Available online at www.sciencedirect.com

ScienceDirect

CAAI Transactions on Intelligence Technology 1 (2016) 313–322
http://www.journals.elsevier.com/caai-transactions-on-intelligence-technology/

Review Article

Building a click model: From idea to practice

Chao Wang, Yiqun Liu, Shaoping Ma*

Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

Received 5 September 2016; accepted 8 December 2016
Available online 23 December 2016

Abstract

Click-through information is considered as a valuable source of users' implicit relevance feedback. As user behavior is usually influenced by a number of factors such as position, presentation style and site reputation, researchers have proposed a variety of assumptions to generate a reasonable estimation of result relevance. Therefore, many click models have been proposed to describe how user click action happens and to predict click probability (and search result relevance). This work builds upon many years of existing efforts from THUIR labs, summarizes the most recent advances and provides a series of practical click models. In this paper, we give an introduction of how to build an effective click model. We use two click models as specific examples to introduce the general procedures of building a click model. We also introduce common evaluation metrics for the comparison of different click models. Some useful datasets and tools are also introduced to help readers better understand and implement existing click models. The goal of this survey is to bring together current efforts in the area, summarize the research performed so far and give a view on building click models for web search.

Keywords: Search engine; User behavior analysis; Click model

1. Introduction

Relevance estimation has been the most critical problem since the birth of web search. As web content was explosively generated, modern search engines have been required to efficiently and effectively retrieve the relevant URLs from a prohibitively large corpus. This raises tremendous challenges for both industrial and academic researchers. Early works on search relevance concentrated on text matching between queries and URLs such as BM25 [1], probabilistic retrieval model [1–3], and vector space model [4].

Besides the content information of search result, user behavior demonstrated great potential for relevance improvement in the industrial setting, and user behavior modeling has been extensively explored for improving search relevance. Commercial search engines usually record large-scale user interaction logs every day and many research issues in Web search (e.g. click prediction, Web search ranking, query suggestion, etc.) are closely related to these behavior logs.

While user clicks provide implicit information about user's perceived relevance on the results, they are not true, accurate, relevance feedback. Therefore, various methods have been proposed to cope with noisy nature of user clicks. Joachims et al. [5] worked on extracting reliable implicit feedback from user behaviors, and concluded that click logs are informative yet biased. Previous studies revealed several bias aspects such as “position” [5,6], “trust” [7] and “presentation” [8] factors. To address these issues, researchers have proposed a number of click models to describe user's examination behavior on search engine result pages (SERPs) and to obtain an unbiased estimation of result relevance [9–11].

Most existing click models are formulated within the framework of probabilistic graphic model. In these models, a group of variables are usually used to model each search result

* Corresponding author.
E-mail address: msp@mail.tsinghua.edu.cn (S. Ma).

Peer review under responsibility of Chongqing University of Technology.

http://dx.doi.org/10.1016/j.trit.2016.12.003

for a specific query. The variables include the observable click actions and some hidden variables such as user examination, result relevance, user satisfaction after viewing this result, etc. Different click models make different user behavior assumptions (e.g. cascade assumption [6]) to construct the network structure among the variables. Once constructed, these click models can be trained on a large set of user click-through logs and then used to predict click probabilities for results or to rerank the search result list according to the inferred relevance.

This paper focuses on sharing our experiences in building click models and introduces two practical click models that have been successfully implemented. We first introduce the Vertical-aware Click Model (VCM) [8]. This model focuses on the problem that when vertical results are combined with ordinary ones, significant differences in presentation may lead to user behavior biases. To build this model, we collected a large scale log data set which contains behavior information on both vertical and ordinary results. We also performed eye-tracking analysis to study users real-world examining behavior. According to these analysis, we found that different result appearances may cause different behavior biases both for vertical results (local effect) and for the whole result lists (global effect). These biases include: examine bias for vertical results (especially those with multimedia components), trust bias for result lists with vertical results, and a higher probability of result revisitation for vertical results. Based on these findings, a novel click model considering these biases besides position bias was constructed to describe interaction with SERPs containing verticals.

The second click model is Partially Sequential Click Model (PSCM) [12]. This model focuses on the problem that most existing click models follow the *sequential examination hypothesis* in which users examine results from top to bottom in a linear fashion but many studies showed that there is a large proportion of non-sequential browsing (both examination and click) behaviors in Web search. To build this model, we carry out a laboratory eye-tracking study to analyze user's non-sequential examination behavior and then propose PSCM model that captures the practical behavior of users.

According to these click models, we may conclude that the main procedures of building a click model are: 1) investigate user behaviors from data to summarize the behavior patterns from different users and different queries; 2) formalize these behavior patterns to mathematical behavior assumptions (build a model); 3) design the learning method (parameter inference method) for this model; 4) train the proposed model with the learning method with large scale user behavior data (usually click-through data with result impressions); 5) use the trained model to make predictions (user click prediction or result relevance estimation); 6) evaluate model performance via different evaluation metrics.

Therefore, the rest of the paper is organized as follows: in Section 2, we review some existing efforts in constructing click models for Web search. In Section 3, we introduce the user behavior analysis methods. The model construction and parameter inference process are discussed in Section 4. In Section 5, we present the popular evaluation metrics for click models. And finally we introduce some useful tools and datasets in Section 6.

Our contributions in this paper are:

- We briefly introduce the common process of building and testing a search click model.
- We use two click models (VCM and PSCM) as specific examples to show the details in each step.
- We introduce the common evaluation metrics for comparing different click models.
- We also describe software packages and public datasets that we find useful to work with click models.

2. Background

In this section, we review a number of essential click models and introduce some preliminary assumptions shared by these models. For details of most of these models, the readers can refer to the recent survey book by Chuklin et al. [13].

