Comparing Client and Server Dwell Time Estimates for Click-Level Satisfaction Prediction

Youngho Kim[1]*, Ahmed Hassan[2], Ryen W. White[2], and Imed Zitouni[2]
[1] University of Massachusetts, 140 Governors Drive, Amherst, MA 01003, USA
[2] Microsoft, One Microsoft Way, Redmond, WA 98052, USA
yhkim@cs.umass.edu, {hassanam, ryenw, izitouni}@microsoft.com

ABSTRACT
Click dwell time is the amount of time that a user spends on a clicked search result. Many previous studies have shown that click dwell time is strongly correlated with result-level satisfaction and document relevance. Accurate estimates of dwell time are therefore important for applications such as search satisfaction prediction and result ranking. However, dwell time can be estimated in different ways according to the information available about the search process. For example, a result reached for the query [Garfield] may involve 145s of "server-side" dwell time (observable to the search engine) and 40s of "client-side" dwell time (observable from the browser). Since search engines can only observe server-side actions (i.e., activity on the search engine result page), server-side dwell times are estimated by measuring the time between a search result click and the next search event (click or query). Conversely, more detailed information about page dwell times can be obtained via client-side methods such as Web browser toolbars. The client-side information enables the estimation of more accurate dwell times by measuring the amount of time that a user spends on pages of interest (either the landing page, or pages on the full navigation trail). In this paper, we define three different dwell times, i.e., server-side, client-side, and trail dwell time, and examine their effectiveness for predicting click satisfaction. For this, we collect toolbar and search engine logs from real users, and provide an analysis of dwell times for improving prediction performance. Moreover, we show further improvements in predicting click-level satisfaction by combining dwell times with other query features (e.g., query clarity).

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Search Process.

Keywords: Dwell time analysis; Click satisfaction.

1. INTRODUCTION
Click dwell time (the time that the user spends on a clicked search result) is one of the most important implicit measures for improving web search quality [5][15]. Previous work found that this feature is strongly correlated with result-level satisfaction and document relevance [5]. To identify satisfied (SAT) clicks, longer dwell time on a clicked page has been considered as a positive signal, e.g., a click is regarded as SAT if its dwell time equals or exceeds 30 seconds [5]. Since click dwell time is implicitly measured by browsing actions related to web search (e.g., search result clicks, returning to the previous page, etc.), dwell time can be estimated in different ways according to what search action information is available. Henceforth, we use the terms "dwell time" and "click dwell time" interchangeably to refer to the time spent on a search result.

Given server-side logs of user search activity, dwell time is generally estimated by measuring the difference between the time stamps of different search actions. For example, click dwell time is estimated as the amount of time between a click on a search result and the next action (query, click, etc.) on the search engine result page (SERP). We refer to this dwell time as the *server-side dwell time*. Server-side dwell time is typically an overestimate of dwell time since we have no means of determining what activity the searcher has been engaged in between observable interactions with the search engine. On the other hand, previous work used client-side applications such as toolbars or browser add-ins (e.g., [1][5][13]) to estimate the actual time spent on the clicked page. We refer to this dwell time as *client-side dwell time*. Since some clicks act as *waypoints* [16] that direct searchers to more relevant information via links to other pages, we can exploit search trail information (i.e., a sequence of page visits starting with a search-result click). This is important since landing pages serving as a gateway to relevant content may have high utility for the current task but a short dwell time if the user clicks a link on that page quickly [1]. Therefore, dwell times of the clicks following the landing page are also included in the dwell time of the initial click. We refer to this as *trail dwell time*.

Although client-side and trail dwell times are quite accurate, they are not easily attainable since only a limited number of search users install client-side applications and provide explicit consent to share their data with search providers. Alternatively, server-side search activity is easily attainable but less accurate since it cannot account for unrelated actions (e.g., interleaving search with checking e-mails or browsing other non-relevant pages).

