Click Chain Model in Web Search

Fan Guo[1]*, Chao Liu[2], Anitha Kannan[3], Tom Minka[4], Michael Taylor[4], Yi-Min Wang[2], Christos Faloutsos[1]

[1]Carnegie Mellon University, Pittsburgh PA 15213, USA
[2]Microsoft Research Redmond, WA 98052, USA
[3]Microsoft Research Search Labs, Mountain View CA 94043, USA
[4]Microsoft Research Cambridge, CB3 0FB, UK
fanguo@cs.cmu.edu, {chaoliu,ankannan, minka, mitaylor, ymwang}@microsoft.com, christos@cs.cmu.edu

ABSTRACT

Given a terabyte click log, can we build an efficient and effective click model? It is commonly believed that web search click logs are a gold mine for search business, because they reflect users' preference over web documents presented by the search engine. Click models provide a principled approach to inferring user-perceived relevance of web documents, which can be leveraged in numerous applications in search businesses. Due to the huge volume of click data, scalability is a must.

We present the *click chain model* (CCM), which is based on a solid, Bayesian framework. It is both scalable and incremental, perfectly meeting the computational challenges imposed by the voluminous click logs that constantly grow. We conduct an extensive experimental study on a data set containing 8.8 million query sessions obtained in July 2008 from a commercial search engine. CCM consistently outperforms two state-of-the-art competitors in a number of metrics, with over 9.7% better log-likelihood, over 6.2% better click perplexity and much more robust (up to 30%) prediction of the first and the last clicked position.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*retrieval models*

General Terms

Algorithms, Experimentation

1. INTRODUCTION

Billions of queries are submitted to search engines on the web every day. Important attributes of these search activities are automatically logged as implicit user feedbacks. These attributes include, for each query session, the query string, the time-stamp, the list of web documents shown in the search result and whether each document is clicked or not. Web search click logs are probably the most extensive, albeit indirect, surveys on user experience, which can be aggregated over weeks, months and even years. Extracting key statistics or patterns from these tera-byte logs is of much interest to both search engine providers, who could obtain objective measures of user experience and useful features to improve their services, and to world wide web researchers, who could better understand user behavior and calibrate their hypotheses and models. For example, the topic of utilizing click data to optimize search ranker has been well explored and evaluated by quite a few academic and industrial researchers since the beginning of this century (*e.g.* , [2, 8, 14, 15, 17]).

A number of studies have been conducted previously on analyzing user behavior in web search and their relationship to click data. Joachims et al. [9, 10] carried out eye-tracking experiments to study participants' decision process as they scan through search results, and further compared implicit click feedback against explicit relevance judgments. They found that clicks are accurate enough as relative judgement to indicate user's preferences for certain pairs of documents, but they are not reliable as absolute relevance judgement, *i.e.* ,"clicks are informative but biased". A particular example is that users tend to click more on web documents in higher positions even if the ranking is reversed [10]. Richardson et al. [16] proposed the *examination hypothesis* to explain the position-bias of clicks. Under this hypothesis, a web document must be examined before being clicked, and user-perceived document relevance is defined as the conditional probability of being clicked after being examined. Top ranked documents may have more chance to be examined than those ranked below, regardless of their relevance. Craswell et al. [4] further proposed the *cascade model* for describing mathematically how the first click arises when users linearly scan through search results. However, the cascade model assumes that users abandon the query session after the first click and hence does not provide a complete picture of how multiple clicks arise in a query session and how to estimate document relevance from such data.

Click models provide a principled way of integrating knowledge of user search behaviors to infer user-perceived relevance of web documents, which can be leveraged in a number of search-related applications, including:

- **Automated ranking alterations**: The top-part of ranking can be adjusted based on the inferred relevance so that they are aligned with users' preference.
- **Search quality metrics**: The inferred relevance and user examination probabilities can be used to compose

*Part of this work was done when the first author was on a summer internship with Microsoft Research.

WWW 2009, April 20–24, 2009, Madrid, Spain.
ACM 978-1-60558-487-4/09/04.

search quality metrics, which correlate with user satisfaction [6].

- **Adaptive search**: When the meaning of a query changes over time, so do user click patterns. Based on the inferred relevance that shifts with click data, the search engine can be adaptive.
- **Judge of the judges**: The inferred first-party relevance judgement could be contrasted/reconciled with well-trained human judges for improved quality.
- **Online advertising**: The user interaction model can be adapted to a number of sponsored search applications such as ad auctions [1, 11].

An ideal model of clicks should, in addition to enabling reliable relevance inference, have two other important properties - *scalability* and *incremental computation*; Scalability enables processing of large amounts (typically, terabytes) of clicklogs data and the incremental computation enables updating the model as new data are recorded.

