On the Value of Page-Level Interactions in Web Search

Jeff Huang
Information School
University of Washington
hcir@jeffhuang.com

ABSTRACT

Users' search data has been useful for understanding search behavior and has been applied to improve web search. Query logs, the primary source of search data in recent information retrieval research, are limited in expressing the user's behaviors; they omit behaviors that do not hit the web server: cursor movements, scrolling, browser tab usage, text highlighting, and duration of a pageview. These actions we call 'page-level interactions'; they can be collected by search systems using client-side scripting, but this is currently not being done by web search engines. Previous studies in related fields have shown that adding additional independent data provides greater improvements than smarter algorithms. Since page-level interaction data is independent from query and click data, collecting and mining page-level interactions may be one direction that search engines can pursue to innovate. These interactions can supplement query logs by helping understanding user intent, disentangling interleaved queries, or providing richer user data for rare queries; they can be particularly useful especially when clicks for queries are unavailable or insufficient.

1. INTRODUCTION

People conduct billions of web searches every day. Search engineers and information retrieval researchers have long since used search data to understand how users search the web at scale. The data that search companies collect about user behaviors can be fed back into the search system to improve itself. Search data can be used for analytics, studying the users themselves and their input into the system to better understand their users and what they are using the search engine for, or applied to infer new knowledge that help in later search sessions.

The most common form of search data is the query logs of search systems that record queries and clicks. Query logs containing this data have been shown to be useful for a multitude of applications [13]. These logs used to personalize search can target search results to the preferences of the current user; they can be used to detect spam web pages; they can help with query formulation tasks such as spelling correction, autocomplete, and offer query suggestions.

Another application is to help rank search results, where ranking signals in web search can be classified into three categories: content signals, structural signals, and web usage [1]. The first search engines drew from techniques in traditional information retrieval, scoring how well each web page's content matched the user's query using algorithms such as tf-idf. As web search engines evolved, signals from the structure of the web gained importance, supplementing content signals through techniques such as mining the links between web documents. Lately, mining the query logs (web usage) has improved search by treating user interactions like clicks as implicit relevance feedback.

While query logs are the most direct and easily understandable, they can still be further enriched with additional search data since they are limited for understanding the user: they are unable to reveal actual user intent, provide little data about uncommon queries, and omit many interactions. Fortunately, there is an uncollected, unobserved dataset that is potentially even richer and larger than the search data in query logs. In this paper, we argue that mining *page-level interactions* can address some of the query logs' limitations. These are user interactions that do not directly hit the search engine's server. Page-level interaction data is created by search pages instrumented with client-side scripting to record user activity on the search engine's web pages. Through this, search engines can record cursor movements, tab usage, and dwell time. For example, a website operator can determine the order in which a user fills out forms on the page, and how long each field takes to enter. Recording a user's cursor trail and hesitations following a query can provide signals for relevance [8, 5], especially when click data is unavailable.

Studies in related fields of natural language processing [2, 6] and data mining [12] have observed that collecting more data gains importance over improving search algorithms. This suggests that search data may continue to be the source of search improvements for the near future, and new independent data such as page-level interactions continue along this front, since unobtrusive techniques in page-level interaction logging may gather more complete user search data.

In this paper, we discuss the potential for collecting different types of page-level interactions, and next steps that would lay the foundation for this work. First, we examine the current sources of user data from recent information retrieval research and list the inherent limitations of these methods. We then introduce different types of page-level interactions, how they supplement query logs, and why they support search engines in improving search ranking and user assistance features. Finally, we describe several promising areas of research in mining page-level interactions.

2. USER DATA SOURCES IN IR STUDIES

We can count the sources of user data in existing information retrieval research. Table 1 tallies existing data sources for understanding user behavior from SIGIR, WSDM, and the CIKM IR Track over the years 2009 and 2010. To find relevant papers about user behavior, we look at papers in conference sessions containing the word 'user', or paper titles containing the word 'user' or 'search'. Still, not all papers filtered using these words were actually about user behavior in web search, so we culled judiciously.

Data Source	Frequency
Query logs	24
Lab study	9
Toolbar logs (Microsoft)	8
Crowdsourced (Pagehunt, Mechanical Turk)	2
Browsing logs (Nielsen)	1
Survey	1
Field study (workplace)	1
Learning to rank (numerical data)	1

Table 1: A sample of sources for user data in information retrieval studies from 2009 and 2010.

User behavioral data is typically sourced from query logs, toolbar logs, and lab studies, with query logs more common than the rest combined. The above studies do not include sessions and workshops specifically focused on query logs which would increase the proportion of query log as user data, e.g. the Workshop on Web Search Click Data from WSDM 2009, the session on Query Analysis and Feedback from CIKM 2010, and the session on Query Log Analysis from SIGIR 2010. What current work is missing is the usage of enriched query logs, i.e. query logs with more than the typical fields.