In this paper, we compare three different ways of estimating dwell time: 1) server-side, 2) client-side, and 3) trail dwell times. We analyze each dwell time by examining its effectiveness in predicting click-level search satisfaction. To do this, we first collect click instances from submitted to a commercial search engine, and label each instance as SAT (satisfied) or DSAT (dissatisfied). We then estimate the three dwell times for every instance (the details are described in Section 3). Given these estimates, we construct binary classifiers which assign a click instance into {SAT, DSAT} using the dwell times as features. In addition, we combine dwell time features with query performance features (e.g., query clarity [3]) since they have been shown to be effective for the SAT/DSAT classification [12]. We examine whether our dwell-time-based features can lead to further improvements over the query performance features alone.

The main contribution of our research is in providing a detailed analysis on the efficacy of three different mechanisms for estimat-

* This work was conducted while interning at Microsoft Research

SIGIR '14, July 06–11 2014, Gold Coast, QLD, Australia

http://dx.doi.org/10.1145/2600428.2609468

ing landing page dwell times. Additionally, we utilize these dwell time variants for predicting click-level satisfaction and study the relative utility of server-side, client-side, and trail dwell times for this task. To the best of our knowledge, this is the first attempt to study and compare the various dwell times estimated by server-side and client-side search information.

2. RELATED WORK

Previous work has extensively studied the utility of implicit feedback (e.g., dwell time, scrolling, clicks) as a relevance estimator, e.g., [1][10]. These studies have found that dwell time is an important feature for inferring search relevance. In addition, other researchers have analyzed dwell time to understand browsing behavior or predict click-level satisfaction (e.g., [12][13]). Liu et al. [13] applied Weibull analysis to estimate the underlying distributions of given dwell time data, and more recently Kim et al. [12] proposed a method to model dwell time data based on query-click attributes such as query type, the reading difficulty of clicked pages, and the topical classification of those pages.

Research on modeling search satisfaction based on search activity such as clicks and their dwell times is also relevant (e.g., [4][5][7]). Hassan et al. [7] proposed user behavior models for predicting task-level search success. They recognized behavior patterns from labeled data, and built Markov models to measure the likelihood of each pattern. Moreover, Kim et al. [12] used discriminative models which combine various satisfaction features such as dwell time, topics on clicked pages, and query performance predictors [3][8]. Given the association between search activity and satisfaction modeling, we utilize satisfaction prediction as the application domain for our experiments with different dwell time estimation methods.

3. DATA ACQUISITION

Our click data was sampled from the logs of consenting users of a browser toolbar distributed by the Microsoft Bing search engine. The logs contained web search activities (e.g., queries submitted, subsequent result clicks, etc.). The logs were sampled from the U.S. English locale to reduce geographic and linguistic variations in search behavior. Access to these logs enabled the computation of client-side and trail dwell times on clicked search results. Each log entry contained search actions including query submission, search result clicks (i.e., SERP clicks), revisits to the SERP, and other result clicks (e.g., clicks following the search result click). Accordingly, the timestamp of each logged action is recorded, and using this timing information, we can estimate the server-side, client-side, and trail dwell times for each SERP click as follows:

Server-side dwell time: A server-side dwell time is calculated by measuring the time between a SERP click and the next interaction with the search engine (query or click). If the user is interleaving search with other tasks (e.g. checking email), we would not be able to determine how much time was actually spent on the clicked page (vs. on pages related to other tasks). Also if the user never returns to the engine, we would not be able to compute dwell time.

Client-side dwell time: A client-side dwell time is measured by estimating the time between the click and the page unload time (i.e. closing the page or navigating to another page). Note that the client-side dwell time, while more accurate than the server-side dwell time, does not take into consideration of periods where the page is open but the user is distracted and is not actively attending to the page.

