Constructing a Comparison-based Click Model for Web Search

Ruizhe Zhang[†],Xiaohui Xie[†],Jiaxin Mao[‡⋆],Yiqun Liu[†],Min Zhang[†],Shaoping Ma[†]
[†]Department of Computer Science and Technology, Institute for Artificial Intelligence,
Beijing National Research Center for Information Science and Technology,
Tsinghua University
Beijing, China
[‡]Beijing Key Laboratory of Big Data Management and Analysis Methods,
Gaoling School of Artificial Intelligence,
Renmin University of China
Beijing, China
maojiaxin@gmail.com

ABSTRACT

Extracting valuable feedback information from user behavior logs is one of the major concerns in Web search studies. Among the tremendous efforts that aim to improve search performance with user behavior modeling, constructing click models is of vital importance because it provides a direct estimation of result relevance. Most existing click models assume that whether or not users click on results only depends on the examination probability and the content of the result. However, through a carefully designed user eye-tracking study, we found that users do not make click-through decisions in isolation. Instead, they also consider the context of a result (e.g., adjacent results). This finding leads to the design of a novel click model named Comparison-based Click Model (CBCM). Different from traditional examination hypotheses, CBCM introduces the concept of an examination viewport and assumes users click results after comparing adjacent results within the same viewport. The experimental results on a publicly available user behavior dataset demonstrate the effectiveness of CBCM. We also public our code of CBCM and dataset.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

Click models, Eye tracking

ACM Reference Format:
Ruizhe Zhang[†],Xiaohui Xie[†],Jiaxin Mao[‡⋆],Yiqun Liu[†],Min Zhang[†],Shaoping Ma[†]. 2021. Constructing a Comparison-based Click Model for Web Search. In *Proceedings of the Web Conference 2021 (WWW '21), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3442381.3449918

WWW '21, April 19–23, 2021, Ljubljana, Slovenia

ACM ISBN 978-1-4503-8312-7/21/04.
https://doi.org/10.1145/3442381.3449918

1 INTRODUCTION

Click-through data is routinely logged in Web searches. Using this log data for improving the performance of search engines has become popular since it can provide abundant, implicit user feedback and can be collected at a very low cost. In particular, users' click-through information have been successfully adopted to improve web search in topics such as result ranking [1, 20], query recommendation [4, 24], query auto-completion [14, 17], optimizing presentations [23] etc.

Although click-through data can provide implicit feedback of users' click preferences, accurate and absolute judgments regarding the relevance of query-document pairs are difficult to derive. This is due to the existence of inherent biases in user behavior [15]. Users' click behavior can be biased in the following aspects: (1) Position bias: Users' attention decreases from top to bottom so that a result displayed at a lower rank receives less attention and is less likely to receive clicks than a result at a higher position. (2) Appearance bias: Different vertical results may have different presentation styles which can affect users' examination behavior and lead to different attention allocation mechanisms. (3) Trust bias: The reputations of web sites where results come from can influence users' click preference.

To alleviate the aforementioned biases in click-through data, click models [8] following certain examination assumptions have been proposed to derive accurate examination probabilities and relevance estimations of a given query-result pair. Most click models follow an examination hypothesis [9] in which a search result being clicked accords with two independent conditions: it is examined and it is relevant. Craswell et al. [9] propose a cascade model which assumes that search users examine results following a top to bottom pattern and decide whether or not to click a result immediately.

Expanding upon the cascade assumption, Dependent Click Model (DCM) [12] assumes that a user may have a certain probability to examine the next result after clicking the current result. In that regard, DCM can deal with multi-click search sessions. Moreover, User Browsing Model [10] assumes that the probability of a given result being examined depends not only on the previous click position but also on the distance between the previous clicked result and the current one. Besides position bias, Chapelle and Zhang [5] propose Dynamic Bayesian Network Model (DBN) to take appearance bias into consideration. They define two facets: actual relevance and perceived relevance. While actual relevance depicts the relevance of the landing page, the perceived relevance indicates relevance represented by titles or snippets in SERPs.

Furthermore, click models have been investigated to ease the bias caused by layouts and presentation forms of vertical results. For example, Wang et al. [22] propose Vertical-aware Click Model (VCM) which models the effect of different vertical results (i.e., text vertical, multimedia vertical, and application vertical) on the sequence and probability of examination. Chen et al. [6] show that user click behavior on vertical results may lead to a stronger indication of relevance due to their presentation style; they propose Federated Click Model (FCM).

Although the above click models achieve promising performance, they only consider results in an isolated manner. In other words, patterns of comparing, revisiting, and clicking results are not considered in their design.

To explore this, we conduct a lab-based eye-tracking study. Based on the eye-tracking data, we found that search users consider the context of a given result instead of making the click decision only based on the result itself. Especially, we found that search users adopt a "compare" examination pattern i.e., they compare adjacent search results. This "compare" pattern is different from the "revisit" [25] pattern since the latter one mainly focuses on recalling the examined results rather than comparing different results. Furthermore, eye-tracking data show that the comparison more likely occurs at the top two results. Users examine the top two results several times in a period of time. Motivated by these observations, we propose a novel click model named comparison-based click model (CBCM). The construction of CBCM is based on two assumptions: (1) From the global perspective, users examine search results following a top to bottom pattern. (2) From the local perspective, users occasionally compare adjacent results.

The comparison between two results have an effect on users' next decision (i.e., examining more results or triggering a click behavior). We conduct extensive experiments to test the performance of our model using a large-scale commercial image search log. The experimental results demonstrate that in terms of behavior prediction (perplexity) and relevance estimation (normalized discounted cumulative gain (nDCG) [13]), CBCM outperforms the state-of-the-art baseline models.

To summarize, the main contributions of this work are as follows:

- We carry out a lab-based eye-tracking study to thoroughly investigate search users' fine-grained examination pattern. We show that users consider the context of a result rather than judge each result in isolation.
- Motivated by our findings, we propose a novel click model named CBCM which considers users' comparison between adjacent results.
- We conduct extensive experiments to test the performance of the proposed CBCM. The experimental results demonstrate that CBCM can better predict click probability and estimate result relevance. The underlying assumptions are closer to practical user behavior than the assumptions made in competing models.

The rest of the paper is organized as follows. In Section 2, we review previous related work. We outline observations from our eye-tracking user study in Section 3. In Section 4, we formally introduce CBCM. We report on experiments using CBCM and compare the results with results of existing models in Section 5. Finally, conclusions and future work are discussed in Section 6.

2 RELATED WORK

We briefly summarize related work in two categories: user behavior in web search and existing click models. The former concentrates on observations from user behavior data collected from search log or user study. The latter reviews both sequential and non sequential click models.

2.1 User behavior in web searches

There's an abundance of relevant information hidden in users' click logs. Search engines collect behavior data in order to improve ranking performance. Click models are proposed to alleviate the position bias, i.e., after the search engine returns the SERP, the user is more inclined to click the top-ranked result.Craswell et al. [9], Joachims et al. [16] To serve this purpose, we need to use mathematical methods to model user behavior and try to alleviate the bias in the user browsing process. Through a series of experiments, Richardson et al. [21] show that the user's process of clicking on a single result consists of two steps: examining the result and verifying whether the result is relevant to the query. This is called the examination hypothesis. The two parts are represented in this paper by the binary random variables E_i and A_i, respectively. $E_i = 1$ means that the user has examined the i-th result in the result sequence; otherwise, the user has not examined the result. $A_i = 1$ means that the result is relevant to the query; otherwise, it is not. A click event occurs if and only if the user checked the result and considers that the result is relevant to the query, i.e., $C_i = 1$. We can use the following formula to represent this process:

$$E_i = 1 \land A_i = 1 \Leftrightarrow C_i = 1$$

Among the results, whether one is relevant to a query is considered to be a factor that is uncorrelated to its position and the other results. Whether the user examines a result is affected by many factors such as the layout of the SERP, which is considered to be a factor uncorrelated to the document itself. Since the two random variables are independent, we can predict the click probability of the result as follows:

$$P(C_i = 1) = P(E_i = 1) \times P(A_i = 1)$$

2.2 Click Models

When examining SERPs, users' click behaviors are understood to have biases. Hence, researchers have proposed click models to obtain unbiased relevance feedback from users' clicks. Most click models follow the examination hypothesis; many different assumptions about the relevant factors of E_i are proposed. Craswell et al. [9] first propose the concept of a click model and used the Cascade Model to represent the user's first click on the SERP. The model addresses the problem of position bias in user behavior to some extent and effectively improves the performance of result ranking. It assumes that the user examines the results from top to bottom until the user clicks on a result or reaches the bottom. Therefore, E_i is 1 for the clicked result and all previous results; otherwise, $E_i = 0$. Subsequently, Guo et al. show that the user may not leave the result page directly after clicking, and it is possible to continue browsing. The probability is related to the position pos of the clicked document which can be represented as:

$$P(E_1 = 1) = 1$$
$$P(E_i = 1|E_{i-1} = 0) = 0$$
$$P(E_i = 1|E_{i-1} = 1, C_{i-1} = 0) = 1$$
$$P(E_i = 1|C_{i-1} = 1) = \lambda_{pos}$$

This model is called Dependent Click Model (DCM) [11] which allows multiple clicks.

Then, Chapelle and Zhang [5] propose Dynamic Bayesian Network Model (DBN) in which the user may decide whether to leave the page according to their satisfaction. In DBN, for each query-document pair, a new random variable S_i is used to indicate whether the user is satisfied after clicking the i-th result. The user may be satisfied only if a result is clicked, i.e.,:

$$P(S_i = 1|C_i = 0) = 0$$
$$P(S_i = 1|C_i = 1) = s_i$$

If the user is satisfied with a result, the browsing ends. Otherwise, the results are further browsed with the probability of gamma, namely:

$$P(E_1) = 1$$
$$P(E_i = 1|S_{i-1} = 1) = 0$$
$$P(E_i = 1|E_{i-1} = 0) = 0$$
$$P(E_i = 1|E_{i-1} = 1, S_{i-1} = 0) = 1$$

The model analyzes the behavior of the user after clicking, which further reduces the position biases.

Dupret and Piwowarski [10] suggest that the user may skip some documents when examining the SERP. The probability of a result being examined is correlated with its position and the position of the last clicked result:

$$P(E_i = 1) = \gamma_{pos,last_{clk}}$$

where pos and $last_{clk}$ are position of the given result and last clicked result. Based on this, the User behavior Model (UBM) is designed.

A body of work is available on the influence of vertical results on users' behavior. Recent studies have found that more and more vertical results appear on SERPs [3], and these vertical results affect the examining behavior of search users [2, 18]. Assumptions made for the organic results cannot be directly applied to the heterogeneous search result pages.

Chen et al. [7] consider vertical results within the click model and propose Federated Click Model (FCM) to address these biases. The model considers that the appearance of vertical results causes the examine probability of results in a page to be biased. The possibility of the user directly clicking the vertical result and finishing the search is greater than clicking organic results. Two binary random variables are introduced, D and A, to successfully model the user's behavior on Federated Web Search. $D = 1$ means that the SERP has an exploration bias, and the user ends the browsing immediately after clicking the vertical result. $A = 1$ indicates that the user's behavior is affected by the vertical result, and the probability of examining the result changes. Later, Wang et al. [22] model the changes brought by the vertical results to the user's inspection order using Vertical-aware Click Model (VCM). The model takes into account that the user has a probability to first examine the vertical results, and then sequentially or reversely examine other results in front of the vertical result. They use the binary variables F and B to model the change in this order. $F = 1$ indicates that the user is attracted by the vertical result and examines the vertical result first. $B = 1$ indicates that the organic result which ranks higher than the vertical result is examined in reverse order (i.e., from the lower rank to the higher rank) after examining the vertical result first. Such modeling approach takes the attraction bias, global bias, first place bias, and sequence bias into account when the user examines a SERP containing vertical results. For the user behavior of SERP with vertical results in Mobile Search, Mao et al. [19] propose MCM (Mobile Click Model). The model considers that the user may obtain valid information directly from the preview displayed in the SERP by the vertical result without a click. Therefore, the model introduces a variable N_i to describe whether or not a type of vertical result needs to be clicked for the user to obtain information, which improves the performance of the model in mobile search scenarios.

However, for models that consider vertical results, FCM and VCM are primarily modeled for situations where the SERP contains exactly one vertical result and do not fit well into existing web search environments. While MCM considers the case where multiple vertical results appear in the SERP, it is mainly developed for mobile searches. The user behaviors of web searches and mobile searches are different. Therefore, it is necessary to model the user behavior of the web search to better extract the behavior of the user in the environment of the existing SERP that contain multiple vertical results in Web search scenarios.

3 EYE-TRACKING EXPERIMENT

To investigate users' behaviors during the search process, we carried out a laboratory study with 30 participants. Each user was asked to complete 20 tasks in the user study and answer corresponding questions.

We asked each user to perform 20 search tasks. Each search task contained a fixed query term (and an explanation of the query term to avoid ambiguity) and the first page of the SERP of the query term given by a commercial search engine. We crawled and saved the search engine's SERP for these 20 query terms, and simulated a network environment for users to ensure that all users saw the same offline version of the SERP.

As users' behaviors may be affected by different types of information needed, we selected queries with different search intents. In our dataset, query-SERP pairs covers "informational" (e.g., "poetry of summer"), "Navigational" (e.g., "Qidianzhongwen website") and "Transactional" (e.g., "Ximalaya (an app)").

In our experiment, we use these hardware devices, software and settings.

- 17-inch LCD monitor.
- Resolution of 1366x768 in pixels.
- Google Chrome browser.
- Head-free eye tracker, Tobii X2-30.
- Tobii Studio.

Through the eye-tracking device, we recorded the eye movement information of each user on the SERPs. Before collecting the eye tracking data of each user, we calibrated the eye tracking device by letting them browse an additional SERP page. The error caused by the eye tracker itself did not exceed one degree in both the horizontal and vertical directions. In the data we collected, there were a small amount of data errors or meaningless data caused by users or software (for example, faults of software or hardware). Among the 600 results given by 30 users to 20 SERPs, there were 518 valid eye movement data results acquired, accounting for 86.3%. We found that among the data, most of eye tracking data (64.1%, 332 of 518) indicated that users did not always browse the entire SERP in order. Therefore, it was necessary to study the non-sequential examination behavior.

In the data recorded by an eye tracking device, there were mainly two types of records: saccades and fixations. Saccades meant that the user's line of sight moved from one point on the screen to another within a period of time. Fixation meant that the user's sight stopped in a small area on the screen for a period of time. In reference to most of the previous work, 200 ms was selected as the lower bound of fixation time in this article. Although some articles mentioned that the relationship between fixation and examination was not very close, it was more convenient and accurate to directly use fixation as a necessary and sufficient condition for examination than to ask users. Therefore, in this article, we directly regarded a fixation that lasted more than 200 ms as an examination.

Equipped with eye-tracking and click data, we sought to answer the following research questions:

- **RQ1**:In the non-sequential revisits during the browsing process, are users more inclined to revisit recent results or not?
- **RQ2**:Does the user's return visit during the browsing process compare adjacent results?
- **RQ3**:During the user's browsing process, is the occurrence of non-sequential examination related to the result location?
- **RQ4**:In a non-sequential visit, does the position bias exist?