2.1. Basic click models

Most click models follow the examination hypothesis [6]: a document being clicked $(C_i = 1)$ should satisfy $(\rightarrow)$ two conditions: it is examined $(E_i = 1)$ and it is relevant $(R_i = 1)$ (most click models assume $P(R_i = 1) = r_u$, which is the probability of the perceived relevance), and these two conditions are independent of each other.

$$C_i = 1 \rightarrow E_i = 1, R_i = 1 \quad (1)$$

$$E_i = 0 \rightarrow C_i = 0 \quad (2)$$

$$R_i = 0 \rightarrow C_i = 0 \quad (3)$$

Following this assumption, the probability of a document being clicked is determined as follows:

$$P(C_i = 1) = P(E_i = 1)P(R_i = 1) \quad (4)$$

Based on the assumption that a user examines from top position to bottom position, this kind of click models naturally takes position bias into account.

Craswell et al. [6] proposed the cascade model, which assumes that while a user examines the results from top to bottom sequentially, he/she immediately decides whether to click on a result. The cascade model is mostly suitable for single-click sessions. A number of succeeding models were proposed to improve both its applicability and performance.

$$P(E_1) = 1 \quad (5)$$

$$P(E_{i+1} = 1 | E_i = 1, C_i) = 1 - C_i \quad (6)$$

Here the examination of the (i+1)-th result indicates the i-th result has been examined but not clicked. Although the cascade model performs well in predicting the click-through rates, this model is only suited for a single-click scenario.

Based on the cascade hypothesis, the Dependency Click Model (DCM) [9] extends the cascade model in order to model user interactions within multi-click sessions. DCM assumes that a user may have a certain probability of examining the next document after clicking the current document, and this probability is influenced by the ranking position of the result. The DCM model is characterized as follows:

$$P(E_{i+1} = 1 | E_i = 1, C_i = 0) = 1 \tag{7}$$

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

where λ_i represents the preservation probability[1] of the position i.

Subsequently, the User Browsing Model (UBM) [10] further refines the examination hypothesis by assuming that the event of a document being examined depends on both the preceding click position and the distance between the preceding click position and the current one.

$$P(E_i = 1 | C_{1\ldots i-1}) = \lambda_{r_i, d_i} \tag{9}$$

where r_i represents the preceding click position and d_i is the distance between the current rank and r_i.

The Dynamic Bayesian Network model (DBN) [10] is the first model to consider presentation bias due to snippet (rather than ranking position). This model distinguishes the actual relevance from the perceived relevance, where the perceived relevance indicates the relevance represented by titles or snippets in SERPs and the actual relevance is the relevance of the landing page.

$$P(R_i = 1) = r_u \tag{10}$$

$$P(S_i = 1 | C_i = 1) = s_u \tag{11}$$

$$P(E_{i+1} | E_i = 1, S_i = 0) = \lambda \tag{12}$$

where S_i represents whether 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, and λ represents the probability of continuing the examination process.

Subsequently, the Click Chain Model (CCM) [14] uses Bayesian inference to obtain the posterior distribution of the relevance. In contrast to other existing models, this model introduces skipping behaviors. CCM is scalable for large-scale click-through data, and the experimental results show that it is effective for low frequency (also known as long-tail) queries.

2.2. Advanced click models

Here we summarize some more recent click models that improve the basic click models. Most of these models built on one of the basic click models by introducing new parameters, using more data and, more generally, incorporating additional knowledge about user behavior.

It is evident that the vertical results are often more visually salient and attract more user attention. Moreover, Chen et al. [15] show that vertical blocks also affect the amount of attention that nearby non-vertical documents get. After performing a deep analysis of different peculiarities related to verticals, Wang et al. [8] suggest a complex model that is based on UBM and incorporates four different types of bias: Attraction bias, Global bias, First place bias and Sequence bias to describe user behavior when facing to vertical results.

Chuklin et al. [16] suggest to look at vertical search as if there are different users coming with different intents (needs): organic web, Image, News, Video, etc. One may then use different examination and click probabilities for different intents, assuming that the intent distribution is known for each query. The authors suggest to go one step further and also take visual aspects into account, hypothesizing that the use of special presentation formats for, e.g., News results will lead to different examination patterns than if these results are presented as organic web results.

It is evident that the more heterogeneous a SERP is, the more likely it is that the user is going to examine it in a nonlinear way. Wang et al. proposed the Partially Sequential Click Model (PSCM) [12] to take click sequence information into consideration. The PSCM model proposed two additional user behavior assumptions based on eye-tracking experiments: The first one assumes that although the examination behavior between adjacent clicks can be regarded as locally unidirectional, users may skip a few results and examine a result at some distance from the current one following a certain direction. The second one assumes that between adjacent clicks, users tend to examine search results in a single direction without changes, and the direction is usually consistent with that of clicks. This model distinguishes the result position from the examination order, and shows a better click prediction performance than position-based click models.

Besides PSCM's efforts in incorporating non-sequential behaviors into click models, there are some other works working on similar directions. For example, Zhang et al. [17] proposed a click model based on Recurrent Neural Networks (RNN) for sponsored search. They directly model the dependency on users sequential behaviors into the click prediction process through the recurrent structure in RNN. Borisov et al. [18] also proposed an RNN based click model to model user's sequential click behaviors. These models only take click sequence information into account and ignore the influence of different click dwell time among click actions.

3. User behavior analysis

As our introduction above, most click models follow a building process that making analysis of user behavior first and then summarize different user behavior patterns. Therefore, we first introduce the user behavior analysis process.

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

3.1. Click log analysis

To investigate user behavior information, we must first collect user interaction data with search engines. As user click information is very useful and very easy to be collected by commercial search engine, lots of user behavior analysis are based on the search and click logs. For example, in Vertical-aware click model [8], the data set contains 53,080,107 query sessions with 15,149,469 distinct queries during the time period in April 2012.