Search companies have been deploying toolbars that users can install (e.g. the toolbars from Google, Bing, Yahoo, Ask, and AOL) and proprietary web browsers that collect browsing data. These applications are able to record similar information as query logs, but there has not been any evidence that search companies have used their toolbars to record detailed page-level interactions on web search pages.

Lab studies involve asking users to enter a lab setting to perform an artificial task. They are useful for qualitatively understanding the user, but mining search data is a more scalable way to automatically improve some search engine attributes like ranking and user assistance features.

2.1 Query Log Specific Limitations

Query logs are low-hanging fruit, since they typically already exist as web server access logs and do not require substantial modification to the search engine, This explains their popularity, but query logs have inherent limitations; some have been noted in the literature [4, 10], but here we discuss some that are specific to query logs and may be resolved using other search data.

Query and click data is sparse for rare queries, which comprise a significant portion of all queries. Since data is only recorded in query logs when the user performs an action that hits the server, a number of useful signals are lost. A rare query might not have any clicks, and thus the query logs can teach little about the relevance of the results.

Modern web browsers support tabs and multiple windows, which allow users to navigate multiple web pages simultaneously. One or more pages can be opened from a single page; browsing flow within a single tab/window may then be interrupted by switching to pages in other tabs/windows. Since the web server is agnostic to which tab or window is active, and will record clicks and queries only in the order they were received. Therefore, switching to a different page to click or opening multiple tabs from a single search result page confuses algorithms that process query logs; it is a challenge to understand interleaved sessions in these logs. Essentially, there are two problems: 1) interleaved sessions and 2) loss of detail about branching behavior (when users open links in new tabs or windows).

New interface features such as Bing's thumbnail page preview or Google Instant change the interaction from a sequential process of 'enter query, review results, click result' to a more dynamic interactive process centered on client-side interaction. Information mined from query and click logs will lack behaviors from these new interface features.

2.2 More Data Beats Better Algorithms

In information retrieval sister fields of natural language processing and data mining, there has been evidence that collecting and mining additional data can be more useful than improving algorithms. A study by Banko and Brill [2] looked at various learning algorithms for disambiguating natural language. They showed that increasing the amount of data by 10-fold would make even the worst algorithm better than the best algorithm. A recent article published by Google Researchers, Alon Halevy, Peter Norvig, and Fernando Pereira, titled "The Unreasonable Effectiveness of Data" [6] highlights the power of web-scale data for machine translation. Natural language involves the concept of context, which algorithms have trouble understanding. However, with enough input, machine translation and speech recognition can be accomplished statistically. Stanford data mining instructor Anand Rajaraman's article titled, "More data usually beats better algorithms" [12], presents anecdotal evidence arguing that, "adding more, independent data usually beats out designing ever-better algorithms to analyze an existing data set". In one example, students in his class competed to recommend Netflix movies given a set of previously rated movies from users, a typical machine learning classification problem. The team that applied a simple algorithm but combined IMDB data with the Netflix data performed much better than the team that applied a sophisticated algorithm on just the Netflix data.

More pertaining to information retrieval, Kohavi et al. recall the power of supplementing Amazon's product search with user data [10],

> ...searches, such as "24", which most humans associated with the TV show [...]. Amazon's search was returning poor results, [...] such as CDs with 24 Italian Songs, clothing for 24-month old toddlers, a 24-inch towel bar, etc. (These results are still visible on Amazon today if you add an advanced search qualifier like "-foo" to the search phrase since this makes the search phrase unique and no mappings will exist from people who searched for it to products.) The behavior-based-search (BBS) algorithm gave top-notch results with the DVDs of the show and with related books, i.e., things that people purchased after searching for "24"...

Having search data at scale is essential because it provides good coverage over the queries (which are known to have a long tail), and allow for stratification over variables such as geography, task type, topic, and user type. Web search already benefits by adding more of the same data—more query logs are generated every second, creating a near-infinite source of temporal data, and thereby improving the accuracy of inferences and analyses made from the data; this is akin to adding more web pages to the index in the early

days of ranking using content signals. However, there may be further benefit from adding more different data, akin to using structural signals to supplement content signals for ranking, a breakthrough in search ranking quality. New independent data can answer new sets of questions and provide information that cannot be inferred from existing data.

What can be better than query logs—a large natural source of search data? One answer is by adding even finer grained user data reflecting every minute action by the user as they are searching. Query logs are nearing their full potential and there is an upper bound in the information they can provide, but we have not yet scratched the surface of mining page-level interactions.