By investigating these four questions, our goal was to obtain the regularity of the user's revisits while browsing the page and the click pattern during the revisit. These results were used to model users' behaviors to improve the performance of click models. To formulate these problems more conveniently, we defined the user's examination sequence as: $E =< E_1, E_2 \cdots, E_{m_1} >$. Each item E_i in the sequence E recorded the position of the result of the user's i-th examination. During the browsing process, users examined each item in the sequence chronologically, and the same result was potentially examined multiple times. However, the user's continuous fixation of the same result was regarded as one examination expressed as follows:

$$E_i \neq E_{i-1}, 2 \leq i \leq m_1 \tag{1}$$

Similarly, the click sequence was defined as: $C =< C_1, C_2 \cdots, C_{m2} >$. Each item C_i recorded the position of the result clicked by the user in chronological order. In addition, we defined $Time(E_i)$ to represent the time of the i-th inspection, and $Time(C_i)$ to represent the time of the i-th click. Because we sorted in a time series, this was expressed as follows:

$$\begin{aligned} Time(E_{i-1}) < Time(E_i), 2 \leq i \leq m_1 \\ Time(C_{i-1}) < Time(C_i), 2 \leq i \leq m_2 \end{aligned} \tag{2}$$

We then investigated three research questions.

RQ1:In the non-sequential revisits during the browsing process, are users more inclined to revisit recent results?

Based on the eye-tracking data, we observed an interesting phenomenon. Although many previous works suggested that users revisit results, by observing the screen recordings during the search process, we found that users did not often scroll up across the screens for revisiting when browsing SERPs. In addition, most of the user's revisits occurred on the same screen. Therefore, we determined that users were more likely to revisit results that were closer to the current results, and less scroll up across the screens. In order to verify the users' behavior for this case, we used the "necessary scroll up length" (NSUL) to measure how users scroll up. The NSUL was defined as follows.

$$NSUL = max(\max_{1 \leq i < j \leq m1} E_i - E_j, 0) \tag{3}$$

It should be noted that this indicator is different from the "Revisit Distance". For example, when the test sequence is $E =< 1, 5, 3, 4, 2 >$, there are 2 return visits, $5 - 3$ and $4 - 2$, respectively. The revisit distance for both visits is 2, but the longest return visit distance is $5 - 2 = 3$. Compared to the revisit distance, NSUL can describe the users' scroll-up behaviours more accurate. It shows that the users longest "scroll-up" distance in global instead of in adjective fixations. By definition, the longest revisit distance is 0 for a sequence of complete sequential inspections.

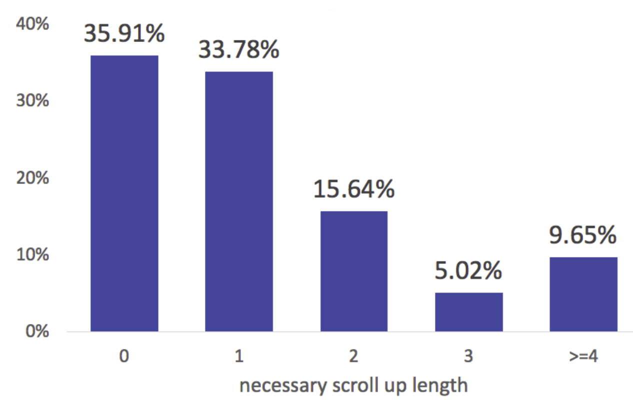

Figure 1: The graph shows the distribution of NSUL of eye-tracking logs. We found that most of eye-tracking logs include revisit. If our model allows revisit with short distances, it can explain 85.3% **eye-tracking logs.**

We computed statistics on the user's "necessary upward scrolling distance", and the results are shown in Figure 1. It can be seen that in more than 64% (332 of 518) of the eye tracking data, users exhibited a revisit behavior. In the revisits, more than half (175 of 332) returned to the nearest results, i.e., had a return distance of 1. The number of results whose return visit distance exceeded four only accounts for less than 10% of the total results. This result confirmed that users were indeed more inclined to revisit with shorter lengths than to revisit in long distances that cross screens. In conclusion, the answer to **RQ1** is "Yes".

RQ2:Do users compare adjacent results during the browsing process?

The short length revisits were reflected in a large amount of eye-tracking data. This motivated us to further investigate exactly what users were doing in non-sequential, short length revisits.

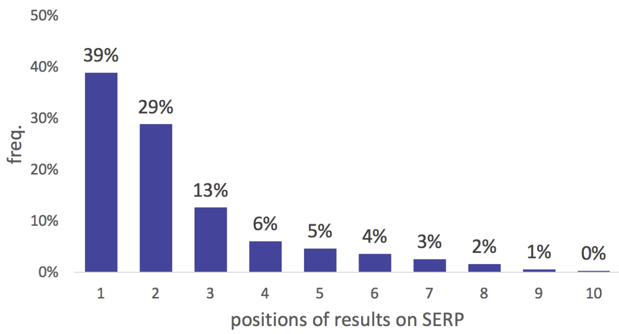

Figure 2: The graph shows how often users fixate on different positions. We found that more than 68% of the fixations occurred on the top two results.

First, we investigated the fixation times of users in different positions, and the results are shown in Figure 2. We found that most fixations occurred in higher ranking places. Generally speaking, after users enter the SERP, they first examined the results with higher rank. Next, we investigated the positions of the first 3 examinations after the SERP is displayed.

1st	2nd	3rd	count	freq
1	2	3	126	24.3%
1	2	1	124	23.9%
1	X	X	119	23.0%
1	2	X	36	6.9%
2	1	2	18	3.5%
1	2	5	14	2.7%
1	2	4	12	2.3%
1	3	2	11	2.1%
1	3	4	9	1.7%
1	2	6	7	1.4%
1	3	X	5	1.0%
1	3	1	5	1.0%
1	4	1	3	0.6%
2	1	3	3	0.6%
2	4	5	3	0.6%
Others			23	4.4%

Table 1: Count and frequency of the first three examinations of combinations. The 'X' means that users leave the page instead of fixating on any results.

We counted the number of times different combinations of the user's first three fixations. In Table 1, we have listed the most common 14 combinations. It can be seen that, in addition to examining the results in order, the most frequent combination for users to examine was 1-2-1. After examining the second result on the SERP, users returned to the first result. In addition, the frequency of the sequence 2-1-2 was also high. These sequences produced quite a lot of close revisits.

Therefore, we hypothesised that a considerable number of users first compare the top two results. As shown in Figure 2, users had fewer examinations on the results with lower rankings, so it was difficult to perform similar statistics on this part of the data. However, we believe that this conclusion can be promoted and applied to the entire SERP browsing process. Therefore, the answer to **RQ2** is 'Yes'.

RQ3:During the user's browsing process, is the occurrence of non-sequential access related to the result location?

This question tries to understand whether users have the same revisit probability when browsing the entire page during the browsing process. In order to study this problem, we computed statistics on the probability of the user's revisits after examining results in different positions. These results are shown in Figure 3. Figure 3

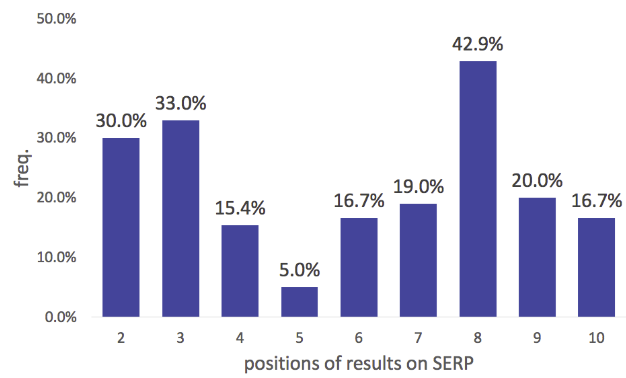

Figure 3: The graph shows how often users revisit previous results after fixating on a different position. Note that the index of the horizontal axis starts at 2 because users cannot revisit after examining the top result. We found that users have a higher probability to revisit a result after examining a high ranking position or one ranked on the 8-th position.

shows that after examining results in different positions, the revisit probability was quite different. After examining the 2nd, 3rd, or 8th result, the revisit probability of revisit was higher. Regarding the user's behavior of revisiting after examining the second and third results, we hypothesise that users are more likely to return to the higher-ranked results out of their experience in using search engines and trust in the higher-ranked search results. When the 8th result was examined, it was usually close to the bottom of the SERP. For this position, users were more inclined to revisit the results that were visited before (because there are no further results to consider). For the other positions, the user was more likely to be in the process of scrolling down, and the possibility of a revisit was lower. During the user's inspection process, the position where the non-sequential access occurred was related to the position of the result. When designing the model, position-related parameters wwere needed to characterize the user's revisit behavior. In conclusion, the answer to **RQ3** is 'Yes'.