Using click logs, we can analysis the click distribution difference among different situations and find some user preference according these kinds of comparisons. For example, in Vertical-aware click model [8], we can figure out two phenomena according to Fig. 1: 1) different verticals have different global influence on users clicking preference; 2) multimedia vertical increases global click-through rate (CTR) while application vertical decreases global CTR.

3.2. Eye-tracking study

Although click log analysis helps us know a lot about user's search behavior, we cannot completely know user's browsing process via click logs. Therefore, some researchers also carry out laboratory eye-tracking study to see user's actual eye movement in search process. Due to the reason that eye-trackers are expensive and are difficult to be widely used, the scale of eye-tracking studies are commonly much smaller than log analysis.

The number of subjects in eye-tracking studies is usually less than 100 [19–21]. With an eye-tracking device (e.g. Tobii X2-30), we can record each subject's eye movement information on search engine result pages (SERPs). For quality control purposes, each subject was asked to make an eye-tracking calibration before the experiment.

With the eye-tracking device, two types of eye movement information can be collected: saccades and fixations. Saccade means fast eye movements from point to point in jerks, while fixation means that eyes stop for a short period of time [22]. Because new information is mainly acquired during fixations, most existing studies [23–25] assumed that eye fixation is equivalent to user examination sequence. Recent study [21] showed that eye fixation does not necessary mean examination in many cases. Therefore, saccade information may also be very useful in search process.

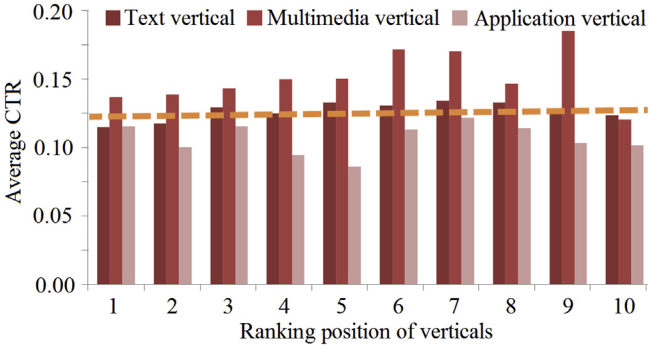


Fig. 1. Average CTR of the first page when different kinds of vertical results appear from rank 1 to rank 10.

3.2.1. Vertical-aware Click Model

For VCM model, we want to find the answers to the following question about users' examination behavior on the SERPs: ***RQ***: Do users examine verticals first?

To analyze which result the user first pays attention to, we collect subjects first 2 s eye fixations on the screen.

Figs. 2 and 3 shows two examples from eye-tracking data which shows users watching area on SERP with different kinds of verticals or no vertical results. We can see that users pay most attention to the first result when there is no vertical in SERP (which should be regarded as sign for position bias). However, when there is a multimedia vertical result at the third position, it attracts a lot of users direct attentions. With this observation, we can formulate the following behavior assumption:

Attraction Bias: If there is a vertical placed in the SERP, there is probability that users examine it first.

3.2.2. Partially Sequential Click Model

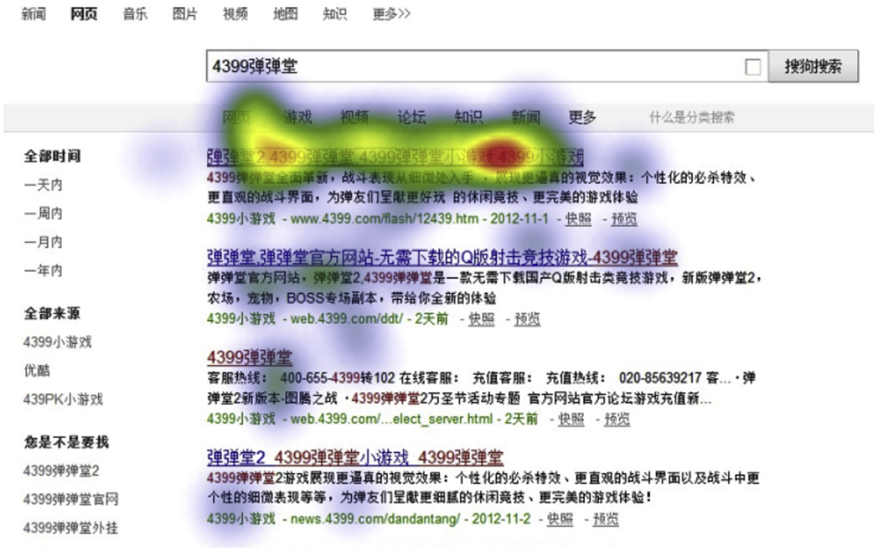
For PSCM model, we want to find the answers to the following two questions about users' examination behavior on the SERPs: ***RQ1***: How often do users change the direction of examination between clicks? ***RQ2***: How far do users' eye gazes jump after examining the current clicked result?

By investigating these two questions, we aim to understand how users behave and to propose corresponding user behavior assumptions in order to model users' examination behavior in a more reasonable way. To simplify the notation, suppose that the first click is at position i and the next click is at position j, if $i<j$, it is a sequential action according to the depth-first assumption (this direction is referred to as "↓"). If $i \geq j$, it is a non-sequential click action according to the definition of revisiting behavior (this direction is referred to as "↑").

To answer the two research questions, we firstly divide all examination sequences into adjacent examination behavior pairs. For a given examination sequence $E = <E_1, E_2, \ldots, E_t, \ldots, E_T>$, it will be divided into $T-1$ pairs: $(E_1, E_2), (E_2, E_3), \ldots, (E_{T-1}, E_T)$. For each pair, similar with the definition of direction in adjacent clicks, we can define its direction as ↑ or ↓ according to whether the sequence of the examination pair follows a depth-first assumption or not.

