition, to address the position bias problem, many researchers attempted to model the relationship between document relevance and click-through behaviors. In this section, we first give a general description of some important hypotheses and click models.

Basic Hypothesis: A document being clicked ($C_i = 1$) accords with($\rightarrow$) two conditions: it is examined ($E_i = 1$) and it is relevant ($R_i = 1$), and these two conditions are independent of each other.

$$C_i = 1 \rightarrow E_i = 1, R_i = 1 \tag{1}$$

$$E_i = 0 \rightarrow C_i = 0 \tag{2}$$

$$R_i = 0 \rightarrow C_i = 0 \tag{3}$$

Therefore,

$$P(C_i = 1) = P(E_i = 1)P(R_i = 1) \tag{4}$$

Examination Hypothesis: Each document at a given position has a certain probability of being examined, and this probability depends on its ranking position. A higher rank usually leads to a bigger examination probability. Taking this factor into the basic hypothesis, Equation 4 is rewritten as follows:

$$p(c|p, u, q) = p(e|p)p(r|u, q) \tag{5}$$

where $p(c|p,u,q)$ represents the click probability of document u at position p for a query q, $p(e|p)$ stands for the examination probability of the ranking position p and $p(r|u,q)$ is the probability of this (query,document) pair being relevant. Hence, if the result is relevant, a higher examination probability will bring more clicks. For click-through logs, only the click events are observed, while examination and relevance events are not. Therefore, estimating the examination probability is usually an important step for most click models to obtain unbiased document relevance estimations.

The cascade model [8] assumes that the first document is always examined and a user will end the search when he

clicks on a result. The corresponding examination hypothesis is as follows.

$$P(E_1) = 1 \tag{6}$$

$$P(E_{i+1} = 1|E_i = 1, C_i) = 1 - C_i \tag{7}$$

Here the (i+1)-th result being examined indicates the i-th result being examined yet not clicked. Although the cascade model has a good performance in predicting the click-through rates, this model is only suited for a single-click scenario.

Grounded in the cascade model, the DCM model [13] extends to model user interactions with multi-click sessions. Compared to the cascade model [8], the DCM model [13] assumes that a user may have a certain probability of examining the next one even if the current document is already clicked, and this probability is associated with the ranking position of the result. The DCM model is characterized as follows:

$$P(E_{i+1} = 1|E_i = 1, C_i = 0) = 1 \tag{8}$$

$$P(E_{i+1} = 1|E_i = 1, C_i = 1) = \lambda_i \tag{9}$$

where λ_i represents the preservation probability[1] of the position i and can be obtained through Equation 10 as follows.

$$\lambda_i = 1 - \frac{\#Query\ sessions\ when\ (last\ clicked\ position = i)}{\#Query\ sessions\ when\ position\ i\ is\ clicked} \tag{10}$$

Subsequently, the UBM model [9] makes further effort on the examination hypothesis. It is modified as Equation 11, where the event of the current document being examined depends on both the preceding click position and their corresponding distance.

$$P(E_i = 1|C_{1...i-1}) = \lambda_{r_i,d_i} \tag{11}$$

where r_i represents the preceding click position and d_i is the distance between the current rank and r_i. A total of $M * (M + 1)/2$(There exist M document in a SERP) global parameters need to be estimated, which makes the UBM model unfeasible for large-scale data. The BBM model [18] inherits the assumptions proposed by the UBM model and makes this model fit at the scalability of terabyte-scale data. This model is designed to estimate global parameters with a single pass of the large-scale data.

In contrast to the above models, the DBM model [7] is the first model to take presentation bias[2] into consideration. This model distinguishes the actual relevance(S_i) from the perceived relevance(R_i), where the perceived relevance indicates the relevance represented by abstracts or snippets in SERPs and the actual relevance is the relevance of the landing page. The user satisfaction is determined by the actual relevance, but the click events depend on its perceived relevance. The DBM model can be represented as follows:

$$R_i = 1, E_i = 1 \rightarrow C_i = 1 \tag{12}$$

$$P(R_i = 1) = r_u \tag{13}$$

$$P(S_i = 1|C_i = 1) = s_u \tag{14}$$

$$P(E_{i+1}|E_i = 1, S_i = 0) = \lambda \tag{15}$$

where S_i represents that whether or not the user is satisfied with the i-th document, s_u is the probability of this event, r_u is the probability of the perceived relevance(R_i), and λ represents the preservation probability.

Subsequently, the CCM model [12] presents a Bayesian inference to obtain the posterior distribution of the relevance. In contrast to other existing models, this model introduces skipping behaviors. According to [12], the CCM model is scalable for large scale click-through data. Moreover, the experimental results show that the CCM model is effective for low frequency(also known as long-tail) queries. Both the DBM model and the CCM model are two present common click models, and we analyze the performance of our model compared to these two ones in our experiments.

3. TEMPORAL HIDDEN CLICK MODEL

3.1 Revisiting Behaviors

Lorigo et al. [19] point out that different user groups have different information-seeking behaviors and that the majority of users have skipping or revisiting behaviors. In this section, we perform an experiment to analyze revisiting behaviors. First, a revisiting pattern can be defined as follows: a user clicks on another higher ranked result after having clicked on a lower result. With the help of a widely-used commercial Chinese search engine, we collect click-through logs from November 1st, 2010 to November 10th, 2010. The data set contains over 115 million query sessions. A query session is initialized when a user submits a query to a search engine, and query reformulations, re-submissions or a session lasting over 30 minutes will be regarded as a new session. For simplicity, we only record user interactions in Web organic results lists; other actions, such as ad clicks, are discarded in our experiments.