RQ4:In non-sequential visit, does the position bias exist?

Window Size	1st	2nd	3rd
2	72.7%	38.7%	-
3	74.8%	40.7%	25.2%

Table 2: The clicked frequent of results on different position of windows when window size is 2 or 3. The 1st, 2nd, 3rd means the result's relative position in the window. The sum of clicked frequent on different positions is more than 100%, because users may click more than one result in the window.

Many click models proposed in the literature verified the existence of position bias. When users browsed the entire SERP, they were more inclined to click on the top ranked results. This research question explores whether these prior results are also applicable to the proposed CBCM.

To investigate this issue, we conducted a study on the clicks of users during "scrolling up" during search. We defined the process of "scrolling up" if and only if the user's most recent exam was the result before the return visit.

For the log of $NSUL \leq 3$, we counted the clicks during the scrolling up process. Considering that the user verification process generally occurs from up to down with comparisons on a small scale, we assumed that there was a "window" for the user. The "window" contained two or three consecutive results that the user was comparing, and the user had the opportunity to click on them. The position of the "window" was determined by the one with the lowest ranking among the results of the user's previous inspection. For example, if the current inspection sequence was $E =< 1, 5, 3 >$ and the window's size was three, the window was at $[3, 5]$. If the window's size was two, this log was discarded (from **RQ2**, the window size less than two covered more than 69% logs, and less than three covered 85%. Therefore, discarding log data beyond a small window size had a limited impact).

As shown in Table 2, we found that users were more inclined to click on the results that were more advanced in the scrolling process of length two or three. This revealed the position bias still existed and required consideration in the next step of the research. Therefore, the answer to **RQ4** is true.

After analysing the logs of the eye-tracking experiment, we found that the four research questions are all true. We use these conclusions to design Click Model in Sec. 4.

4 COMPARISON-BASED CLICK MODEL

This section formally introduces our click model: comparison-based click model (CBCM). As discussed in the previous section, we determine that existing click models do not fully describe user behavior. This is mainly due to the presence of many non-sequential visits in user behavior. Moreover, existing non-sequence click models cannot fully describe these non-sequential revisits. The main problem with existing click models is that it does not take into account the user's behavior of comparing results with adjacent locations during the top-down inspection process. Our model aims to describe this type of user behaviors.

Different from existing models, our model is based on the assumption that users consider multiple results at the same time. So, we redefine the three variables of user examination, click, and satisfaction. Because the existence of comparison behavior is considered, the correlation parameter R_{pos} is also adjusted. We use R_{pos} to indicate the correlation between the result at the pos position and the query term. In particular, the value range of R_{pos} is any real number.

4.1 Hypotheses of CBCM

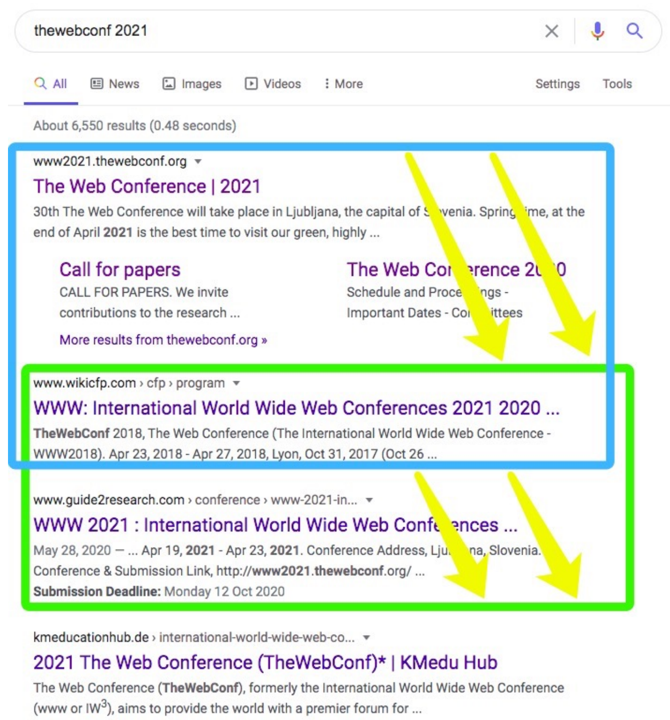

Figure 4: A typical SERP page and WoE(Window of Examination). The first screen presents three results after the user searches for the keyword "thewebconf 2021". We hypothesise that users examine them in a "weak order". The user's WoE stays at the top-2 results at first and recognises these two results at the same time. Then, the user's WoE moves down, recognising the 2nd and 3rd results.

We define examination hypothesis proposed above as the "weak order hypothesis". From the global perspective, users examine the results from top to bottom. From the local perspective, users compare two or three results simultaneously instead of judging only one result. We called these two or three results a Window of Examination (WoE). Here, we take the user's comparison of two results at the same time as an example. The situation where the user considers three results at the same time is similar. After obtaining the SERP, the user first recognizes the first and the second results. After checking the top two results, the user may or may not click. We model the basis of whether the user clicks or not in the next paragraph. Then, the user performs a "move down" operation. The meaning of moving down is as follows. The user removes the top result (i.e., the 1st result in the WoE) in the current consideration range (1-2) from the recognizing range. This removes the top result out of the consideration range; it is moved to the recognizing range. The user's inspection process continues to move down according to this rule until a) the user clicks on a satisfactory result to end the current query, or b) all of the results in the entire SERP are recognized and the current query ends. As shown in the Fig. 4, this process can be vividly described as "WoE moving down". When considering two results in a WoE, users are faced with three different choices: click on the first result in the WoE, click on the second result in the WoE, or move down the WoE.

4.2 Click-based Click Model

We can formalize the results on the search process discussed in the previous subsection as shown in Figure 5. Users have a "window" containing two or three results. When the user examines the SERP, they choose two every time:

- Click one result in the current window
- Move down the window

When the window reaches the end of the SERP or the user feels satisfied with any results, the searching ends.

We define the clicks vector $\vec{c}$ and operation of it as follows:

- $\vec{c}$ denotes the clicks of the results in the WoE. (e.g., if the 1st and the 3rd results have been clicked and the 2nd has not been clicked, $\vec{c} = [1, 0, 1]$).
- The index of $\vec{c}$ starts at 0. $\vec{c}_p$ means the p-th variable in $\vec{c}$.
- $\vec{c}^{+1}$ denotes the new click vector in the WoE after moving down.(e.g., the size of the WoE is three and the 1st and the 2rd results are clicked: $[1, 0, 1]^{+1} = [0, 1, 0]$).
- $\vec{c} + p$ denotes the new click vector in the WoE after the p-th result is clicked in the WoE: $i+p$-th global (e.g., $[0, 0, 1] + 0 = [1, 0, 1]$).

Since users consider all of the results in a WoE at the same time, we redefine a series of random variables. In the following equations, we define $E_{i,k,\vec{c}}$ as:

- WoE begins at i result and ends at $i + b$-th result (b is two for example)
- k number of clicks happen since the user started examining the SERP
- $\vec{c}$ describe clicks in the WoE

We use $C_{i,k,\vec{c},p}$ to describe that besides conditions depicted by $E_{i,k,\vec{c}}$ being satisfied, and the user also clicks $i + p$-th result in the WoE. From the definition, we have:

$$E_{i,k,\vec{c}} = 0 \Rightarrow C_{i,k,\vec{c},p} = 0 \text{ }, \forall 1 \leq i, i + b \leq N, 0 \leq p \leq 2 \qquad (4)$$

Next, we describe how users make choices in a WoE.