To investigate ***RQ1***, we consider the examination sequence between ↑ and ↓ adjacent clicks separately. Intuitively, one may believe that the examination sequence between ↓ adjacent clicks should follow the depth-first assumption. In other words, the examination sequence would be consistent with the click sequence.

However, it is also possible that some parts in the examination sequence follow a non-sequential order. Similarly, the examination sequence between ↑ adjacent clicks may also contain ↓ adjacent examination pairs. To find out how often the examination direction change happens between adjacent



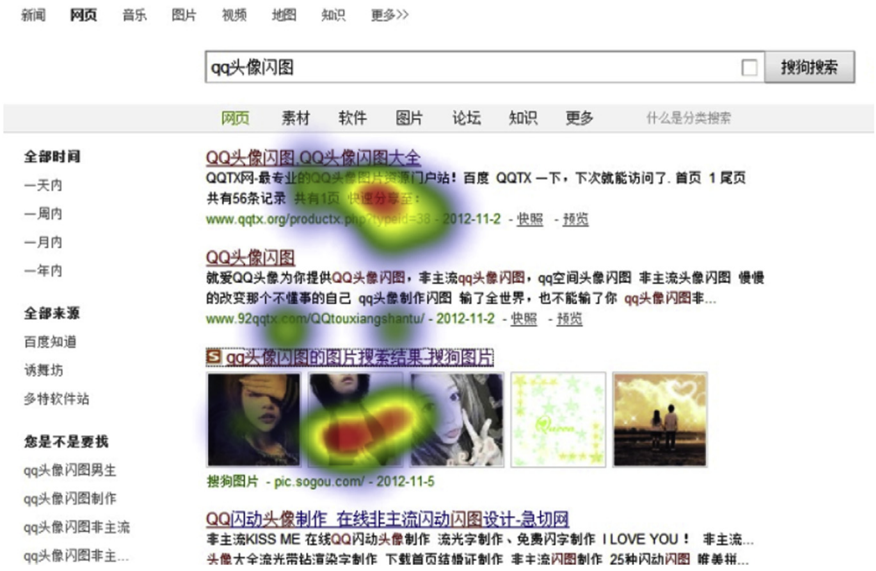
弹弹堂2,4399弹弹堂,4399弹弹堂小游戏,4399小游戏
4399弹弹堂全面革新，战斗表现从细微处入手，展现更逼真的视觉效果：个性化的必杀特效、更直观的战斗界面，为弹友们呈献更好玩 的休闲竞技、更完美的游戏体验
4399小游戏 - www.4399.com/flash/12439.htm - 2012-11-1 - 快照 - 预览

弹弹堂,弹弹堂官方网站-无需下载的Q版射击竞技游戏-4399弹弹堂
弹弹堂官方网站，弹弹堂2,4399弹弹堂是一款无需下载国产Q版射击类竞技游戏，新版弹弹堂2，农场，宠物，BOSS专场副本，带给你全新的体验
4399小游戏 - web.4399.com/ddt/ - 2天前 - 快照 - 预览

4399弹弹堂
客服热线： 400-655-4399转102 在线客服： 充值客服： 充值热线： 020-85639217 客... ·弹弹堂2新版本-图腾之战 ·4399弹弹堂2万圣节活动专题 官方网站官方论坛游戏充值新...
4399小游戏 - web.4399.com/...elect_server.html - 2天前 - 快照 - 预览

弹弹堂2 4399弹弹堂小游戏 4399弹弹堂
4399弹弹堂2游戏展现更逼真的视觉效果：个性化的必杀特效、更直观的战斗界面以及战斗中更个性的细微表现等等，为弹友们呈献更细腻的休闲竞技、更完美的游戏体验！
4399小游戏 - news.4399.com/dandantang/ - 2012-11-2 - 快照 - 预览

Fig. 2. Heat map of the subjects eye fixation areas in first 2 s on an SERP with no vertical.



QQ头像闪图,QQ头像闪图大全
QQTX网-最专业的QQ头像图片资源门户站！百度 QQTX 一下，下次就能访问了. 首页 1 尾页 共有56条记录 共有1页 快速分享至：
www.qqtx.org/productx.php?typeid=38 - 2012-11-2 - 快照 - 预览

QQ头像闪图
就爱QQ头像为你提供QQ头像闪图，非主流qq头像闪图，qq空间头像闪图 非主流头像闪图 慢慢的改变那个不懂事的自己 qq头像制作闪图 输了全世界，也不能输了你 qq头像闪图非...
www.92qqtx.com/QQtouxiangshantu/ - 2012-11-2 - 快照 - 预览

S qq头像闪图的图片搜索结果-搜狗图片
搜狗图片 - pic.sogou.com/ - 2012-11-5

QQ闪动头像制作 在线非主流闪动闪图设计-急切网
非主流KISS ME 在线QQ闪动头像制作 流光字制作、免费闪字制作 I LOVE YOU！ 非主流... 头像大全流光带钻渲染字制作 下载首页结婚证制作 非主流闪图制作 25种闪动闪图 唯美拼...

Fig. 3. Heat map of the subjects eye fixation areas in first 2 s on an SERP with multimedia vertical placed at the third position.

clicks, we count the number of examination direction changes and the distributions are shown in Fig. 4.