To have clearer statistics on revisiting behaviors, click-through logs were divided into 5 groups in terms to their own query frequencies. The statistics on revisiting behaviors of different query frequencies are shown in Table 1. The proportions of multi-click sessions are roughly equal among different query frequencies. On average, 24.1% of multi-click sessions contain the revisiting behaviors. This result is in accordance with the eye-tracking studies in [19], and indicates that revisiting behaviors are important parts of user interactions. Users may not follow the ranking order of a SERP in some cases.

In addition, we conduct a position-based analysis of revisiting behaviors. For each position, the corresponding document may be revisited from the lower ranked results and also has a certain probability of being skipped from the higher ranked ones. Conducted on the collected click-through data, the revisiting behaviors related probabilities on different positions are shown in Figure 1. The result shows that the top positions have a higher probability of being revisited, while the lower positions are more likely to be revisited from. In Figure 1, the first ranked documents are revisited from the lower ranked ones in 60.2% of all multi-click sessions, and the users who have clicked on the 10th result will perform revisiting behaviors in 34.7% multi-click sessions.

To examine the relevance influence from the revisited results, according to the time-series click events, we extract

[1]The probability of the (i+1)-th result being examined when the i-th document is clicked

[2]Different from the position bias, the presentation bias is a bias caused by the presentation form of the results list, such as the abstract, the snippet and so on

Table 1: Revisiting Features over Different Query Frequencies

Query Frequency	[1,9]	[10,30]	[31,99]	[100,499]	[500,∞)
Total query sessions	40,036,796	9,369,628	9,619,008	13,520,812	42,835,848
Single-click sessions	31,024,348	8,021,876	8,301,504	11,766,600	36,976,904
Multi-click sessions without revisiting behaviors	6,859,335	1,030,918	1,017,741	1,304,911	4,402,210
Multi-click sessions with revisiting behaviors	2,153,113	316,834	299,763	449,301	1,456,734
The proportion of sessions with revisiting behaviors	0.239	0.235	0.228	0.256	0.249

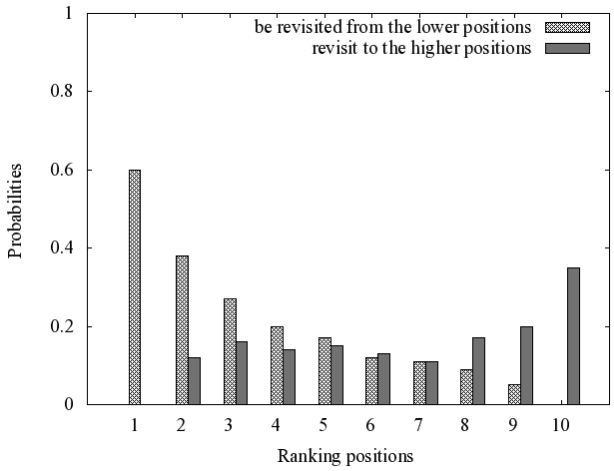

Figure 1: Revisiting behaviors related probabilities on different positions in a SERP

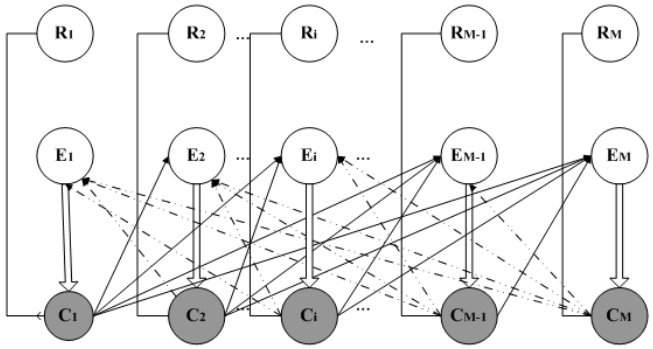

Figure 2: The graphical model by introducing revisiting behaviors. Here the solid line represents the forward event and the dotted line stands for the backward event(revisit). Observed click variables C_i are shaded.

10,450 revisited items from the collected click-through data. Each item is presented in the form: (query, URL, relevance), where the relevance is manually labeled by three professional assessors. The details of labeling rules are available in Section 5.1. These relevance labels are divided into 5 levels(bad, fair, good, excellent, perfect). Through our statistics, 50.2% of these revisited items have a high relevance(excellent or perfect) with the original query, and only 14.5% items show bad performance in relevance(bad).

Summarizing the above studies, we find that revisiting behaviors are relatively common, and have a strong correlation with the relevance. As a result, revisiting behaviors may be important signals of click preferences. Based on this fact, we will incorporate revisiting behaviors into our click models.

