# Understanding User Adjustment to Slow Search

Ryan Burton
University of Michigan School of Information
105 S. State St.,
Ann Arbor MI, 48109
ryb@umich.edu

Kevyn Collins-Thompson
University of Michigan School of Information
105 S. State St.,
Ann Arbor MI, 48109
kevynct@umich.edu

## ABSTRACT

In this paper, we present a behavioral analysis of a system that embodies characteristics of *slow search*, where retrieval speed is traded off for a simulated improvement in quality. We compare system usage by the topic of the users' search task and show that particularly when users use slow search and are exposed to relevance gains over time, their behaviour adjusts differently as they progress towards their goal. We also compare the variances in the performance of the system by subject condition, and argue that the comparatively low variance in the outcomes of slow search makes it a robust alternative to traditional Web search.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Storage and Retrieval– *Search process*; H.5.m [**Information interfaces and presentation (e.g., HCI)**]: Miscellaneous

## Keywords

search behavior; interactive information retrieval; user interfaces; slow search

## 1. INTRODUCTION

Much of the engineering effort behind contemporary information retrieval systems is focused on returning search results as quickly as possible. Typically, this involves making simplifying assumptions about the linguistic and semantic complexity of queries and documents for matching and ranking [5, 3]. This limits the degree of sophistication that may otherwise be possible when implementing measures to increase effectiveness, such as personalization or result diversification.

Slow search has been proposed as a possible avenue to allow systems to trade speed for effectiveness [4]. First introduced as a concept in a paper by Dörk et al. [2], Teevan et al. [5] performed a series of surveys to investigate users' attitudes towards waiting for search results [5]. We take this idea yet further, implementing an asynchronous system in the form of a Chrome extension that allows users to wait for better results for a search query by simulating a gain in relevance.

In this study, we investigate how such a system is used to complete search tasks, comparing a system with asynchronous slow search capabilities and ranking improvement over time with usage where there is no time-based gain and a baseline Web search system. Our focus in this paper will be on behavioral differences based on information need. It is important for us to know for what purpose they will use such a system, and what their behaviors are when they elect to do so. The results of this analysis will have implications on the design of future search engines, which may include more flexible time constraints for gains in quality.



## 2. STUDY DESIGN

We performed a laboratory study in which we presented a list of four topics (Education, Entertainment, Local Businesses, and Shopping), each with four search tasks, and asked users to complete one task from their choice of two separate topics. Each task was a multi-attribute question, which required answers that satisfied all constraints of the problem. As an example, one task in the Local Businesses topic was as follows: *Name five I.T. companies in Ann Arbor with at least 50 employees.*

We gave partial credit based on the number of correct items given in the response, but an item was only considered correct for satisfying all the constraints. We also asked users to provide three relevant Web pages to answering the question. Answering the question perfectly resulted in a reward of $2, and giving three relevant Web pages resulted in a $3 bonus ($1 for each relevant page).

For our search interface, we created a Chrome browser extension that augments the functionality of Web search engines by providing a "Work Harder" button on the search engine result page. Clicking this button adds the current query into a sidebar present on the result page. For queries being processed, we display a progress bar for the query as well as the top three results and snippets at any given time. The user may click on a "more results" link to view the full list of results as they improve with time. This page also presents its own progress bar, and updates asynchronously such that the page may be left open while a user continues to search on the main page. The current study restricted the number of concurrent slow queries per user at one at a time, but this query may be cancelled at any time and removed from the sidebar. We provide a screen shot of the system in Figure 1.

On the backend, we implemented a server that communicates with the Chrome extension to simulate an improvement in relevance over the course of five minutes for each slow query submitted. For each task that a user may choose to tackle, we manually selected five to ten high-quality documents and snippets that help to solve