Trail dwell time: Given a SERP click, we assume that its search trail consists of the clicked pages and all subsequent page visits originating from clicks on hyperlinks on the clicked page or other pages on the trail (i.e., pages on the trail had to be linked, conveying relatedness). For each trail page, we estimate dwell time by measuring the client-side dwell time as explained above and then we sum all trail dwell times to compute the overall dwell time for the trail.

After measuring dwell times per these three strategies, we randomly sampled 7,500 queries and their clicks from the initial data, and employed human assessors to label each click as either satisfied or not. The set of clicks chosen had dwell times estimated by all three of the methods described earlier. Judges were shown the query, the clicks to be labeled, along with the previous and the next query to help them understand the search context. Note that timestamps and any information related to dwell times was hidden from judges to avoid biasing their labels. For every click, the judges were asked to examine the query and the content of the clicked URL and then rate user satisfaction with the click on a five-point scale with the following response options: *none*, *slight*, *moderate*, *high*, *perfect*. To derive a binary satisfaction score from these multi-point ratings, clicks labeled as *high* or *perfect* were considered satisfied (SAT); otherwise, they were labeled as dissatisfied (DSAT).

In order to obtain a ground truth satisfaction estimate for each click, we first collected the judgment results from two annotators. If the two annotators agreed on the binary label, then we used that label. If they disagreed, we requested that another judge label the instance and we use the majority label among the three judges as the satisfaction estimate for the click. Once every click was judged, we observed that 82.8% of all clicks were labeled as SAT. Using this distribution of labels directly would have caused the statistical model trained by this skewed data to be biased to the majority class (i.e., SAT) and less effective when making predictions for the minority class (i.e., DSAT). To address this concern, we generated a 50/50 balanced dataset by randomly down-sampling the SAT class. This resulted in a final data set used for our analysis that contained 3,204 click instances: 1,602 instances of each type.

4. CLASSIFICATION MODEL

The classification model to predict SAT click instances is defined as follows. Given a set of n labeled examples, $\{x_1, x_2, \ldots, x_n\}$, we generate a feature vector of each x_i, i.e., $f(x_i)$, and the label of each x_i is indicated by $l(x_i) \in \{\text{SAT}, \text{DSAT}\}$. A set of training examples is defined as $\text{T} = \{\langle f(x_i), l(x_i)\rangle\}_{i=1}^{n}$, and the classification function maps a feature vector to a SAT or DSAT label. This model is learned by minimizing the disagreement between a mapped label and original label for every training instance.

4.1 Features

The features used for the classification model are described as follows. We use the three dwell time estimates directly as features in the model. In addition, we leverage other features from previous work to characterize the query itself [3][8]. This is intended to allow the classifier to treat dwell time differently for different types of search queries. More specifically, we chose to use the query clarity score [3], inverse collection term frequency [8], and query term length as features in our models. To estimate inverse collection term frequency, we exploit term probabilities obtained from the web n-gram services [14]. For those query-term frequency features, we calculate the sum, standard deviation, ratio of the maximum to the minimum, maximum, arithmetic mean, and geometric mean among the term probabilities of all query terms. Table 1 summarizes the classification features that we use to generate a feature vector. In experiments, we analyze the effectiveness of each dwell time feature as well as query performance predictors.

Table 5: Classification features.

Category	Feature
Click dwell time	Server-side dwell time Client-side dwell time Train dwell time
Query performance predictor	Query clarity [3] Inverse collection term frequency Query term length

5. EXPERIMENT

We conduct experiments to analyze server-side, client-side, and trail dwell times. For this, we develop the click classification model described in Section 4, and examine the effectiveness of dwell times as features for the classification. We first describe how to set up the experiments and then provide their results.