3.2 Model Specification

We first introduce some definitions and notations. Here a session is treated as a unit in our model, and it records the whole user interaction with top-M results (usually M=10). In our model, all of our variable sets can be represented as the sequences. The presented results are represented as an impression sequence: $A =< a_1, a_2, ..., a_i, ..., a_M >$, i corresponds to the ranking position and a_i is ranked higher than a_j if $i < j$. According to the timestamps, clicks can also be re-organized as a temporal click sequence: $C =< C_1, C_2, ..., C_t, ..., C_T >$, where t is a discrete time variable and C_t represents the corresponding ranking of the result being clicked at time t. In addition, we introduce another variable sequence: $U =< U_1, U_2, ..., U_t, ..., U_T >$, where U_t represents the ranking position of the preceding click, i.e, $U_t = C_{t-1}$. The examination sequence $E =< E_1, E_2, ..., E_t, ..., E_T >$, where E_t is an M dimensional vector. $E_t = (E_{t1}, E_{t2}, ..., E_{ti}, ..., E_{tM})$ is used to represent examination events at time t.

Figure 2 shows the graphic model by incorporating revisiting behaviors. The whole user interactions are no longer strictly from top to bottom. In a SERP, a user examines the i-th document, followed by examining the j-th document. If $i < j$, we will define this event as the forward event; otherwise, we call it as the backward event. In Figure 2, both forward examination probability and backward examination probability constitute the whole examination probability. Given a position i, it may be examined from the (i-1)-th document, and also be revisited from lower documents. We can see that there can be some loops in Figure 2, and previous approaches will be difficult for solving generative process. Hence we need to provide a temporal generative process which is illustrated in Figure 3. We call it Temporal Hidden Click Model(THCM) and the corresponding hypotheses are as follows.

$$p(E_{t(i+1)} = 1 | E_{ti} = 1) = \alpha \tag{16}$$

$$p(E_{t(i-1)} = 1 | E_{ti} = 1) = \gamma \tag{17}$$

$$0 \leq \alpha + \gamma \leq 1, \alpha \geq 0, \gamma \geq 0 \tag{18}$$

$$p(C_t | C_1, C_2, ..., C_{t-1}) = p(C_t | C_{t-1}) \tag{19}$$

$$p(C_t = i) = p(E_{ti} = 1).p(R_i = 1) \tag{20}$$

For the THCM model, if a user examines a document, then he will have a α probability to examines the next one(Equation 16). Different from previous models, we introduce the revisiting probability γ: if the user examines a result, he may

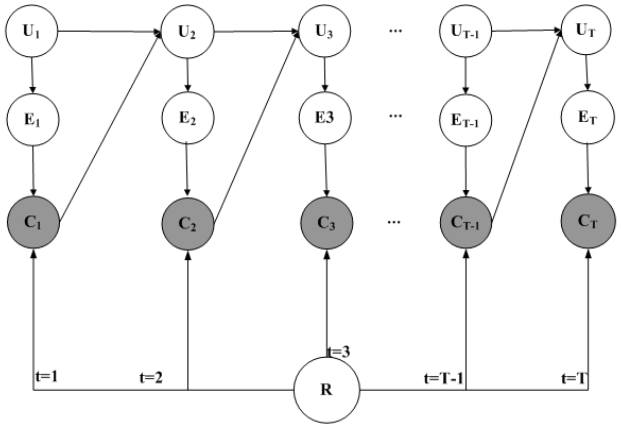

Figure 3: The graphical model on the temporal order. C_i represents the observed click events at time t, while E_t implies the examination vector for M results at time t, and U_t represents the rank of the click events at time $t-1$.

also have a γ probability of revisiting the preceding rank result(Equation 17). The relationship between α and γ needs to satisfy Equation 18. To make our model simple, we introduce the **First-order Click Hypothesis**: the click event at time t are determined by that at time $t-1$(Equation 19). Similar with previous models, if a result is both examined and relevant, the user will click this result(Equation 20). Thus, the probability of the event that the results ranked lower than C_{t-1} is examined can be calculated as follows:

$$\begin{aligned} &p(E_{ti}=1|C_{t-1}<i) \\ &=\prod_{j=U_t}^{i-1} p(E_{t(j+1)}=1|E_{tj}=1)=\alpha^{i-C_{t-1}} \end{aligned} \tag{21}$$

In Equation 21, the probability of a lower result being examined is determined by both the forward probability α and the distance between current rank and the previous clicked position. In our model, we assume that the relevance event(R_i) is a binary variable. If there exist follow-up clicks for a clicked result a_m, the user may not be satisfied with this result, i.e., $R_{a_m}=0$. Thus, given a position i at time t, if the preceding clicked position C_{t-1} is ranked higher than i, this document is clicked by forward examination events(Equation 22); otherwise, the i-th document is revisited by backwards examination events(Equation 23).

$$\begin{aligned} &p(C_t=i|C_{t-1}<i) \\ &=p(R_{C_{t-1}}=0).\underbrace{p(C_t=i|E_{ti}=1)}_{R_i}.\underbrace{p(E_{ti}=1|C_{t-1}<i)}_{\alpha^{i-C_{t-1}}} \\ &=(1-R_{C_{t-1}}).R_i.\alpha^{i-C_{t-1}} \end{aligned} \tag{22}$$

$$\begin{aligned} &p(C_t=i|C_{t-1}>=i) \\ &=p(R_{C_{t-1}}=0).\underbrace{p(C_t=i|E_{ti}=1)}_{R_i}.\underbrace{p(E_{ti}=1|C_{t-1}>=i)}_{\gamma^{C_{t-1}-i}} \\ &=(1-R_{C_{t-1}}).R_i.\gamma^{C_{t-1}-i} \end{aligned} \tag{23}$$

In our model, α and γ are global parameters for each query session, and represent the forward and backward examination probabilities, respectively. In the next section, we will assume that the document relevance $\mathbf{R}$ and the click events of different sessions are independent of each other. Based on this assumption, we will discuss the inference of document relevance and introduce a feasible algorithm that has a acceptable scalability with using a large volume of click logs.