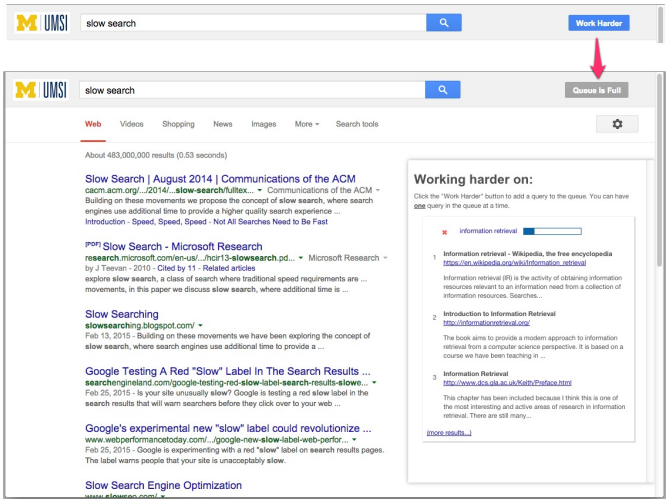


**Figure 1: Interface with "Work Harder" button and sidebar. Clicking the "Work Harder" button in the upper right adds the current query to the queue. The user may continue searching normally on a separate query.**

the problem posed by the task. When the "Work Harder" button is used, the server selects documents from the pool to insert into the ranking every twenty seconds. Similarly, another process on the server increases the ranking of high-quality documents currently in the ranking over the course of the five minute period until these documents reach the top of the ranking.

## 2.1 Procedure

We assigned participants to one of three conditions. In the baseline condition, the interface resembles a conventional Web search engine, with no "Work Harder" button or sidebar. In the "*static gain*" condition, the interface adds the "Work Harder" button and sidebar to the conventional interface. Furthermore, the system inserts highly-relevant documents in the middle of the ranking and the rank position of these documents stays the same over the course of the five minutes. Finally, in the "*dynamic gain*" condition, the interface is the same as in the "*static gain*" condition, but the system inserts documents at the bottom of the ranking and then continuously increases the position of documents at 20 second intervals, over the five minute time window, until they finish at the top of the ranking.

## 2.2 Participants

We recruited 44 participants (26 women and 18 men), most of whom reported being very experienced with Web search. The majority ($n = 38$) also reported using Web search more than once per day. The mean age of the participants was 23.5 ($\sigma = 5.9$), and most were undergraduates ($n = 18$) or had already received an undergraduate degree ($n = 12$).

## 2.3 Results

### 2.3.1 How users progressed towards a goal over time

In Figure 2, we plot the average relevance of the documents clicked by users, faceted by topic. The two most popular topics, *Entertainment* and *Local Businesses*, were the only two to have representation from all conditions in both sessions (with 27 *Entertainment* sessions and 31 *Local Businesses* sessions in total). We will first compare user behavior by topic, and then turn our attention to contrasting the characteristics of the sessions for each session as they correspond to the conditions within these topics.

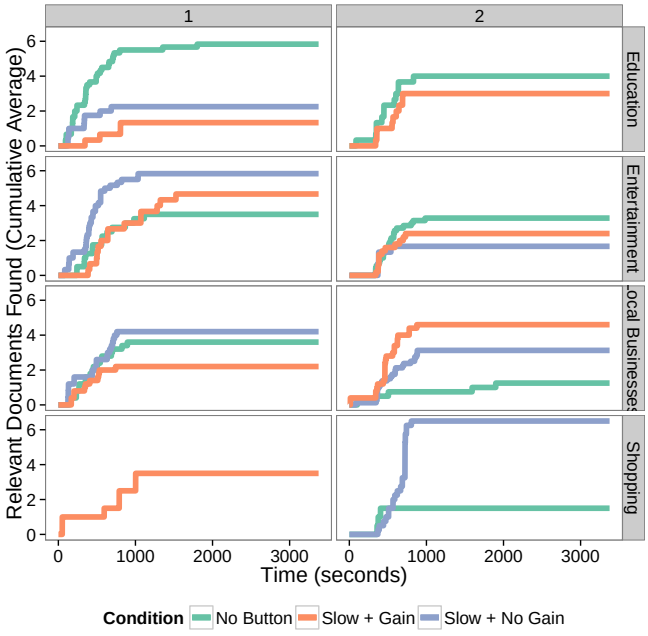


**Figure 2: Average time-relevance curves by topic.**

**Comparing behaviors by topic.** In Table 1, we outline a list of features that we computed to characterize search behavior. We present these features by topic, as a central question for the development of such a system is whether it will be used differently depending on the type of information need. We aggregated the needs by topic and compared the top two most addressed topics, *Entertainment* (28 completed tasks) and *Local Businesses* (32 completed tasks), against the combination of *Education* (21 completed tasks) and *Shopping* (7 completed tasks). We also performed pairwise Mann-Whitney U tests adjusted with Bonferroni correction to compare differences across topics.