From this figure, we can see that no matter whether the click direction is ↑ or ↓, in most cases (72.7% for ↓ and 78.9% for ↑) the whole examination sequences follow the same direction as click direction without any direction changes. The percentage of sequences with direction changes between ↓ clicks is slightly larger than that between ↑ clicks. This phenomenon corresponds well to the behavior pattern in which users re-examine some higher-ranked results before moving to the lower-ranked ones. With this observation, we can formulate the following behavior assumption:

Locally Unidirectional Examination Assumption: Between adjacent clicks, users tend to examine search results in a

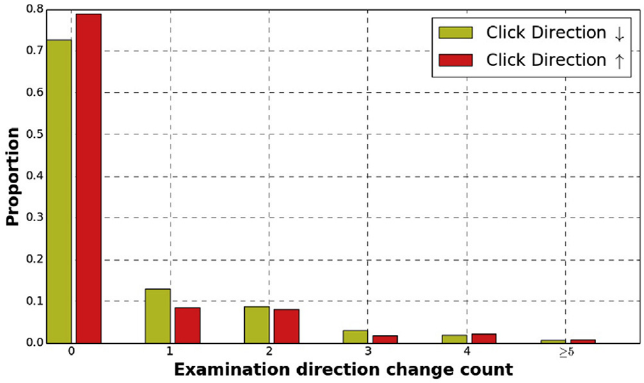


Fig. 4. Distribution of examination direction change count for two types of adjacent clicks.

single direction without changes, and the direction is usually consistent with that of clicks no matter it is ↑ or ↓.

To answer ***RQ2***, we look at the average examination transition distance within adjacent examination pairs. For a given adjacent examination pair (E_{t-1}, E_t), suppose that the first examination E_{t-1} is at position k while the next examination E_t is at position l, the transition distance can be calculated as $|k - l|$. Fig. 5 shows the distribution of transition distance in different result positions.

We can see that all transition distances are around 1.25 when user follows top-down (↓) click sequences. While when user follows bottom-up (↑) click sequences, his/her eyes may skip several results to find a specific result.

In particular, we observe larger transition distances for bottom ranking positions, which tend to bring back to the middle positions (positions 5–6) in the list. As all the transition distances are statistically significantly larger than 1 ($p - value < 0.01$ for each position and each click direction based on t-test), we can make the following behavior assumption:

Non First-order Examination Assumption: although the examination behavior between adjacent clicks can be regarded as locally unidirectional, users may skip a few results and examine a result at some distance from the current one following a certain direction.

4. Model construction

After analyzing user behavior patterns, we need to abstract these patterns into mathematical formulations and infer the learning method of such models.

4.1. Vertical-aware click model

The proposed Vertical-aware click model (VCM) is described as follows:

$$P(C_i = 1|E_i = 0) = 0 \tag{13}$$

$$P(C_i = 1|E_i = 1) = P(R_i = 1|E_i = 1) \tag{14}$$

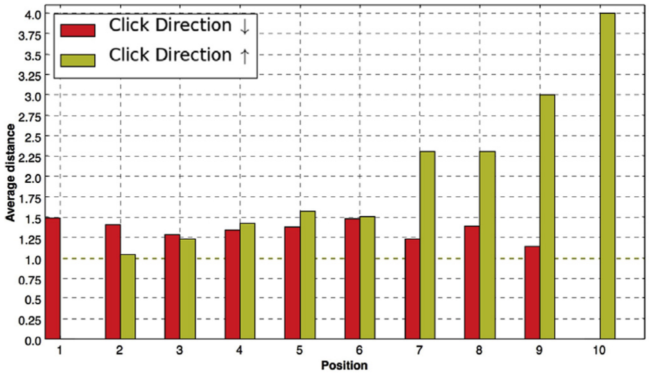


Fig. 5. Average examination transition distance according to different examination transition start positions for two types of adjacent clicks.

$$P(F = 1) = \phi_{t_v, l_v} \tag{15}$$

$$P(E_i = 1|F = 0, C_{1:i-1}) = \gamma_{i,i-l_i} \tag{16}$$

$$P(E_i = 1|F = 1, C_{1:i-1}) = \gamma_{i,i-l_i} + \theta_{q,i} \tag{17}$$

$$P(R_i = 1|E_i = 1, F = 0) = \alpha_{q,i} \tag{18}$$

$$P(R_i = 1|E_i = 1, F = 1) = \alpha_{q,i} + \beta_{q,i} \tag{19}$$

$$P(B = 1|F = 0) = 0 \tag{20}$$

$$P(B = 1|F = 1) = \sigma_{t_v, l_v} \tag{21}$$

Fig. 6 shows the decision-making process of VCM. When user begins with a query session, the user will have the opportunity to examine the vertical first if there is a vertical result in SERP. After examining the vertical first, the user will decide to scan back to the previous document in bottom up sequence or top down sequence.

4.2. Partially Sequential Click Model

The *First-order Click Hypothesis* is usually accepted in most click models such as DBN and UBM. We do the same in this work. It supposes that the click event at time $t + 1$ is only determined by the click event at time t. According to this hypothesis, user's click action $C = <C_1, C_2, \ldots, C_t, \ldots, C_T>$ can be independently separated to $T + 1$ adjacent click pairs: $<C_0, C_1>, \ldots, <C_{t-1}, C_t>, \ldots, <C_T, C_{T+1}>$ (C_0 represents the begining of search process and C_{T+1} represents the end of

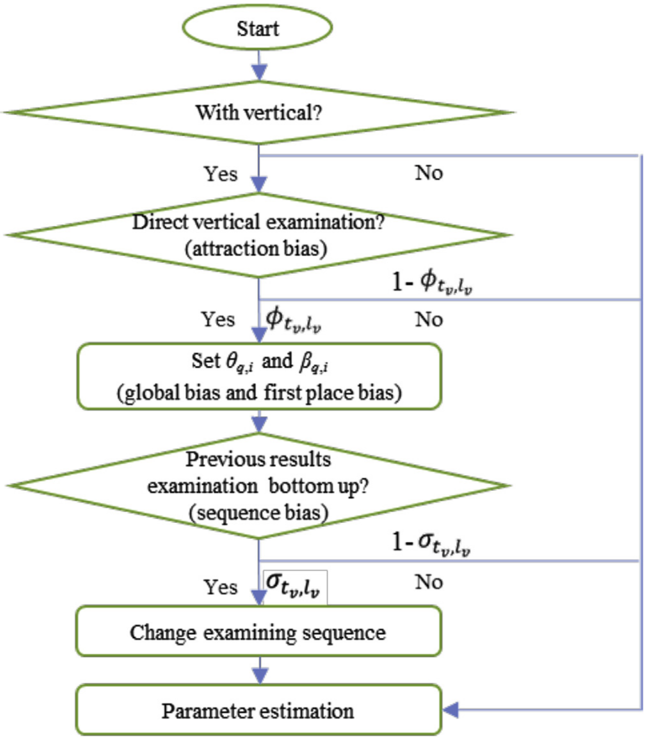


Fig. 6. Decision-making process of VCM.

search process). This makes it possible for us to divide a click sequence into sub-sequences (adjacent click pairs).