4. ALGORITHMS

4.1 Relevance Inference

The goal of the click models is to obtain an unbiased estimation of the relevance for each query-document pair. Previous click models all require the following assumption: users examine a document and decide to click or skip, but they cannot revisit previous documents. According to our statistics on revisiting behaviors in Section 3.1, revisiting behaviors may be an important feedback for Web search performance. Given the click-through data $C^{1\cdots n}$, incorporating revisiting behaviors into previous models will increase the computational complexity of the posterior probability over $\mathbf{R}$. To address this problem, our THCM model re-organizes the click sequences into a dynamical temporal order.

Given a query with N sessions, $A=<A^1,A^2,...,A^N>$ and $C=<C^1,C^2,...,C^N>$ represent the corresponding impression and temporal click sequences, respectively. We assume that the click sequences of different sessions are independent of each other. Thus, according to the Bayes principle, the posterior probability is calculated as follows.

$$p(\mathbf{R}|C^{1,...,N}) \propto p(\mathbf{R})p(C^{1,...,N}|\mathbf{R}) \tag{24}$$

Because $p(\mathbf{R})$ is a known prior, we assume that $p(\mathbf{R})$ follows the prior beta distribution, thus we need to compute $p(C^{1,...,N}|\mathbf{R})$. Since click sequences of different sessions are treated as conditionally independent variables for each query session given $\mathbf{R}$, we obtain the following equation:

$$p(C^{1,...,N}|\mathbf{R}) \propto \prod_{n=1}^{N} p(C^n|\mathbf{R}) \tag{25}$$

where C^n represents the click sequence of the n-th session. In the following steps, we assume that the user clicks the m-th document at time t, i.e, $C_t=a_m$. According to the First-order Click Hypothesis, the relevance of the t-th clicked document is related to both the preceding clicked document(C_{t-1}) and next clicked document(C_{t+1}), and is independent of the remaining documents. Therefore, we obtain Equation 26 as follows.

$$\begin{aligned} &p(C^n=<C_1,C_2,...,C_t,...,C_T>|R_m,A^n) \\ &=p(C^n=<C_1,C_2,...,C_t=a_m,...,C_T>|R_m,A^n) \\ &=\prod_{t=1}^{T-1} p(<C_{t-1},C_t=a_m,C_{t+1}>)|R_m,A^n) \end{aligned} \tag{26}$$

Here we will use the symbol $\square$ to represent no click and the symbol $\times$ to represent any click. According to the rankings of different times, a click sequence $C=<C_{t-1},C_t=a_m,C_{t+1}>$ can be categorized into different behavior patterns. Table 2 lists the set of these behavior patterns, and we only present the derivation of Cases 1-3 due to space limitations.

Table 2: Different Cases for Computing $p(C|R_m, A)$ on Different Click Sequences

Cases	Sequences	Results
1	$< \square, a_m, \square >$	$R_m\alpha^m$
2	$< \square, a_m, a_j >, j > m$	$\frac{R_m \cdot (1-R_m)(\alpha^{m+1}-\alpha^{M+1})}{2(1-\alpha)}$
3	$< \square, a_m, a_j >, j < m$	$\frac{R_m \cdot (1-R_m)(\gamma-\gamma^m)}{2(1-\gamma)}$
4	$< a_j, a_m, \times >, j < m$	$\frac{R_m \cdot (\alpha-\alpha^m)}{6(1-\alpha)}$
5	$< a_j, a_m, \times >, j > m$	$\frac{R_m \cdot (\gamma-\gamma^{M-m+1})}{6(1-\gamma)}$

Case 1: $t = 1$, $T = 1$, $C =< \square, a_m, \square >$, $C_0 = 0$, that is, the m-th document is the first and only click in this session. $C_1 = a_m$, According to Equations 22, $p(C =< \square, a_m, \square >) = p(C_1 = a_m|C_0 = 0)$, and the document a_m is examined through the forward event.

$$\begin{aligned} &p(C =< \square, a_m, \square > |R_m, A) \\ &= p(C_1 = m|E_{1m} = 1)p(E_{1m} = 1|U_1 = C_0 = 0) = R_m\alpha^m \end{aligned} \tag{27}$$

Case 2: $t = 1$, $T > 1$, $C =< \square, a_m, a_j >$, $j > m$, $C_0 = 0$, that is, the m-th document is the first click and the next clicked document is ranked lower than m. This event can be explained by Equations 19 and 22, $C_1 = a_m$, $C_2 = a_j$. This sequence is divided into two parts: $< \square, a_m >$ and $< a_m, a_j >$. The probability of this case can be described as follows:

$$\begin{aligned} &p(C =< \square, a_m, a_j > |R_m, A) \\ &= \underbrace{p(< \square, a_m > |R_m, A)}_{R_m \cdot \alpha^m} \underbrace{p(< a_m, a_j > |R_m, A)}_{(1-R_m)R_j\alpha^{j-m}} \\ &= R_m(1 - R_m)R_j\alpha^m\alpha^{j-m} = R_m(1 - R_m)R_j\alpha^j \end{aligned} \tag{28}$$