5.1 Experimental Set-up

As described in Section 3, our click data contains 3,204 labeled instances (50% SAT and 50% DSAT). Our objective is given a query, an associated result click, and different estimates of dwell time on that clicked page, predicting whether the click is satisfied (SAT) or dissatisfied (DSAT). For our learning algorithm, we used Gradient Boosted Decision Trees (GBDT) [6], and performed 10-fold cross-validation using random partitioning. Our experiments revealed that the results are fairly insensitive to the choice of learning algorithm. For evaluation purposes we measure Precision, Recall, and F1-score for each class. In addition, accuracy (i.e., weighted precision) is used to compute overall classification performance.

5.2 Classification Results

To analyze the three types of dwell times, we first perform the classification using only a single dwell time feature. Table 2 shows the results from this task. From the table it is clear that trail dwell time performs significantly better than the others in many metrics (e.g., accuracy and DSAT F1). Trail dwell time seems to more accurately capture the true utility that searchers derived from the click, and hence their satisfaction with the click. We also notice that the client-side dwell time estimate performs poorly and is out-performed by the server-side dwell time estimates. We believe that server-side dwell time performs better because it is a better estimate of the dwell time of the entire search trail when compared to client-side dwell time which only considers the time spent on the clicked page, independent of any follow-on activity by the searcher.

Interestingly, client-side dwell time performs better only in terms of DSAT recall. We believe that client-side and trail dwell times are better than server-side dwell time in terms of DSAT recall because typically dissatisfied pages are less likely to lead to a long trail and more likely to result in the user returning to the SERP to rewrite the query, and then click on another result or abandon the search. Since client-side instrumentation is more likely to result in a better estimate of the time spent of the clicked page only, we notice that it outperforms server-side estimates for DSAT recall. In other words, for satisfied clicks what matters is not only the time spent on the click but also the time spent on all page visits originating from it resulting in a better performance for trail and server-side dwell time estimates. Conversely, for dissatisfied clicks the accurate dwell time of the click only is more important resulting in better performance for both the trail- and client-side estimates of dwell time.

Table 3 depicts the mean average and standard deviation for each of the dwell time estimates over the click data used in our study. From Table 3, we can observe that the average difference between SAT and DSAT client-side dwell times is much smaller (i.e., 3.58 vs

Table 1: Classification results using a single dwell time feature. In each row, a significant improvement is denoted by the first letter of each feature, e.g., ^{T} indicates an improvement over "Trail", and the Wilcoxon rank-sum test is performed with $p < 0.05$. P, R, and F1 denote Precision, Recall, and F1-score, respectively. The best result is marked by bold.

Metric	Server	Client	Trail
Accuracy	0.5682 ^{C}	0.5091	**0.6007** ^{S,C}
SAT P	0.5758 ^{C}	0.5189	**0.6378** ^{S,C}
SAT R	**0.5180** ^{C,T}	0.2521	0.4664 ^{C}
SAT F1	**0.5453** ^{C}	0.3392	0.5387 ^{C}
DSAT P	0.5620 ^{C}	0.5060	**0.5792** ^{S,C}
DSAT R	0.6184	**0.7662** ^{S,T}	0.7351 ^{S}
DSAT F1	0.5888	0.6095 ^{S}	**0.6480** ^{S,C}

Table 2: Dwell time statistics. Diff. indicates the difference between SAT and DSAT mean average dwell times.

Dwell Time Type	Class	Mean	Diff.	Std. Dev.
Server	SAT	264.82	+86.27	619.89
	DSAT	178.55		320.94
Client	SAT	71.52	+3.58	168.95
	DSAT	67.94		187.65
Trail	SAT	287.02	+132.24	892.22
	DSAT	154.78		543.16

Table 3: Classification results using dwell time combinations. For example, S+C denotes the classification using server-side and client-side dwell times. In each row, a significant improvement is marked by the column order of each feature combination, e.g., ^{23} indicates a significant improvement over "C+T" and "T+S" (the Wilcoxon rank-sum test with $p < 0.05$). The best result is marked by bold.