Here, we use the softmax method to describe the user's click probability and the probability of moving down. That is, we believe that the probability of a user clicking on a result is related to the relevance level of this result, which can be expressed as:

$$P(C_{i,k,\vec{c},p} = 1|E_{i,k,\vec{c}} = 1) \propto exp(R_{i+p})(0 \leq p \leq b) \qquad (5)$$

In addition, we noticed that when considering the results in the same window, users are more inclined to click on the previous results. If there is a result that has already been clicked in the window, the user's willingness to click on the result decreases. Therefore, we introduce the parameter $\gamma_{i,p}$, which represents the bias that the user clicks on the $i+p$ result when the window is from the i-th result to the $i+b$-th result. The parameter $\theta_{i,p}$ refers to the user's click bias of the $i+p$-th result when the window is located from the i-th result to the $i+b$-th result and the $i+p$-th result has been clicked before.

We add these two parameters into the formula and obtain:

$$\begin{aligned} &P(C_{i,k,\vec{c},p} = 1|E_{i,k,\vec{c}} = 1) \\ &\propto exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c}_i = 1) \times \theta_{i,p})(0 \leq p \leq b) \end{aligned} \qquad (6)$$

We also need to take the probability of moving down the positions of the results into account. We realize that users prefer to click on high ranked results; they also prefer to jump and move down to the low ranked results. When the window is in the same position, the user's click tendency is also related to the number of

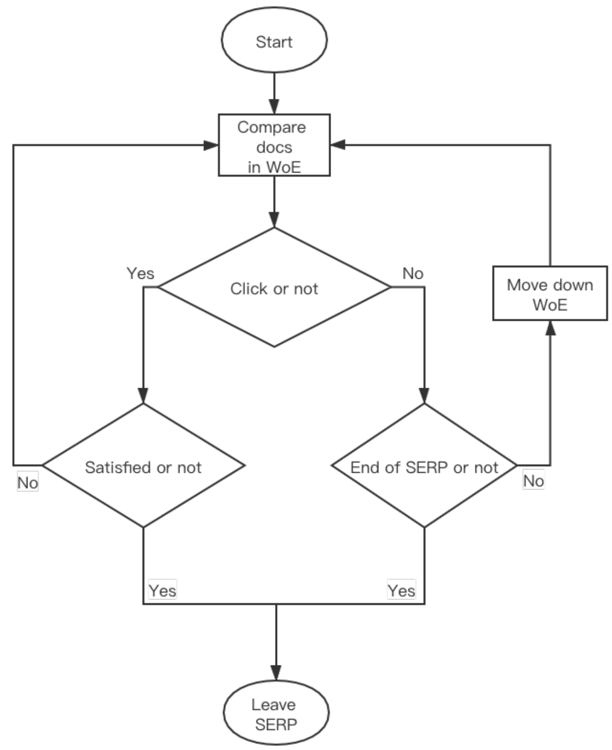

Figure 5: The process graph describe users' behaviours from making a query to leaving a SERP. After a user enters the SERP, the user sets the WoE at the top two or top three results. Each time, a user may choose to click one result in the WoE or not. If a user clicked one result and is satisfied with it, then they leave the page. Otherwise, the user returns back to the WoE in the SERP and continues to search. If the user chooses not to click any results in the WoE, they move down the WoE unless the WoE is already at the end of the SERP. In this case, the user leaves the SERP.

results that have been clicked before. Therefore, we introduce the parameter $g_{i,k}$ to indicate the user's tendency to move down when these two conditions are met:

- WoE is located from i to $i + b$
- k number of clicks happen since the user starts examining the SERP

This parameter is related to the probability of the user moving down the ranked results. We describe the probability of moving down by the formula below:

$$P(E_{i+1,k,\vec{c}'} = 1 | E_{i,k,\vec{c}} = 1) \propto exp(g_{i,k})(1 \le i, i + b + 1 \le N) \tag{7}$$

Based on the above equations, the following formula can be used to calculate the probability of each choice in a WoE.

$$Z_{i,k,\vec{c}} = exp(g_{i,k}) + \sum_{p=0}^{b} exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c}_p = 1)\theta_{i,p}) \tag{8}$$

$$P(C_{i,k,\mathbf{c},p_1} = 1 | E_{i,k,\mathbf{c}} = 1) = \frac{exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c}_p)\theta_{i,p})}{Z_{i,k,\vec{c}}} \quad (1 \le i, i + b + 1 \le N, 0 \le p \le b) \tag{9}$$

$$P(E_{i+1,k,\vec{c}^{+1}} = 1 | E_{i,k,\vec{c}} = 1) = \frac{exp(g_{i,k})}{Z_{i,k,\vec{c}}} \tag{10}$$

After the user clicks on a result, if they are satisfied with the clicked result, they end the browsing and leave the SERP. Otherwise, the user returns to the previous WoE and continues to browse the SERP. We assume that the probability the user is satisfied is only related to the relevance level of search results which can be formulated as:

$$P(E_{i+1,k+1,\vec{c}'} = 1 | C_{i,k,\vec{c},p} = 1) = 1 - s_{i+p} \tag{11}$$

where $\vec{c}'$ may be a different click vector because one result in a WoE may be changed from non-clicked to clicked, and s_{i+p} is the probability the user is satisfied with the $i + p$-th result after a click.

This completes the introduction of the entire model. The user follows this process to select and browse the results.

4.3 Train

In this subsection, we introduce the training method of the model. We use the SGD method to train the model and fit the parameters of the model in the training set. The goal of fitting is to maximize the Log-Likelihood (LL) of click predictions.

In CBCM, one result has multiple click opportunities when the WoE is located in different positions. For example, if the WoE size is 3, the 4th result may be clicked when the WoE is located at 2..4,3..5 or 4..6. When calculating the LL, it is necessary to fully consider that when the WoE is in different positions, when a result is clicked in at least one position, it is considered as clicked.

Then, we introduce the method of calculating the LL. Let the forward probability $F[i, k, \vec{c}]$ represent the probability that the following conditions are met at the same time during the user's browsing process:

- The WoE is located from i to $i + b$.
- The click prediction of 1st to $(i-1)$-th documents are completely correct.
- k is the number of clicks that happened since the user started examining the SERP.
- $\vec{c}$ describes the clicks in the WoE.

In particular, when browsing starts, the user's window is at the top of the page and no result has been clicked. So, we have:

$$F[1, 0, \vec{0}] = 1 \tag{12}$$

According to CBCM's assumptions, if the user meets the above conditions while browsing, the user has two options for the next operation: click one result in the WoE or move down. The probability is given by equations 9 and 10. If the current state is not the initial state in the browsing process ($F[1, 0, \vec{0}]$), the way to calculate F is given by the equation 20 in the appendix.

After calculating the forward probability F, we use a similar approach to calculate the backward probability B. Let the backward probability $B[i, k, \vec{c}]$ represent the probability that all of our model's click predictions are correct when all four conditions of F mentioned above are met at the same time during the user's browsing process.

We treat the end of the examination as a special "move down": when a user wants to "move down" at the end of the SERP, they leave the SERP. When $i = N - b + 1$, the examination of the SERP is ended with no more clicks. So, for any k or $\vec{c}$ we have:

$$B[N - b + 1, k, \vec{c}] = \prod_{p=0}^{b-1} \boldsymbol{I}(\vec{c}_p = C_{N-b+1+p}) \tag{13}$$

C_i with only one subscript describes if the user clicked the i-th result in the log data.

We can calculate B by the Equation 21 in the appendix.

Then, we define the LL as:

$$LL(\Theta) = B[1, 0, \vec{0}] \tag{14}$$

Now, we present the approach to calculate the LL of a SERP. Next, we introduce the derivation of the parameters for the SGD.