To calculate $p(C|R_m, A)$, other hidden random variables, such as R_j, need to be integrated out in Equation 29.

$$\begin{aligned} &p(C =< \square, a_m, a_j >, j > m|R_m, A) \\ &= \sum_{j=m+1}^{M} p(C =< \square, a_m, a_j > |R_m, A) \\ &= \sum_{j=m+1}^{M} \int_{R_j=0}^{1} R_m(1 - R_m)R_j\alpha^j \\ &= \frac{R_m(1 - R_m)(\alpha^{m+1} - \alpha^{M+1})}{2(1 - \alpha)} \end{aligned} \tag{29}$$

Case 3: $t = 1$, $T > 1$, $C =< \square, a_m, a_j >$, $j < m$, that is, the m-th document is the first click and the next clicked document is ranked higher than m. This click event comes from the revisiting events. According to Equations 19,22 and 23, this sequence can also be divided into two parts: $< \square, a_m >$ and $< a_m, a_j >$.

$$\begin{aligned} &p(C =< \square, a_m, a_j > |R_m, A) \\ &= \underbrace{p(< \square, a_m > |R_m, A)}_{R_m\alpha^m} \underbrace{p(< a_m, a_j > |R_m, A)}_{(1-R_m)R_j\gamma^{m-j}} \\ &= R_m(1 - R_m)R_j\alpha^m\gamma^{m-j} \end{aligned} \tag{30}$$

By integrating and summing the hidden random variable R_j, the probability can be represented as follows.

$$\begin{aligned} &p(C =< \square, a_m, a_j >, j < m|R_m, A) \\ &= \sum_{j=1}^{m-1} p(C =< \square, a_m, a_j > |R_m, A) \\ &= R_m(1 - R_m)\alpha^m \sum_{j=1}^{m-1} \int_{R_j=0}^{1} R_j\gamma^{m-j} \\ &= \frac{R_m(1 - R_m)(\gamma - \gamma^m)}{2(1 - \gamma)} \end{aligned} \tag{31}$$

Cases 4 and 5 are also obtained by taking a summing and integrating over the corresponding hidden random variables. In addition, different sessions of a given query are independent of each other. Thus, the posterior of R_m has the following un-normalized form:

$$p(R_m|C, A) \propto \prod_{i=1}^{5} O_i(R_m)^{N_i^m} \tag{32}$$

where O_i is the closed form of $p(C|R_m, A)$ for i-th kind of click sequence in Table 2.

4.2 Parameter Estimation

In this section, we adopt the maximum-likelihood (ML) algorithm to estimate the global parameters α and γ. By integrating the hidden random variable R_m and summing the log likelihood of all of the click sequences, the whole log-likelihood function is illustrated in Equation 33.

$$\begin{aligned} L(\alpha, \gamma) = \sum_{m=1}^{M} \{ &N_1^m log[\frac{\alpha^m}{2}] + N_2^m log[\frac{\alpha^{m+1} - \alpha^{M+1}}{12(1 - \alpha)}] \\ &+ N_3^m log[\frac{\gamma - \gamma^m}{12(1 - \gamma)}] + N_4^m log[\frac{\alpha - \alpha_m}{12(1 - \alpha)}] \\ &+ N_5^m log[\frac{\gamma - \gamma^{M-m+1}}{1}]\} \end{aligned} \tag{33}$$

Through taking the derivation of the above likelihood function on α, we get the following equation:

$$\begin{aligned} &N_1^m \frac{2m}{\alpha} + (N_2^m + N_4^m)\frac{1}{1 - \alpha} + N_4^m \frac{1 - m\alpha^{m-1}}{\alpha - \alpha^m} \\ &+ N_2^m \frac{(m + 1)\alpha^m - (M + 1)\alpha^M}{\alpha^{m+1} - \alpha^{M+1}} = 0 \end{aligned} \tag{34}$$

The equation about γ can be obtained through the same approach. However, we find that there exists no closed form for our global parameters α and γ. Thus, we need to seek for an approximate solution. According to Equation 18, the feasible zone is convex and the log likelihood is a concave function; hence, solving for the parameters is a convex optimization problem. In our experiment, we use the primal-dual interior-point method [4] to obtain approximate values for parameters α and γ.

5. EXPERIMENTS

5.1 Experiment Setup

As described in Section 3.1, we collect click-through logs from November 1, 2010 to November 10, 2010, including 93 million queries with 115 million query sessions. Taking the users' privacy into consideration, we only collect the following information for each session: the original query, the top

10 documents in the first SERP, the timestamps and the list of which documents are clicked. In our experiments, 2,000 queries and 47,891 related documents are randomly sampled as our data set. 10,450 items are revisited among these query-document pairs, and they are selected as the data set of revisiting behaviors in Section 3.1. The frequency distribution of these randomly selected queries is shown in Figure 4. The frequency range is from 1 to 62,688, indicating that our data set contains different queries of high, middle and low frequencies. Three professional assessors took two weeks to label the relevance of these sampled query-document pairs. Each of them labeled 1,335 queries and their corresponding results, and the average Kappa coefficient is 0.68.