According to the *Locally Unidirectional Examination Assumption*, given an observation of adjacent clicks at time t: $O = \{<C_{t-1} = m, C_t = n>\}$, users tend to examine the results on the path from m to n without any direction changes. Then the examination and click sequence between C_{t-1} and C_t can be noted as $<\overline{E}_m, \ldots, \overline{E}_j, \ldots, \overline{E}_n>$ and $<\overline{C}_m, \ldots, \overline{C}_j, \ldots, \overline{C}_n>$, respectively. Note that different from C_t which is used to record the position of click event, $\overline{E}_j$ and $\overline{C}_j$ $(m \le j \le n\, or\, n \le j \le m)$ are all binary variables representing whether examination or click behavior happens (=1) or not (=0) on the corresponding result position. In addition, we can also deduce that in the click sequence, only $\overline{C}_m$ and $\overline{C}_n$ have value 1 and the other positions on the path have value 0.

The proposed Partially Sequential Click Model (PSCM) adopts these two assumptions. It is then described as follows:

$$P(C_t|C_{t-1}, \ldots, C_1) = P(C_t|C_{t-1}) \tag{22}$$

$$\begin{aligned} &P(C_t = n|C_{t-1} = m) = \\ &P\left(\overline{C}_m = 1, \ldots, \overline{C}_i = 0, \ldots, \overline{C}_n = 1\right) \end{aligned} \tag{23}$$

$$\begin{aligned} &P\left(\overline{E}_i = 1\middle|C_{t-1} = m, C_t = n\right) = \\ &\begin{cases} \gamma_{imn}, m \le i \le n\, or\, n \le i \le m \\ 0, other \end{cases} \end{aligned} \tag{24}$$

$$\overline{C}_i = 1 \Leftrightarrow \overline{E}_i = 1, R_i = 1 \tag{25}$$

$$P(R_i = 1) = \alpha_{uq} \tag{26}$$

Equation (22) encodes the *first-order click hypothesis* while Equation (23) encodes the *locally unidirectional examination assumption* by restricting the examination process to one-way from m to n. We define the examination probability of $\overline{E}_i$ as Equation (24) because according to Fig. 5, the examination behavior between adjacent clicks may not follow cascade assumptions (non first-order examination assumption). The probability of examination depends on the positions of the clicks. This is similar to UBM, which also allow skips, but only within sequential behaviors. PSCM also follows examination hypothesis described in Equation (25) as in most existing click models. α_{uq} corresponds to the relevance of the document URL u at position i to the specific query q.

Fig. 7 shows the framework of the PSCM model. Unlike previous position-based models (such as UBM or DBN) which suppose user examine results top-down sequentially, PSCM allows non-sequential interactions. A user may click on a lower position (m) and then a higher position ($n > m$), with all the documents between them having some probability to be examined. Such a behavior is modeled by Equation (23) and Equation (24).

4.3. Parameter inference

We use the expectation-maximization (EM) algorithm to complete the inference step. The EM algorithm is used to find the maximum likelihood estimates of parameters. The EM iteration alternates between performing an E-step, which creates a function for the expectation of the Log-Likelihood evaluated using the current estimate for the parameters, and M-step, which computes parameters maximizing the expected Log-Likelihood found on the E-step. The detail inference step can be found in the original papers [8,12].

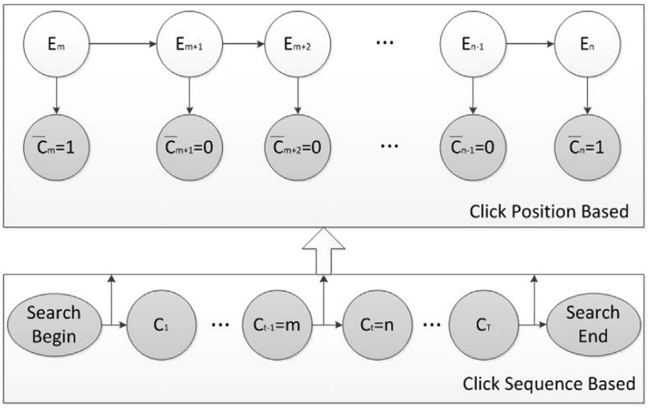


Fig. 7. Sketch of Partially Sequential Click Model. Click actions are listed according to their click timestamps. For each adjacent click pair, a position-based framework is constructed based on their click positions.

5. Evaluation metrics

In this section, we briefly introduce common evaluation metrics for click models. As a click model can predict user click for a new search sessions, click probability prediction is the first evaluation field for these models. Most of click models can also give query-result relevance estimation, therefore, if we have human labels for such query-result pairs, we can also test click model's performance via some ranking evaluation metrics.

5.1. Click prediction

5.1.1. Perplexity

The first metric designed specifically for comparing click models was cross-entropy proposed by Craswell et al. [6]. This metric was not easy to interpret and it did not become widely used. Instead a conceptually similar perplexity metric was proposed by Dupret and Piwowarski [10]:

$$Perplexity_i = 2^{-\frac{1}{N}\sum_j^N (C_i log p_i + (1-C_i) log(1-p_i))} \tag{27}$$

where $Perplexity_i$ is the perplexity score in i^{th} result position and N is the total session count, C_i is the actual user click information and p_i is the predicted click probability. The overall click perplexity score is the average of all positions (10 in our dataset).