Col. No.	1	2	3	4
Feature	**S+C**	**C+T**	**T+S**	**S+C+T**
Accuracy	0.5696	0.6531 ^{13}	0.6326 ^{1}	**0.6722** ^{123}
SAT P	0.5760	**0.6924** ^{13}	0.6438 ^{1}	0.6911 ^{13}
SAT R	0.5274	0.5510 ^{1}	0.5934 ^{12}	**0.6228** ^{123}
SAT F1	0.5506	0.6136 ^{1}	0.6176 ^{1}	**0.6552** ^{123}
DSAT P	0.5643	0.6272 ^{1}	0.6229 ^{1}	**0.6567** ^{123}
DSAT R	0.6119	**0.7552** ^{134}	0.6717 ^{1}	0.7215 ^{13}
DSAT F1	0.5871	0.6852 ^{13}	0.6464 ^{1}	**0.6876** ^{13}

Table 4: Classification results using query performance predictors and dwell time features. Query Performance contains query performance predictors (see Table 1). Dwell Times use all server-side, client-side, and trail dwell times. In each row, a * denotes a significant improvement over "Query Performance".

Feature	Query Performance	Query Performance + Dwell Times
Accuracy	0.6343	0.7520 *
SAT P	0.6402	0.7635 *
SAT R	0.6205	0.7301 *
SAT F1	0.6302	0.7465 *
DSAT P	0.6287	0.7415 *
DSAT R	0.6480	0.7738 *
DSAT F1	0.6382	0.7573 *

132.24), which means that client-side dwell time might be less discriminative as a classification feature.

Another interesting observation is that the mean server-side dwell time is similar to the mean trail dwell time (see Table 3). This further explains why server-side dwell time outperforms client-side dwell time in Table 2. In addition, we believe that server-side dwell time is often over-estimated but it is more effective in predicting SAT clicks than client-side dwell time since it captures activity on the navigation trail beyond the clicked page.

Next we conduct experiments by grouping together different types of dwell time features (e.g., using both server-side and client-side dwell times). Table 4 presents the results of these experiments. The addition of each dwell time variant appears to positively impact classification performance. For example, the results for "C+T" (client-side + trail dwell times) are better than when client-side or trail dwell time are used in isolation (see Table 2). In accordance with this, the highest performance is achieved when all three dwell times are combined. In addition, client-side dwell time becomes more useful when it is combined with trail dwell time, perhaps because this better captures the usefulness of the landing page.

In our final experiment, we combined the dwell time features with query features used previously to predict click satisfaction [12] (See the "Query performance prediction" category in Table 1). All experiments in this paper thus far have used only dwell time features. In this analysis we combine dwell time estimates with the query features. Table 5 shows those results and those from a baseline that uses only the query features to predict satisfaction. As shown in the table, dwell time features can lead significant improvements over the query performance predictors. We also note that dwell-time only features can outperform query-only features (e.g., the query features accuracy is 0.6344, see Table 5, compared to an accuracy of 0.6722 ("S+C+T" system in Table 4). This clearly illustrates the important role of dwell time in modeling search satisfaction.

6. CONCLUSION

In this paper, we compared three different ways to estimate dwell times (server-side, client-side, and trail) for the task of predicting click-level search satisfaction. Server-side dwell time is calculated by using only search actions in search engine results pages (SERPs). However, the other dwell times are measured by client-side information (i.e. the actual time pages are open in the browser). While client-side dwell time measures the time spent on only initially clicked result pages (i.e., the clicked pages displayed in a SERP), trail dwell time also considers the following clicks originating from the initial click (i.e., the relevant pages not shown in a SERP but visited on the navigation trail following a SERP click). To collect click data, we sampled from toolbar and search logs from a commercial search engine, and employed human assessors to review search behavior and identify SAT and DSAT clicks. Using this data, we estimated the dwell time on clicked results using the three dwell time techniques, and developed a classification model to recognize click satisfaction by featurizing dwell times and search queries.