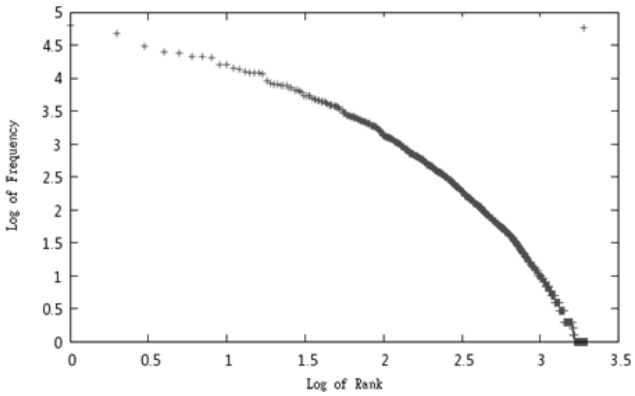

Figure 4: Frequency distribution of randomly sampled queries

According to our statistics in Section 3.1, revisiting behaviors may be important signals to reflect click preferences. Previous models usually discard sessions that contain revisiting behaviors because these sessions cannot satisfy the depth-first examination hypotheses. However, these sessions are preserved in our experiments, and our model can explain revisiting behaviors well. What's more, the method of 5-fold cross-validation is used in our experiments. Compared to the DBM model [7] and the CCM model [12], the performance of our model is evaluated in different aspects, including the NDCG values, the click perplexity and the click distribution of position-bias.

5.2 Evaluation on NDCG

The Normalized Discounted Cumulative Gain (NDCG, [15]) is always an important metric to measure the relevance of ranking functions. The NDCG at position $p(NDCG_p)$ is computed as:

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(1+i)} \tag{35}$$

$$NDCG_p = \frac{DCG_p}{IDCG_p} \tag{36}$$

where rel_i is the manual label of the i-th document, and $IDCG_p$ represents an ideal DCG value obtained when sorting the documents by relevance. The NDCG metric indicates that the rankings of higher relevance results play more important roles than that of the lower relevance ones. For example, if a high relevance document is ranked low, the NDCG value becomes relatively small and then the performance of this ranking function will become relatively poor. The NDCG value measures the distance between the current ranking and an ideal ranking, and this value of an ideal ranking is 1.0. First, we report on the changes of the NDCG values of the THCM, DBM and CCM models.

Table 3: P-values on Different Positions

$NDCG_{pos}$	THCM over DBM	THCM over CCM
$NDCG_1$	$1.30 * 10^{-2}$	$5.20 * 10^{-3}$
$NDCG_3$	$4.00 * 10^{-4}$	$7.00 * 10^{-4}$
$NDCG_5$	$8.00 * 10^{-4}$	$2.30 * 10^{-3}$
$NDCG_{10}$	$1.02 * 10^{-2}$	$4.00 * 10^{-3}$

Table 4: P-values on Different Query Frequencies

$frequency_q$	THCM over DBM	THCM over CCM
[1,9]	$\mathbf{9.00 * 10^{-6}}$	$\mathbf{4.00 * 10^{-3}}$
[10,30]	$\mathbf{5.00 * 10^{-2}}$	$\mathbf{2.70 * 10^{-2}}$
[31,100]	$\mathbf{2.00 * 10^{-2}}$	$\mathbf{1.20 * 10^{-2}}$
[100,499]	$8.90 * 10^{-1}$	$5.00 * 10^{-3}$
[500,∞)	$3.30 * 10^{-1}$	$3.00 * 10^{-5}$

Given each query, we estimate the relevances of all corresponding documents based on these three models. For the THCM model, on the training set, we learn the model parameters by the method of Section 4.2, and then obtain estimated relevances on the test set. The implementations of the DBM and CCM models are available in [7, 12]. Figures 5 and 6 show the performance on the NDCG values of different positions and different query frequencies, respectively. Figure 5 shows that all of $NDCG_1$, $NDCG_3$, $NDCG_5$ and $NDCG_{10}$ of our model show more improvement than that of the other two models. The $NDCG_1$ value of our model is 0.841, with 8.52% and 25.0% improvement over DBM and CCM, respectively. Figure 6 shows that our model improves the performance on long-tail queries. As described in [12], the CCM model has a commendable performance on the tail queries, which can also be seen in Figure 6. Compared to the CCM model, our model attains a 1.50% improvement of $NDCG_5$. For tail queries, related click-through behaviors are also sparse, and the user interaction may be fairly complex. Thus, revisiting behaviors may provide a key evidence for predicting click preference for long-tail queries. However, our model shows relatively poor performance for high frequency queries. For high frequency queries, compared to enough click-through data, different types of revisiting behaviors may be "noisy" for estimating document relevance.

In addition, we use the t-test to quantify whether our model is better than the others. Having no significant difference is denoted as the null hypothesis, with a significant difference as the alternative hypothesis. If the p-value is less than $5.0 * 10^{-2}$($5.0 * 10^{-2}$ is usually set as the critical level), the null hypothesis will be rejected and the alternative hypothesis will suffice. The testing values for NDCGs on different positions and query frequencies are illustrated in Tables 3 and 4. The result shows that our model may be no significant improvement for high frequency queries, but the improvement is relatively sharp for moderate frequency and rare frequency queries.