3. PAGE-LEVEL INTERACTIONS

Page-level interactions can be recorded with minimal intrusion using JavaScript, which is built into all modern web browsers. Small snippets of JavaScript can record page-level interactions and quietly send the data back to the search system using a common trick of sending the data in the URL of an image GET request. The following list describes some page-level interactions that can be captured.

Cursor Activity Cursor movements and hesitations can be recorded using JavaScript at fine levels of detail. Some users move their cursor over text as they read, move the cursor slowly when thinking, or toss aside the cursor to reveal content that the cursor was obscuring. Collecting cursor data can be used to diagnose usability issues at an individual level when the session is replayed; furthermore, the data can be analyzed in aggregate as heatmaps. Other cursor activity such as scrolling, highlighting text, or non-navigational clicks may be explored. We have seen during preliminary observations that some users clicked on whitespace or text as they focused on that region. Scrolling was also common and may indicate dissatisfaction with the initial view of results. Knowing the user's current viewport approximates the examination region [11].

Parallel browsing behavior People can use multiple tabs or new windows simultaneously while searching. Users have been found to switch tabs in over 57.4% of browsing sessions [7]. This is important since interleaved browsing sessions cause problems for analyzing query logs. To disentangle this behavior, client-side scripts on the search page can record which pages are open, length of time opened, and whether a 'new tab or window' action was used on a certain link..

Web browser metadata There are a number of variables available to JavaScript which may help enrich the profile of the user. For example, the browser window resolution can be a factor that determines how content is laid out on the search pages. The user agent, operating system, timezone can show up in server logs but they rarely influence the search interface because this requires an additional http request. However, with server push technology such as HTML5 WebSockets, it may now be practical to update the displayed interface from browser metadata.

Accurate dwell time measurements Dwell time can be accurately measured on search pages by sending a timestamp to the server when the web page is closed and possibly when the page has user focus. In contrast, query logs record nothing when the page is closed, so dwell time is typically calculated by subtracting consecutive events in the log. This confounds the meaning of dwell time by not differentiating between active time (when the user is reading and interacting with the page) to passive time (when the user may be focusing on a different page).

← Time	
	Query: lady gaga concert tickets
	Cursor moves from top to hover over 3rd search result
	Cursor pauses for 3 seconds
	Text "Tour Dates Only" is highlighted with the cursor
	Cursor moves to the 4th search result, pausing 1s
	User scrolls to the 5th search result, pausing 3s
	Cursor returns to the 4th search result and clicks
	Click: Result 4 [http://gaga.com/tix/]

Table 2: A user searches for "lady gaga concert tickets", examines the first page of results, and clicks the 4th search result. Typical query logs contain query and click data (bold), but no page-level interactions.

Compared to query logs, page-level interactions provide information about behaviors that do not necessarily hit the web server. Page-level interactions such as hovering over certain portions of the page or cursor reading behavior (via highlighting or back-and-forth movements), can in fact give indications of relevance. Although the data may be noisy and inferring meaning may pose a challenge, one can imagine how an oracle watching every cursor move, window scrolling, and tab usage, all with corresponding dwell times, would know substantially more about a user's intentions than an oracle who only sees a user's queries and clicks.

Compared to toolbar logs, page-level interaction data is not biased by users self-selecting to install additional software. While toolbars and browser plugins are typically applied to web browsing in general, page-level interaction logging is implemented on the search pages themselves, so they can be tailored to search-specific components such as recording the rank of the search result the cursor has crossed, or whether the cursor hovered over a search advertisement.

Table 2 presents fictional searches along with the corresponding query logs and page-level interaction logs. In this and many other cases, the page-level interaction data reveals substantially more information about the user's intent. In the above scenario, the query logs show a query was made, and that some time later, the 4th result was clicked. This is useful information, but the page-level interaction data supplements this by showing the user was active the whole time examining several results, that the user likely examined the 5th result and returned back to the 4th result, indicating the 1-3 and 5th results may have been less relevance than the 4th result. The highlighted text also suggests that the user is interested in information surrounding tour dates, and that this has something to do with the concert tickets.

Recording the amount of time spent on the search results page is another scenario that may help determine user intent. A combination of scrolling and a long dwell time but no clicks may indicate that the user did review the entire page of search results but may have been dissatisfied with the relevance of the results and decided to abandon the search. Corresponding query logs would show very little information—only that the query was issued.