In CBCM, we have five types of parameters:

- R_{id}, relevance between document id and query word.
- $g_{i,k}$, the tendency to move down when the WoE at i and clicked k times previous.
- $\gamma_{i,p}$, the position bias of results in the WoE.
- $\theta_{i,p}$, the clicked bias of results in the WoE.
- s_{id}, the probability that user leaves the page (i.e., satisfied with the result) after clicking into a result.

For any parameter in these five types, we can derive them with a similar approach. This approach is presented in the appendix. Once derived, we can use the SGD to optimize the parameters of CBCM.

5 EXPERIMENTS

In this section, we compare CBCM with the mainstream click model, i.e., DCM [12], DBN [5], MCM [19], UBM [10], VCM [22] and FCM [6]. We aim to answer the following research question by experiments conducted on a large-scale search log dataset collected from a commercial search engine in China:

- Does CBCM perform better than other models in terms of click predictions?

- Does CBCM perform better than other models in terms of relevance estimation?

5.1 Dataset

The dataset is randomly sampled from a search log in September 2017. We collected search log data for 7 days. Then we discard queries with fewer than 10 search sessions to make sure CBCM and baseline models can capture enough information when training and test enough when testing. We also discard queries with more than 10,000 search sessions to prevent a small number of queries from dominating the data. The details about the dataset can be found in Table 3. We split the whole dataset into training, validate and test sets at a ratio of 7:1:2 by session start time.

Table 3: Description of experiment dataset ("#" refers to "number of").

	#Distinct queries.	#Sessions	#Documents.
Training	942,579	43,058,176	15,188,816
Validate	874,734	5,707,306	220,103
Test	942,579	12,904,515	11,464,097

5.2 Click prediction

As in previous work, we select average perplexity and the LL to evaluate the performance of click predictions. Perplexity is defined as:

$$Perp. = 2^{-\frac{1}{|S|}\sum_{s\in S} C_i^s log_2(q_i^s)+(1-C_i)^s log_2(1-q_i^s)} \tag{15}$$

When calculating the performance improvement of the model based on perplexity, the value of the test with a prediction accuracy of 100% is 1 and the improvement between $p2$ and $p1$ can be expressed as:

$$(p2 - p1)/(p2 - 1) \tag{16}$$

When calculating the LL promotion, we use the following formula to describe the elevation of LL_1 relative to LL_2:

$$e^{LL_1 - LL_2} - 1 \tag{17}$$

Table 4: The performance of click predictions measured in average perplexity and Log-Likelihood(LL)

Model	Perp.	CBCM impr.	LL.	CBCM impr.
CBCM	**1.2031**	-	**-1.7634**	-
DCM	1.2293	11.43%	-1.9678	22.63%
DBN	1.2237	9.22%	-1.9325	18.53%
MCM	1.2191	7.29%	-1.8822	13.43%
UBM	1.2228	8.84%	-1.9150	16.30%
VCM	1.2209	8.06%	-1.9027	18.41%
FCM	1.2201	7.72%	-1.8917	5.53%

We tested the results of each model on the dataset. As shown by Table4, CBCM performs better on click predictions than other models and computing the LL promotion.

5.3 Relevance estimation

An important goal of the proposed CBCM is to improve the search ranking of web search engines. To construct the golden standard, we recruited three annotators to provide relevance judgments for each query-document pair. The relevance judgments were gathered on the 4-point scale: irrelevant(1), somewhat relevant(2), fairly relevant(3), highly relevant(4). To measure the effectiveness of a click model on relevance estimation, the often used metrics include nDCG(normalized Discounted Cumulative Gain) and MRR(Mean reciprocal rank).

For nDCG, we use the following formula to calculate the value of DCG@k score.

$$DCG@k = \sum_{i=1}^{k} \frac{2^{r(i)}}{log(i+1)} \tag{18}$$

where $r(i)$ is the relevance score at position i and k is the depth of this ranked list of documents. Then the nDCG@k can be obtained by normalizing DCG@k using ideal DCG@k which measures the perfect ranking.

The nDCG results obtained are shown in Table 5. We can observe from Table 5 that CBCM achieves the best performance in terms of nDCG@1 and nDCG@3 while the performance of CBCM decreases when the cut-off of nDCG becomes larger (@5). The reason for this may be two-fold:

- Users may conduct more comparison behaviors on results with the higher ranks since they tend to believe that search engines will rank results with the higher relevance scores at the higher ranks. Our eye-tracking experiments showed that users compare results more frequently in the top position.
- Compared with VCM, we do not incorporate additional vertical result type information into the model optimization.

In the future, we plan to incorporate the types of vertical results into our model and try to further improve the performance of our model.

For MRR, the relevance score of a given query-document pair should be mapped to a binary scale. In this paper, we consider results with the score 4(highly relevant) or 3 (fairly relevant) to be relevant, and the score 2(somewhat relevant) and 1 (irrelevant) are considered to be irrelevant. We calculated MRR using the following formula:

$$MRR = \frac{1}{|Q|}\sum_{q}\frac{1}{rank_q} \tag{19}$$

Table 5: Relevance estimation performance in terms of nDCG@1, 3, 5 and MRR

Model	nDCG@1	nDCG@3	nDCG@5	MRR
CBCM	**0.6445**	**0.6814**	0.7357	**0.9851**
DCM	0.6320	0.6730	0.7173	0.9785
DBN	0.6103	0.6637	0.7233	0.9629
MCM	0.6054	0.6680	0.7340	0.9653
UBM	0.6246	0.6751	0.7383	0.9703
VCM	0.5981	0.6749	**0.7386**	0.9802

where $rank_i$ refers to the rank position of the first relevant document for the query q. The MRR results are shown in Table 5. We can see that the proposed CBCM achieves the best performance in terms of MRR.

In summary, in terms of relevance estimation, our model CBCM achieves better performance compared to the state-of-the-art click models.

6 CONCLUSION AND FUTURE WORK

In this paper, we conduct a lab-based eye-tracking study to thoroughly investigate fine-grained examination behavior of search users. Through eye-tracking data, we find that users perform a comparison strategy to examine search results. Although users examine search results following a top to bottom pattern, they occasionally compare adjacent results, especially results at the top position. Based on these observations, we propose a hypothesis called "weak sequence" and design a novel click model named CBCM which takes the context of a result (e.g., adjacent results) into consideration. Through extensive experimentation, we demonstrate that in terms of behavior prediction and result ranking, CBCM significantly outperforms the state-of-the-art baseline models. In the future, we plan to further explore the factors that affect users' comparative behavior and propose more sophisticated models.

7 CODE AND DATA

To facilitate reproducibility of our results, we share the code and data at https://github.com/ufozgg/DJClickModels.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China (2018YFC0831700), Natural Science Foundation of China (Grant No. 61732008, 61532011, 61902209), Beijing Outstanding Young Scientist Program (NO. BJJWZYJH012019100020098), Beijing Academy of Artificial Intelligence (BAAI) and Tsinghua University Guoqiang Research Institute. This research is partly supported by the Tsinghua-Sogou Tiangong Institute for Intelligent Computing.

REFERENCES

[1] Eugene Agichtein, Eric Brill, and Susan Dumais. 2006. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 19–26.

[2] Jaime Arguello and Robert Capra. 2014. The effects of vertical rank and border on aggregated search coherence and search behavior. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 539–548.

[3] Jaime Arguello, Fernando Diaz, Jamie Callan, and Jean-Francois Crespo. 2009. Sources of evidence for vertical selection. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 315–322.

[4] Huanhuan Cao, Daxin Jiang, Jian Pei, Qi He, Zhen Liao, Enhong Chen, and Hang Li. 2008. Context-aware query suggestion by mining click-through and session data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 875–883.

[5] Olivier Chapelle and Ya Zhang. 2009. A dynamic bayesian network click model for web search ranking. In *Proceedings of the 18th international conference on World wide web*. ACM, 1–10.

[6] Danqi Chen, Weizhu Chen, Haixun Wang, Zheng Chen, and Qiang Yang. 2012. Beyond ten blue links: enabling user click modeling in federated web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. 463–472.