In experiments, we found that trail dwell time performs the best in predicting click satisfaction. We also found that server-side dwell time is better than client-side dwell time. Server-side dwell time also has the advantage that it is computable for all users of the search engine and does not require any client-side instrumentation. This suggests that using trail dwell time is the best option when client-side information is available, but when it is not available server-side dwell time is a useful alternative.

Moreover, we showed that the dwell times can lead to significant improvements in performance when combined with other query features. This is important because it suggests that satisfaction classification models such as [7] can be further improved by using more features in addition to dwell times, and considering the source of the dwell time estimates that they employ. For future work, we plan to investigate how to handle situations where dwell time information is not available directly in the logs (e.g., for the last click in session where there is no follow-on event on which to base dwell time estimates), explore and develop additional click satisfaction features devised by various dwell times. There are other important areas such as the impact of user revisitation on page dwell times, as well as the nature of the pages themselves (e.g., whether or not they are waypoints [16]). Future work will study the impact of these issues on the performance of SAT and DSAT click prediction models.

REFERENCES

[1] Bilenko, M. and White, R. W. (2008). Mining the search trails of surfing crowds: Identifying relevant websites from user activity. *Proc. WWW*, 51–60.

[2] Claypool, M., Le, Phong, Waseda, M., and Brown, D. (2001). Implicit interest indicators. *Proc. IUI,* 33–40.

[3] Cronen-Townsend, S., Zhou, Y., and Croft, W.B. (2002). Predicting query performance. *Proc. SIGIR*, 299–306.

[4] Downey, D., Dumais, S.T., and Horvitz, E. (2007). Models of searching and browsing: Languages, studies, and applications. *Proc. IJCAI*, 2740–2747.

[5] Fox, S., Karnawat, K., Mydland, M., Dumais, S., and White, T. (2005). Evaluating implicit measures to improve web search. *ACM TOIS*, 23(2): 147–168.

[6] Friedman, J.H. (2001). Greedy function approximation: a gradient boosting machine. *The Annals of Statistics*, 29(5): 1189–1232.

[7] Hassan, A., Jones, R., and Klinkner, K.L. (2010). Beyond DCG: User behavior as a predictor of a successful search. *Proc. WSDM*, 221–230.

[8] He, B. and Ounis, I. (2006). Query performance prediction. *Information Systems*, 31(7): 585–594.

[9] Huffman, S. and Hochster, M. (2007). How well does result relevance predict session satisfaction? *Proc. SIGIR*, 567–574.

[10] Kelly, D. and Belkin, N.J. (2001). Reading time, scrolling, and interaction: Exploring implicit sources of user preferences for relevance feedback. *Proc. SIGIR*, 408–409.

[11] Kelly, D. and Belkin, N.J. (2004). Display time as implicit feedback: Understanding task effects. *Proc. SIGIR*, 377–384.

[12] Kim, Y., Hassan, A., White, R.W., and Zitouni, I. (2014). Modeling dwell time to predict click-level satisfaction. *Proc. WSDM*, 193–202.

[13] Liu C., White, R.W., and Dumais, S. (2010). Understanding web browsing behaviors through Weibull analysis of dwell time. *Proc. SIGIR*. 379–386.

[14] Wang, K., Thrasher, C., Viegas, E., Li, X. and Hsu, P. (2010). An overview of Microsoft web n-gram corpus and applications. *NAACL HLT Demo Session*, 45–48.

[15] White, R.W. and Kelly, D. (2006). A study on the effects of personalization and task information on implicit feedback performance. *Proc. CIKM*, 297–306.

[16] White, R.W. and Singla, A. (2011). Finding our way on the Web: Exploring the role of waypoints in search interaction, *Proc. WWW*, 147–148.

[17] Xu, S., Jiang, H., and Lau, F.C.M. (2011). Mining user dwell time for personalized web search re-ranking. *Proc. IJCAI*, 2367–2372.