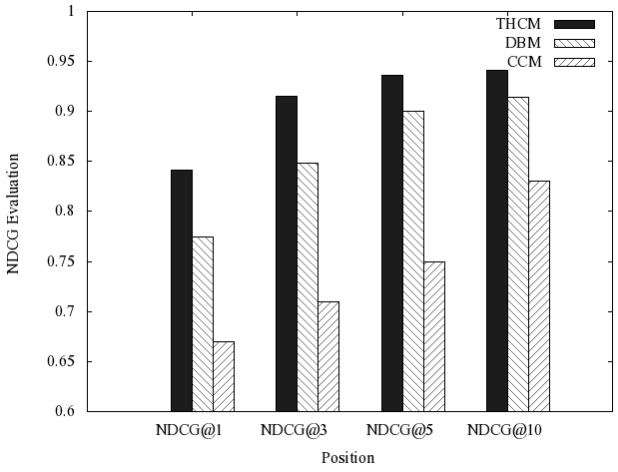

Figure 5: Performance comparison of THCM, DBM and CCM with $NDCG_n$

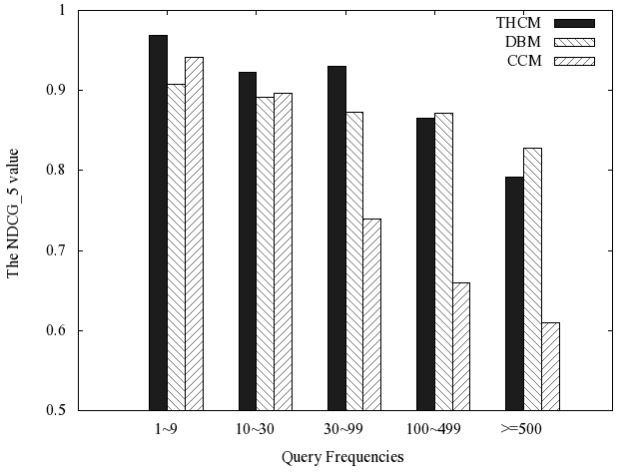

Figure 6: Performance comparison of THCM, DBM and CCM for queries with different frequencies

5.3 Evaluation on Click Perplexity

In addition to the NDCG value, the click perplexity [12] is another metric to evaluate the performance of each position. A smaller perplexity value indicates a better performance, and the value reaches 1 in an ideal case. In our experiment, the click-through logs are organized into a temporal order for each query session. The click perplexity at a given position is computed as the following equation.

$$CP_i = 2^{-\frac{1}{U}\cdot\sum_{u=1}^{U}(C_i^U.log_2 p_i^u + (1-C_i^U.log_2(1-p_i^u)))} \quad (37)$$

where CP_i is the click perplexity of position i, U is the total number of sessions for a given query. C_i^U represents the click event of the i-th document, and p_i^u is the corresponding probability. The improvement of click perplexity CP_1 over CP_2 is calculated through $\frac{CP_2-CP_1}{CP_2-1} * 100\%$, and the average click perplexity is obtained using the arithmetic mean. Based on our experimental data, the average perplexities on different positions are illustrated in Figure 7.

In Figure 7, we can see that our THCM model performs best for all of the positions. The average perplexity of all positions is 1.1402 for THCM, 1.28 for DBM and 1.237 for CCM. What's more, the perplexities have a varying degree of improvement, with an 18% improvement over the DBM model and a 7.8% improvement over the CCM model for the first position. The click perplexity of the THCM model indicates that temporal click sequences will obtain a better click prediction.

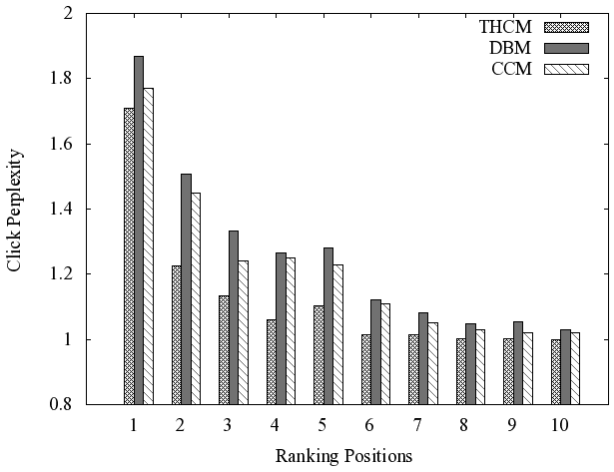

Figure 7: Performance comparison of THCM, DBM and CCM with click perplexity of different positions

5.4 Results on Position Bias

To address the position bias problem, the click distributions of different positions are also introduced to evaluate the effect of click models [12]. Given a position, the click distribution metric is quantified as follows: collect all of the sessions on the test set, calculate the click events (which documents will be clicked) according to learned model parameters in Section 4 and count the total number of the document at the given position being clicked. The click probabilities of a given position can be estimated through the total number of the position being clicked divided by the total number of the test set. Different click models can estimate different click distributions, and empirical click distributions can be obtained using actual click behaviors. Thus, the distance between the estimated click distributions of click models and that of empirical clicks can measure the prediction effect, and the smaller distance indicates more accurate click predictions.

Figure 8 illustrates that the click distributions derived from THCM, DBM and empirical click on the test set. We can see that all of the click distributions have a position bias, with a document that is ranked higher having a higher click probability compared to the lower documents. In Figure 8, the THCM model has a closer match with empirical clicks than the DBM model has on different positions, and their correlation coefficient reaches 0.99. The THCM model provides a temporal perspective on users interactions, which is closer to actual user's information-seeking behaviors. The results on the click distribution of position-bias further verifies that the THCM model may produce a better click prediction.