[7] Danqi Chen, Weizhu Chen, Haixun Wang, Zheng Chen, and Qiang Yang. 2012. Beyond ten blue links: enabling user click modeling in federated web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 463–472.

[8] Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 7, 3 (2015), 1–115.

[9] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the 2008 international conference on web search and data mining*. ACM, 87–94.

[10] Georges E Dupret and Benjamin Piwowarski. 2008. A user browsing model to predict search engine click data from past observations.. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 331–338.

[11] Fan Guo, Chao Liu, and Yi Min Wang. 2009. Efficient multiple-click models in web search. 124–131. https://doi.org/10.1145/1498759.1498818

[12] Fan Guo, Chao Liu, and Yi Min Wang. 2009. Efficient multiple-click models in web search. In *Proceedings of the second acm international conference on web search and data mining*. 124–131.

[13] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.* 20, 4 (Oct. 2002), 422–446. https://doi.org/10.1145/582415.582418

[14] Jyun-Yu Jiang, Yen-Yu Ke, Pao-Yu Chien, and Pu-Jen Cheng. 2014. Learning User Reformulation Behavior for Query Auto-Completion. In *Proceedings of the 37th International ACM SIGIR Conference on Research amp; Development in Information Retrieval* (Gold Coast, Queensland, Australia) *(SIGIR '14)*. Association for Computing Machinery, New York, NY, USA, 445–454. https://doi.org/10.1145/2600428.2609614

[15] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2017. Accurately interpreting clickthrough data as implicit feedback. In *ACM SIGIR Forum*, Vol. 51. Acm New York, NY, USA, 4–11.

[16] Thorsten Joachims, Laura A Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately interpreting clickthrough data as implicit feedback. In *Sigir*, Vol. 5. 154–161.

[17] Yanen Li, Anlei Dong, Hongning Wang, Hongbo Deng, Yi Chang, and ChengXiang Zhai. 2014. A two-dimensional click model for query auto-completion. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 455–464.

[18] Zeyang Liu, Yiqun Liu, Ke Zhou, Min Zhang, and Shaoping Ma. 2015. Influence of vertical result in web search examination. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 193–202.

[19] Jiaxin Mao, Cheng Luo, Zhang Min, and Shaoping Ma. 2018. Constructing Click Models for Mobile Search. 775–784. https://doi.org/10.1145/3209978.3210060

[20] Neil O'Hare, Paloma de Juan, Rossano Schifanella, Yunlong He, Dawei Yin, and Yi Chang. 2016. Leveraging User Interaction Signals for Web Image Search. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pisa, Italy) *(SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 559–568. https://doi.org/10.1145/2911451.2911532

[21] Matthew Richardson, Ewa Dominowska, and Robert Ragno. 2007. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*. ACM, 521–530.

[22] Chao Wang, Yiqun Liu, Min Zhang, Shaoping Ma, Meihong Zheng, Jing Qian, and Kuo Zhang. 2013. Incorporating vertical results into search click models. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 503–512.

[23] Yue Wang, Dawei Yin, Luo Jie, Pengyuan Wang, Makoto Yamada, Yi Chang, and Qiaozhu Mei. 2016. Beyond Ranking: Optimizing Whole-Page Presentation. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining* (San Francisco, California, USA) *(WSDM '16)*. Association for Computing Machinery, New York, NY, USA, 103–112. https://doi.org/10.1145/2835776.2835824

[24] Wei Wu, Hang Li, and Jun Xu. 2013. Learning Query and Document Similarities from Click-through Bipartite Graph with Metadata. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* (Rome, Italy) *(WSDM '13)*. Association for Computing Machinery, New York, NY, USA, 687–696. https://doi.org/10.1145/2433396.2433481

[25] Danqing Xu, Yiqun Liu, Min Zhang, Shaoping Ma, and Liyun Ru. 2012. Incorporating Revisiting Behaviors into Click Models. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining* (Seattle, Washington, USA) *(WSDM '12)*. Association for Computing Machinery, New York, NY, USA, 303–312. https://doi.org/10.1145/2124295.2124334

A METHOD OF CALCULATE PROBABILITY

$$
\begin{aligned}
&F[i,k,\vec{c}](1 \le k, 2 \le i) \\
&= \sum_{\vec{c_0}} \boldsymbol{I}(\vec{c_0}^{+1} = \vec{c} \wedge \vec{c_0}_0 = C_{i-1}) \cdot F[i-1,k,\vec{c_0}] \cdot P(E_{i,k,\vec{c}} = 1 | E_{i-1,k,\vec{c_0}} = 1) \\
&\quad + \sum_{\vec{c_0}} \boldsymbol{I}(\vec{c_0} + p = \vec{c}) \cdot F[i,k-1,\vec{c_0}] \cdot P(E_{i,k,\vec{c}} = 1 | E_{i,k-1,\vec{c_0}} = 1) \\
&= \sum_{\vec{c_0}} \boldsymbol{I}(\vec{c_0}^{+1} = \vec{c} \wedge \vec{c_0}_0 = C_{i-1}) \cdot F[i-1,k,\vec{c_0}] \cdot P(E_{i,k,\vec{c}} = 1 | E_{i-1,k,\vec{c_0}} = 1) \\
&\quad + \sum_{\vec{c_0}} \sum_{p=0}^{b} \boldsymbol{I}(\vec{c_0} + p = \vec{c}) \cdot F[i,k-1,\vec{c_0}] \cdot P(C_{i,k-1,\vec{c_0},p} = 1 | E_{i,k-1,\vec{c_0}} = 1) \cdot P(E_{i,k,\vec{c}} = 1 | C_{i,k-1,\vec{c_0},p} = 1) \\
&= \sum_{\vec{c_0}} \boldsymbol{I}(\vec{c_0}^{+1} = \vec{c} \wedge \vec{c_0}_0 = C_{i-1}) \cdot F[i-1,k,\vec{c_0}] \cdot \frac{exp(g_{i-1,k})}{Z_{i-1,k,\vec{c_0}}} \\
&\quad + \sum_{\vec{c_0}} \sum_{p=0}^{b} \boldsymbol{I}(\vec{c_0} + p = \vec{c}) \cdot F[i,k-1,\vec{c_0}] \cdot \frac{exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c_0}_p = 1)\theta_{i,p})}{Z_{i,k-1,\vec{c_0}}} \cdot (1 - s_{i+p})
\end{aligned} \tag{20}
$$

$$
W_p = \begin{cases} 1(C_i = 0, \forall i \ge p) \\ 0(C_i = 1, \exists i \ge p) \end{cases}
$$

$$
\begin{aligned}
&B[i,k,\vec{c}](0 \le k, 1 \le i, i+b \le N) \\
&= \sum_{\vec{c_1}} \boldsymbol{I}(\vec{c}^{+1} = \vec{c_1} \wedge \vec{c}_0 = C_i) \cdot B[i+1,k,\vec{c_1}] \cdot P(E_{i+1,k,\vec{c_1}} = 1 | E_{i,k,\vec{c}} = 1) \\
&\quad + \sum_{\vec{c_1}} \sum_{p=0}^{b} \boldsymbol{I}(\vec{c} + p = \vec{c_1}) \cdot B[i,k+1,\vec{c_1}] \cdot P(C_{i,k,\vec{c},p} = 1 | E_{i,k,\vec{c}} = 1) \cdot P(E_{i,k+1,\vec{c_1}} = 1 | C_{i,k,\vec{c},p} = 1) \\
&\quad + \sum_{\vec{c_1}} \sum_{p=0}^{b} P(C_{i,k,\vec{c},p} = 1 | E_{i,k,\vec{c}} = 1) \cdot P(E_{i,k+1,\vec{c_1}} = 0 | C_{i,k,\vec{c},p} = 1) \cdot \boldsymbol{I}(\vec{c}_p = C_{i+p} \forall 0 \le p \le b \wedge W_{i+b+1}) \\
&= \sum_{\vec{c_1}} \boldsymbol{I}(\vec{c}^{+1} = \vec{c_1} \wedge \vec{c}_0 = C_i) \cdot B[i+1,k,\vec{c_1}] \cdot \frac{exp(g_{i,k})}{Z_{i,k,\vec{c}}} \\
&\quad + \sum_{\vec{c_1}} \sum_{p=0}^{b} \boldsymbol{I}(\vec{c} + p = \vec{c_1}) \cdot B[i,k+1,\vec{c_1}] \cdot \frac{exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c}_p = 1)\theta_{i,p})}{Z_{i,k,\vec{c}}} \cdot (1 - s_{i+p}) \\
&\quad + \sum_{\vec{c_1}} \sum_{p=0}^{b} \frac{exp(R_{i+p} + \gamma_{i,p} + \boldsymbol{I}(\vec{c}_p = 1)\theta_{i,p})}{Z_{i,k,\vec{c}}} \cdot s_{i+p} \cdot \boldsymbol{I}(\vec{c}_p = C_{i+p} \forall 0 \le p \le b \wedge W_{i+b+1})
\end{aligned} \tag{21}
$$