4. PROMISING AREAS OF RESEARCH

4.1 Using Cursor Behavior Data

Performance is a concern since cursor data must first be collected by the web browser and transferred to the search system. Tracking the cursor and sending the data over the network may slow down the user's computer if implemented inefficiently. While existing work has reported one efficient approach for tracking cursor movements [8], future research can explore methods for summarizing cursor activity that keeps the essential features of the cursor behavior but can also be collected at large scale. There may be approaches for sampling cursor movement entailing approximation or identifying sub-movement boundaries [9]. An important next step is exploring efficient methods to tune the trade-off between performance and data granularity.

Determining how to separate the signal from the noise is a difficult problem with cursor movements. Some movements may be unintentional while others are ambiguous. For example, users will move a cursor to whitespace simply to get it out of the way from reading; however, they will also move their cursor to areas of interest. Potential algorithms could understand higher-level abstractions such as reading behavior, differentiating pauses of interest vs. stepping away from the computer, and personalizing the analysis to account for users' habits.

While hovering over links and cursor movement speed correlates with relevance and abandonment reasons [8], one useful application of behavior mining is to use cursor data for directly improving search result ranking. Learning to rank from cursor behavior requires methods to separate the signal from noise, but a simple initial step may be to apply cursor data to click models (e.g. [3]) to uncover the latent variables to increase the models' accuracy. We can also gain a better understanding of result examination behavior, since cursor has been shown to correlate with gaze [8], and this may improve the design of search interfaces.

4.2 Using Parallel Browsing Data

Being able to record how long a user keeps a web page open along with whether a link was clicked gives us the ability to disentangle browsing threads. Another piece of useful information that can be gathered by scripting is whether the current page has the user's focus at the time. Together, these pieces of information can reveal how long the user has spent on the search results page, whether they opened a new window or tab, and when or if they returned to the search results page. Interleaved sessions can be completely disentangled, and session boundaries can be detected more accurately.

These data that describe the parallel browsing behavior of a user can also be used to better model their search result examination behavior. In a number of click models (e.g. [3]), there are hidden states representing whether a user continued examining search results after clicking. Having accurate information about a user's focus can improve these models. Additionally, knowing whether a user opened a search result in new tabs or windows (i.e. branching) can provide a different signal than if they simply clicked. For example, branching may cause users to click results in a sequential top-to-bottom order, opening whatever is interesting, but clicking may be a different strategy of finding the most interesting search result.

5. CONCLUSIONS

Little work has been done to enrich query logs as the source of analyzing user behavior in web search engines. These query logs are limited in the interactions they record, giving them a theoretical upper bound of what they can teach. Studies in related fields have shown that having more independent data is the best way to improve system quality. Page-level interactions, which are behaviors on web search pages that do not hit the web server, are a useful supplemental source of search behavior data. They can improve understanding of the user and their search process, and allow search engines to improve their ranking and user assistance features. We believe this will be an important area of research in the near future, as having more independent data beyond query logs will be a promising area for improving search.

6. REFERENCES

[1] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., 2nd edition, 2011.

[2] M. Banko and E. Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of ACL*, pages 26–33, 2001.

[3] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *Proceedings of WWW*, pages 1–10, 2009.

[4] C. Grimes, D. Tang, and D. M. Russell. Query logs alone are not enough. In *WWW Workshop on Query Log Analysis*, 2007.

[5] Q. Guo and E. Agichtein. Ready to buy or just browsing?: detecting web searcher goals from interaction data. In *Proceedings of SIGIR*, pages 130–137, 2010.

[6] A. Halevy, P. Norvig, and F. Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8 –12, 2009.

[7] J. Huang and R. W. White. Parallel browsing behavior on the web. In *Proceedings of Hypertext*, pages 13–18, 2010.

[8] J. Huang, R. W. White, and S. Dumais. No clicks, no problem: using cursor movements to understand and improve search. In *Proceedings of CHI*, pages 1225–1234, 2011.

[9] R. Jagacinski, D. Repperger, M. Moran, S. Ward, and B. Glass. Fitts' law and the microstructure of rapid discrete movements. *J. Exp. Psychol. [Hum. Percept.]*, 6(2):309–320, January 1980.

[10] R. Kohavi, R. Longbotham, D. Sommerfield, and R. Henne. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery*, 18(1):140–181, 2009.

[11] D. Lagun and E. Agichtein. Viewser: Enabling large-scale remote user studies of web search examination and interaction. In *Proceedings of SIGIR*, pages 365–374, 2011.

[12] A. Rajaraman. More data usually beats better algorithms. `http://anand.typepad.com/datawocky/2008/03/more-data-usual.html`, 2008.

[13] F. Silvestri. Mining query logs: Turning search usage data into knowledge. *Found. Trends Inf. Retr.*, 4(1-2):1–174, January 2010.