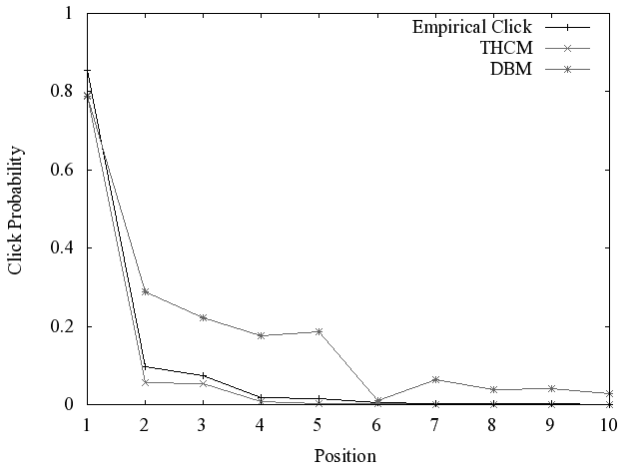

Figure 8: Click probability distribution of THCM and DBM on the top 10 positions

6. CONCLUSIONS AND DISCUSSIONS

In this paper, a statistical investigation on large-scale click-through data was presented, indicating that revisiting behaviors are important aspects of user interaction. Based on this finding, we proposed the novel THCM model, which models the temporal click sequences and incorporates revisiting behaviors. This paper also provides a solution to the estimation of document relevance and other relevant parameters. The experimental results on huge-volume click-through data show that our model outperforms existing models in a number metrics, including the NDCG value, the click perplexity and the click distributions of different positions.

Despite these successes, several assumptions of the THCM model need to be further discussed. In our experiment, we introduce the First-order Click Hypothesis, which indicates that the current state only depends on the preceding state. However, empirical click events may not be independent of each other. To facilitate describing and solving problems, we simplify this process to a first-order model. In the future, we will move forward to multi-order models.

According to [14], information needs are classified into three categories: navigational, informational and transactional. Different informational needs tend to generate different types of user interactions. For a navigational query, the information demand is relatively concentrated, so the corresponding behaviors may be simple. However, for informational queries, the relatively dispersed resources may lead to more clicks, and it is more likely that revisiting behaviors occur. In our model, the method for calculating the global parameters α and γ is a globally optimal solution, but it may be not efficient for all individual users. In the future, we will distinguish the global parameters from different categorizations of user behaviors; this development will be a new direction that we will explore in the future.

7. REFERENCES

[1] E. Agichtein, E.Brill, and D.Susan. Improving web search ranking by incorporating user behavior information. *In proceeding of SIGIR06*, 2006.

[2] E. Agichtein, E.Brill, S.Dumais, and R.Ragno. Learning user interaction models for predicting web search result preferences. *In proceeding of SIGIR06*, 2006.

[3] R. Baeza-Yates, C. Hurtado, M. Mendoza, and G. Dupret. Modeling user search behavior. *In LA-WEB 2005*, 2005.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] B. Carterette and R. Jones. Evaluating search engines by modeling the relationship between relevance and clicks. *Advances in Neural Information Processing Systems*, 2008.

[6] D. Chakrabarti, D. Agarwal, and V. Josifovski. Contextual advertising by combining relevance with click feedback. *In proceeding of WWW08*, 2008.

[7] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. *In proceedings of WWW2009*, 2009.

[8] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. *In proceedings of WSDM2008*, 2008.

[9] G. Durpret and B. Piwowarski. A user browsing model to predict search engine click data from past observations. *In proceedings of SIGIR2008*, 2008.

[10] Y. Freund, R. Iyer, R. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *The Journal of Machine Learning Research*, 2003.

[11] L. Granka, T. Joachims, and G. Gay. Eye-tracking analysis of user behavior in www search. *In proceeding of SIGIR04*, 2004.

[12] F. Guo, C. Liu, A. Kannan, T. Minka, M.Taylor, Y.Wang, and C. Faloutos. Click chain model in web search. *In proceedings of WWW2009*, 2009.

[13] F. Guo, C. Liu, and Y. Wang. Efficient multiple-click models in web search. *In proceedings of WSDM09*, 2009.

[14] B. J. Jansen and D. Booth. Classifying web queries by topic and user intent. *In the proceeding of CHI2010*, 2010.

[15] K. Jarvelin and J. Kelalainen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information System*, 2002.

[16] T. Joachims, L. Granka, B. Pan, H.Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. *In proceeding of SIGIR05*, 2005.

[17] K. Klockner, N. Wirschum, and A. Jameson. Depth-and breadth-first processing of seach result lists. *In proceeding of CHI04*, 2004.

[18] C. Liu, F. Guo, and C. Faloutsos. Bbm: Beyesian browsing model from petabyte-scale data. *In proceedings of KDD09*, 2009.

[19] L. Lorigo, B. Pan, H. Kembrooke, T. Joachims, L. Granka, and G. Gay. The influence of task and gender on search and evaluation behavior using google. *Information Processing and Management*, 2005.

[20] F. Radlinski and T. Joachims. Query chains: learning to rank from implicit feedback. *In proceeding of KDD2005*, 2005.

[21] M. Richardson, E. Dominowska, and R.Rango. Predicting clicks: estimating the click-through rate for news ads. *In proceeding of WWW07*, 2007.