B METHOD OF CALCULATE DERIVATIVE

First, for calculate the derivative easily, we define auxiliary variable $G[i,k,\vec{c},p]$ and $U_{i,k,\vec{c}}$.

$G[i,k,\vec{c},p]$ is the probability that all predictions of documents in SERP are right after user clicked the $i+p$-th result when these conditions met:

- WoE is located from i to $i+b$.
- The click prediction of 1st to $(i-1)$-th documents are completely correct.
- k results have been clicked before.
- $\vec{c}$ describe clicks in WoE.

We can calculate G as follows:

$$
G[i,k,\vec{c},p] = \boldsymbol{I}(\vec{c}_p = C_{i+p} \forall 0 \le p \le b \wedge W_{i+b+1}) \cdot s_{i+p} + B[i,k,\vec{c}] \cdot (1 - s_{i+p}) \tag{22}
$$

And $U_{i,k,\vec{c}}$ is the sum of probability that all predictions of documents in SERP are right when conditions privious met.

We can calculate U as follows:

$$
U_{i,k,\vec{c}} = \frac{g_{i,k}}{Z_{i,k,\vec{c}}} \cdot B[i+1,k,\vec{c}^{+1}] + \sum_{p=0}^{b} \frac{exp(R_{i+p} + \gamma_{i,y} + \boldsymbol{I}(\vec{c}_y = 1)\theta_{i,y})}{Z_{i,k,\vec{c}}} \cdot G[i,k,\vec{c},p] \tag{23}
$$

B.1 Calculation of the derivative of R_{id}: $\frac{\partial LL(R_{id})}{\partial R_{id}}$

P_{tot} is the likelihood, i.e. the probability that click sequence appears.

$D_{i+y} = id$ means that the $i+y$-th document in SERP is the id-th document globally.

$$\frac{\partial LL(R_{id})}{\partial R_{id}} \tag{24}$$
$$= \frac{\sum\limits_{i,k,\vec{c},x,y|D_{i+y}=id} F[i,k,\vec{c}] \cdot (\boldsymbol{I}(x=y)\frac{exp(R_{id}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})\cdot G[i,k+1,\vec{c}+x,x]}{U_{i,k,\vec{c}}} - \frac{exp(R_{i+x}+\gamma_{i,x}+\boldsymbol{I}(\vec{c}_x=1)\theta_{i,x})\cdot exp(R_{id}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})\cdot G[i,k+1,\vec{c}+y,y]}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$
$$- \frac{\sum\limits_{i,k,\vec{c},y|D_{i+y}=id} F[i,k,\vec{c}] \cdot B[i+1,k,\vec{c}^{+1}](\frac{exp(g_{i,k})\cdot exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$

B.2 Calculation of the derivative of $\gamma_{i,y}$: $\frac{\partial LL(\gamma_{i,y})}{\partial \gamma_{i,y}}$

P_{tot} and D_{i+y} are defined in previous subsection.

$$\frac{\partial LL(\gamma_{i,y})}{\partial \gamma_{i,y}} \tag{25}$$
$$= \frac{\sum\limits_{i,k,\vec{c},x,y|D_{i+y}=id} F[i,k,\vec{c}] \cdot (\boldsymbol{I}(x=y)\frac{exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})\cdot G[i,k+1,\vec{c}+x,x]}{U_{i,k,\vec{c}}} - \frac{exp(R_{i+x}+\gamma_{i,x}+\boldsymbol{I}(\vec{c}_x=1)\theta_{i,x})\cdot exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})\cdot G[i,k+1,\vec{c}+y,y]}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$
$$- \frac{\sum\limits_{i,k,\vec{c},y} F[i,k,\vec{c}] \cdot B[i+1,k,\vec{c}^{+1}](\frac{exp(g_{i,k})\cdot exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$

B.3 Calculation of the derivative of $\theta_{i,y}$: $\frac{\partial LL(\theta_{i,y})}{\partial \theta i,y}$

P_{tot} is defined in previous subsection.

$$\frac{\partial LL(\theta_{i,y})}{\partial \theta_{i,y}} \tag{26}$$
$$= \frac{\sum\limits_{i,k,\vec{c},x,y|D_{i+y}=id} F[i,k,\vec{c}] \cdot \boldsymbol{I}(\vec{c}_y=1)(\boldsymbol{I}(x=y)\frac{exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})\cdot G[i,k+1,\vec{c}+x,x]}{U_{i,k,\vec{c}}} - \frac{exp(R_{i+x}+\gamma_{i,x}+\boldsymbol{I}(\vec{c}_x=1)\theta_{i,x})\cdot exp(R_{i+y}+\gamma_{i,y}+\theta_{i,y})\cdot G[i,k+1,\vec{c}+y,y]}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$
$$- \boldsymbol{I}(\vec{c}_y=1)\frac{\sum\limits_{i,k,\vec{c},y} F[i,k,\vec{c}] \cdot B[i+1,k,\vec{c}^{+1}](\frac{exp(g_{i,k})\cdot exp(R_{id}+\gamma_{i,y}+\theta_{i,y})}{U^2_{i,k,\vec{c}}})}{P_{tot}}$$

B.4 Calculation of the derivative of $g_{i,k}$: $\frac{\partial LL(g_{i,k})}{\partial g_{i,k}}$

P_{tot} is defined in previous subsection.

$$\frac{\partial LL(g_{i,k})}{\partial g_{i,k}} = \frac{\sum\limits_{i,k,\vec{c}} F[i,k,\vec{c}] \cdot B[i+1,k,\vec{c}^{+1}](\frac{exp(g_{i,k})}{U_{i,k,\vec{c}}})}{P_{tot}} - \frac{\sum\limits_{i,k,\vec{c},x} F[i,k,\vec{c}] \cdot G[i,k+1,\vec{c}+x,x](\frac{exp(R_{i+x}+\gamma_{i,x}+\boldsymbol{I}(\vec{c}_x=1)\theta_{i,x})\cdot exp(g_{i,k})}{U^2_{i,k,\vec{c}}})}{P_{tot}} \tag{27}$$

B.5 Calculation of the derivative of s_{id}: $\frac{\partial LL(s_{id})}{\partial s_{id}}$

P_{tot} and D_{i+y} are defined in previous subsection.

$$\frac{\partial LL(s_{id})}{\partial s_{id}} \tag{28}$$
$$= \frac{\sum\limits_{i,k,\vec{c},y|D_{i+y}=id} F[i,k,\vec{c}] \cdot \frac{exp(R_{i+y}+\gamma_{i,y}+\boldsymbol{I}(\vec{c}_y=1)\theta_{i,y})}{U_{i,k,\vec{c}}} \cdot (\boldsymbol{I}((\vec{c}+y)_p = C_{i+p}\forall 0 \le p \le b \wedge W_{i+b+1}) - B[i+1,k,\vec{c}+y])}{P_{tot}}$$