Two click models are recently proposed which are based on the same examination hypothesis but with different assumptions about user behaviors. The user browsing model (UBM) proposed by Dupret and Piwowarski [5] is demonstrated to outperform the cascade model in predicting click probabilities. However, the iterative nature of the inference algorithm of UBM requires multiple scans of the data, which not only increases the computation cost but renders incremental update obscure as well. The dependent click model (DCM) which appears in our previous work [7] is naturally incremental, and is an order of magnitude more efficient than UBM, but its performance on tail queries could be further improved.

In this paper, we propose the *click chain model* (CCM), that has the following desirable properties:

- *Foundation:* It is based on a solid, Bayesian framework. A closed-form representation of the relevance posterior can be derived from the proposed approximation inference scheme.
- *Scalability:* It is fast and nimble, with excellent scalability with respect to both time and space; it can also work in an incremental fashion.
- *Effectiveness:* It performs well in practice. CCM consistently shows performance improvements over two of the state-of-the-art competitors in a number of evaluation metrics such as log-likelihood, click perplexity and click prediction robustness.

The rest of this paper is organized as follows. We first survey existing click models in Section 2, and then present CCM in Section 3. Algorithms for relevance inference, parameter estimation and incremental computation are detailed in Section 4. Section 5 is devoted to experimental studies. The paper is concluded in Section 6.

2. BACKGROUND

We first introduce definitions and notations that will be used throughout the paper. A web search user initializes a *query session* by submitting a *query* to the search engine. We regard re-submissions and reformulations of the same query as distinct query sessions. We use *document impression* to refer to the *web documents* (or URLs) presented in the first result page, and discard other page elements such as sponsored ads and related search. The document impression

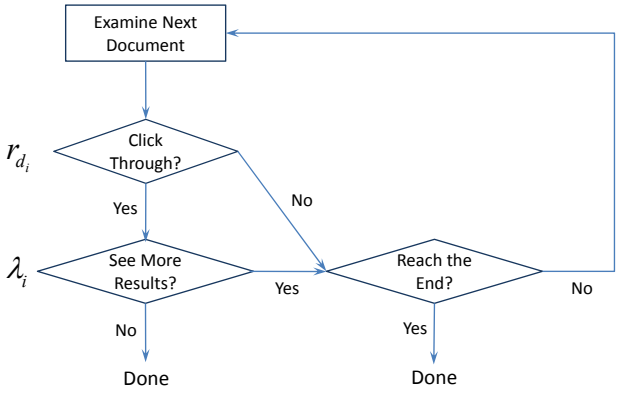

Figure 1: The user model of DCM, in which r_{d_i} is the relevance for document d_i at position i, and λ_i is the conditional probability of examining the next position after a click at position i.

can be represented as $D = \{d_1, \ldots, d_M\}$ (usually $M = 10$), where d_i is an index into a set of documents for the query. A document is in a higher *position* (or rank) if it appears before those in lower positions.

Examination and clicks are treated as probabilistic events. In particular, for a given query session, we use binary random variables E_i and C_i to represent the examination and click events of the document at position i,respectively. The corresponding, examination and click probabilities for position i are denoted by $P(E_i = 1)$ and $P(C_i = 1)$, respectively.

The *examination hypothesis* [16] can be summarized as follows: for $i = 1, \ldots, M$,

$$P(C_i = 1|E_i = 0) = 0,$$
$$P(C_i = 1|E_i = 1) = r_{d_i},$$

where r_{d_i}, defined as the *document relevance*, is the conditional probability of click after examination. Given E_i, C_i is conditionally independent of previous examine/click events $E_{1:i-1}, C_{1:i-1}$. It helps to disentangle clickthroughs of various documents as being caused by position and relevance. Click models based on the examination hypothesis share this definition but differ in the specification of $P(E_i)$.

The *cascade hypothesis* in [4] states that users always start the examination at the first document. The examination is strictly linear to the position, and a document is examined only if all documents in higher positions are examined:

$$P(E_1 = 1) = 1,$$
$$P(E_{i+1} = 1|E_i = 0) = 0.$$

Given E_i, E_{i+1} is conditionally independent of all examine/click events above i, but may depend on the click C_i.

The *cascade model* [4] puts together previous two hypotheses and further constrain that

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

which implies that a user keeps examining the next document until reaching the first click, after which the user simply stops the examination and abandons the query session.

We first introduce the *dependent click model* (DCM) [7]. Its user model is illustrated in Figure 1. It generalizes the cascade model to multiple clicks by putting position-dependent parameters as conditional probabilities of examining the next