Click perplexity indicates how well a model can predict the clicks. A smaller perplexity value indicates a better modeling performance, and the value reaches 1 in the ideal case. The improvement of click perplexity CP_1 over CP_2 is usually calculated as $\frac{CP_2 - CP_1}{CP_2 - 1} * 100\%$ [8,15].

5.1.2. Log-likelihood

Whenever we have a statistical model, we can evaluate its accuracy by looking at the likelihood of some held-out test set [26]. For each session in the test set we compute how likely this session is according to a click model. If we further assume independence of the sessions, we can compute the logarithm of the joint likelihood. This metric is known as log-likelihood and usually decomposed using the formula of total probability. As a logarithm of a probability measure, this metric always has non-positive values with higher values representing better prediction quality.

5.2. Relevance estimation

Targeting on assessing base relevance, professional editors judgment is leveraged. Editors manually judge the query-result pairs in five grade: Perfect, Excellent, Good, Fair and Bad. Then Normalized Discounted Cumulative Gain (NDCG) [27] is used as the common metric to evaluate the search relevance performance. The NDCG is an important and popular metric for measuring the performance of ranking algorithms. As each click model can provide its query-result relevance prediction after training process, once we obtain the relevance label for each query-result pair, we are able to test the ranking performances with NDCG.

5.3. User preference test

Besides the evaluation in relevance estimation, sometimes we also want to find out whether the ranking lists provided by one specific model are preferred by real users than other models. Therefore, a side-by-side user preference test can be conducted to test the user's real preference [12,28]. According to this kind of test, users are asked to label their preference on the whole ranking list of different click models and the final preference is generated via voting methods.

6. Useful dataset and tools

To infer parameters and evaluate performance of click models, researchers use click logs, i.e., logs of user search sessions with click-through information. Such logs are produced by live search systems and contain highly sensitive information in terms of privacy and commercial value. For this reason, publicly releasing such data is very challenging and requires a lot of work. Therefore, we discuss publicly available click logs. We also describe software packages and libraries that we find useful to work with click models.

6.1. Datasets

- AOL. One of the first publicly released datasets was the AOL query log released in 2006. It was a comprehensive dataset containing twenty million search sessions for over 650,000 users over a 3-month period. The data was not redacted for privacy, which led the company to withdraw the dataset just a couple of days after its release. It is one of the few datasets that contain actual queries and document URLs, which makes it valuable in spite of the fact that it represents a different generation of web search users.
- WSCD. Provided by Yandex.com [29,30], the WSCD 2012 dataset6 consists of user search sessions extracted from Yandex logs around 2009. The dataset contains anonymized queries, URL rankings, clicks and relevance judgments for ranked URLs. In addition, queries are grouped into search sessions. The WSCD 2013 dataset was extracted from Yandex logs around 2011 and the WSCD 2014 dataset is collected around 2012.
- SogouQ. Provided by Sogou.com, the SogouQ dataset contains anonymized user ids, queries, URL rankings and clicks. Query strings and document URLs are not obfuscated and provided verbatim in the click log. This allows researchers to perform query similarity analysis, document analysis and other applications that are not possible with numeric ids. The downside of this dataset is that it only provides information about clicked documents, so the exact set of documents shown to a user can only be approximated.

6.2. Tools

- ClickModels project.[2] It provides an open-source implementation of state-of-the-art click models, namely Dynamic Bayesian Network model (DBN) [11] (simplified and full versions) and User Browsing Model (UBM) [10].
- PyClick.[3] It provides an open-source implementation of state-of-the-art click models, namely Task-centric Click Model (TCM) [31], Federated Click Model (FCM) [15], and Vertical-aware Click Model (VCM) [8].
- THUIRClick.[4] It provides an open-source implementation of state-of-the-art click models, namely Partially Sequential Click Model (PSCM) [12], partially observable Markov Model (POM) [32], Temporal Hidden Click Model (THCM) [33], Temporal Click Model (TCM) [34].

7. Conclusion

In this paper, we give an introduction of how to build an effective clcik model. We use two click models (VCM and PSCM) as specific examples to introduce the general procedures of building a click model. We show the examples of how to analyze user behavior from eye-tracking studies. We also introduce common evaluation metrics for the comparison of different click models. Some useful datasets and tools are also introduced to help readers better understand and implement existing click models. The goal of this survey is to bring together current efforts in the area, summarize the research performed so far and give a view on building click models for web search.

[2] https://github.com/varepsilon/clickmodels.

[3] https://github.com/markovi/PyClick.

[4] https://github.com/THUIR/PSCMModel.

References

[1] S. Robertson, H. Zaragoza, The Probabilistic Relevance Framework: BM25 and Beyond, Now Publishers Inc, 2009.

[2] S. Robertson, H. Zaragoza, M. Taylor, Simple bm25 extension to multiple weighted fields, in: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, ACM, 2004, pp. 42–49.

[3] J.H. Paik, A novel tf-idf weighting scheme for effective ranking, in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 343–352.

[4] G. Salton, M.J. McGill, Introduction to Modern Information Retrieval, 1986.

[5] T. Joachims, L. Granka, B. Pan, H. Hembrooke, G. Gay, Accurately interpreting clickthrough data as implicit feedback, in: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2005, pp. 154–161.

[6] N. Craswell, O. Zoeter, M. Taylor, B. Ramsey, An experimental comparison of click position-bias models, in: Proceedings of the 2008 International Conference on Web Search and Data Mining, ACM, 2008, pp. 87–94.

[7] Y. Yue, R. Patel, H. Roehrig, Beyond position bias: examining result attractiveness as a source of presentation bias in clickthrough data, in: Proceedings of the 19th International Conference on World Wide Web, ACM, 2010, pp. 1011–1018.

[8] C. Wang, Y. Liu, M. Zhang, S. Ma, M. Zheng, J. Qian, K. Zhang, Incorporating vertical results into search click models, in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 503–512.

[9] F. Guo, C. Liu, Y.M. Wang, Efficient multiple-click models in web search, in: Proceedings of the Second ACM International Conference on Web Search and Data Mining, ACM, 2009, pp. 124–131.

[10] G.E. Dupret, B. Piwowarski, 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, 2008, pp. 331–338.

[11] O. Chapelle, Y. Zhang, A dynamic bayesian network click model for web search ranking, in: Proceedings of the 18th International Conference on World Wide Web, ACM, 2009, pp. 1–10.

[12] C. Wang, Y. Liu, M. Wang, K. Zhou, J. Nie, S. Ma, Incorporating non-sequential behavior into click models, in: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 283–292.

[13] A. Chuklin, I. Markov, M. de Rijke, Click Models for Web Search, Morgan and Claypool, 2015.

[14] F. Guo, C. Liu, A. Kannan, T. Minka, M.J. Taylor, Y.M. Wang, C. Faloutsos, Click chain model in web search, in: WWW'09, 2009, pp. 11–20.

[15] D. Chen, W. Chen, H. Wang, Z. Chen, Q. Yang, 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, 2012, pp. 463–472.

[16] A. Chuklin, P. Serdyukov, M. De Rijke, Using intent information to model user behavior in diversified search, in: European Conference on Information Retrieval, Springer, 2013, pp. 1–13.

[17] Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, T.-Y. Liu, Sequential click prediction for sponsored search with recurrent neural networks, in: Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014.

[18] A. Borisov, I. Markov, M. de Rijke, P. Serdyukov, A neural click model for web search, in: The 25th International Conference on World Wide Web, 2016.

[19] L.A. Granka, T. Joachims, G. Gay, Eye-tracking analysis of user behavior in www search, in: SIGIR 04, ACM, 2004, pp. 478–479.

[20] E. Cutrell, Z. Guan, What are you looking for?: an eye-tracking study of information usage in web search, in: CHI 07, ACM, 2007, pp. 407–416.

[21] Y. Liu, C. Wang, K. Zhou, J. Nie, M. Zhang, S. Ma, From skimming to reading: a two-stage examination model for web search, in: CIKM'14, ACM, 2014, pp. 849–858.

[22] K. Rayner, Eye movements and attention in reading, scene perception, and visual search, Q. J. Exp. Psychol. 62 (8) (2009) 1457–1506.

[23] J. Huang, R.W. White, S. Dumais, No clicks, no problem: using cursor movements to understand and improve search, in: CHI's 11, ACM, 2011, pp. 1225–1234.

[24] G. Buscher, S. White, W. Ryen, J. Huang, Large-scale analysis of individual and task differences in search result page examination strategies, in: WSDM'12, ACM, 2012, pp. 373–382.

[25] R. Navalpakkam, Vidhya, S. Ravi, A. Ahmed, A. Smola, Measurement and modeling of eye-mouse behavior in the presence of nonlinear page layouts, in: WWW 13, 2013, pp. 953–964.

[26] R.A. Fisher, On the mathematical foundations of theoretical statistics, Philosophical Trans. R. Soc. Lond. Ser. A, Contain. Pap. a Math. or Phys. Character 222 (1922) 309–368.

[27] K. Järvelin, J. Kekäläinen, Cumulated gain-based evaluation of ir techniques, ACM Trans. Inf. Syst. (TOIS) 20 (4) (2002) 422–446.

[28] L. Jie, S. Lamkhede, R. Sapra, E. Hsu, H. Song, Y. Chang, A unified search federation system based on online user feedback, in: SIGKDD'13, ACM, 2013, pp. 1195–1203.

[29] P. Serdyukov, G. Dupret, N. Craswell, Wscd2013: workshop on web search click data 2013, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, ACM, 2013, pp. 787–788.

[30] P. Serdyukov, G. Dupret, N. Craswell, Log-based personalization: the 4th web search click data (wscd) workshop, in: Proceedings of the 7th ACM International Conference on Web Search and Data Mining, ACM, 2014, pp. 685–686.

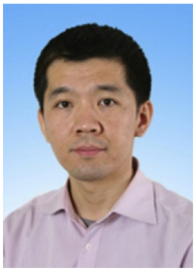
[31] Y. Zhang, W. Chen, D. Wang, Q. Yang, User-click modeling for understanding and predicting search-behavior, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2011, pp. 1388–1396.

[32] K. Wang, N. Gloy, X. Li, Inferring search behaviors using partially observable markov (pom) model, in: Proceedings of the Third ACM International Conference on Web Search and Data Mining, ACM, 2010, pp. 211–220.

[33] D. Xu, Y. Liu, M. Zhang, S. Ma, L. Ru, Incorporating revisiting behaviors into click models, in: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, ACM, 2012, pp. 303–312.

[34] W. Xu, E. Manavoglu, E. Cantu-Paz, Temporal click model for sponsored search, in: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2010, pp. 106–113.

Chao WANG is a senior researcher at Baidu.com. His research interest focuses on User behavior modeling and Search Ranking.



Yiqun LIU is working as an associate professor in Department of Computer Science and Technology in Tsinghua University. He is a council member of CAAI and mainly works on Information Retrieval, User Behavior Modeling and Natural Language Processing.

Shaoping MA is working as a professor in Department of Computer Science and Technology in Tsinghua University. He is the vice president of CAAI and mainly works on Information Retrieval, Pattern Recognition and Knowledge Engineering.