As Table 1 shows, *Other* tasks had the longest mean session length at 1001 seconds. In comparison, *Local Businesses* had the shortest sessions on average at 793.4 seconds, while *Entertainment* had a mean session length of 841 seconds. *Local Businesses* also had the most regular queries in its sessions, with an average of 7.656. With this combination, *Local Businesses* also had the highest rate of regular queries in a session at 0.007928 queries per second, which is significantly higher than that of *Other* at 0.00497 ($p = 0.029$). *Local Businesses* additionally had an average dwell time of 247.2 seconds, which is significantly shorter than that of the *Other* category (392.8 seconds; $p = 0.023$).

Comparing rewards (and precision, which is related), *Entertainment* ends up having the worst performance outcome by users' answers at a $3.943 average reward and an average precision of 0.7286. This is significantly lower than the respective outcomes of the *Other* category, which has an average reward of $4.348 and an average precision of 0.8593.

**Comparing task sessions.** We computed the same behavioral features, separated by topic, to compare sessions being completed first versus sessions being completed second for the same topic. We additionally performed Mann-Whitney U tests to determine whether the differences observed between sessions were statistically

| | Entertainment (E) | Local Businesses (L) | Other (O) | U Test |
|---|---|---|---|---|
| **Baseline features** | | | | |
| Session length (sec.) | 841 | 793.4 | 1001 | - |
| Regular queries | 6.821 | 7.656 | 5.593 | - |
| Regular queries/sec | 0.006827 | 0.007928 | 0.00497 | L > O |
| **Slow query features** | | | | |
| Slow queries | 0.75 | 0.3438 | 0.4074 | - |
| Slow queries/sec | 0.000913 | 0.000416 | 0.000526 | - |
| Slow queries cancelled | 0.4286 | 0.0625 | 0.1852 | - |
| Slow queries cancelled/sec | 0.000439 | 0.000079 | 0.000232 | - |
| **Query features** | | | | |
| Query word length | 4.982 | 4.907 | 5.565 | - |
| Query character length | 28.7 | 29.8 | 35.17 | - |
| **Click features** | | | | |
| Pages in session | 13.14 | 14.41 | 11.85 | - |
| Clicks per query | 2.684 | 2.707 | 3.335 | - |
| Time to first click (sec.) | 15.1 | 23.85 | 36.06 | - |
| Dwell time (sec.) | 245.9 | 247.2 | 392.8 | L < O |
| **Outcomes** | | | | |
| Click relevance | 3.679 | 3.219 | 3.889 | - |
| Reward (dollars) | 3.943 | 4.075 | 4.348 | E < O |
| Reward Variance (dollars) | 1.025 | 1.579 | 1.79 | - |
| Precision | 0.7286 | 0.8187 | 0.8593 | E < O |

**Table 1: Comparison of behavioral features by topic.**

significant. The full table was omitted for space. Most of the significant differences were seen in the *dynamic gain* condition.

For users in the *dynamic gain* condition, those who performed *Entertainment* queries cancelled fewer slow queries ($p = 0.05$) in the second session (2 slow queries cancelled to 0.2 cancelled on average). It should be noted that these users also performed fewer slow queries in the second session, but this difference was not statistically significant (a drop from 2 slow queries on average to 1; $p = 0.49$). This was the only condition and topic of the two for which there was a meaningful difference in cancellation behavior based on task order. For the same condition, users performing *Local Businesses* tasks did not cancel any queries in either session.

We also observed that *dynamic gain* users performing *Entertainment* tasks saw a decrease in the average relevance score of documents clicked in the session (from 4.67 to 2.4, $p = 0.03$). While the difference for *Local Businesses* was not significant, we note for comparison that the average relevance for *Local Businesses* increased from 2.2 to 4.6 between tasks ($p = 0.16$). Similarly, the relevance score for Education, which has data for this condition for both task sessions, increased from 1.33 to 3 ($p = 0.35$). Users in this condition conducting *Entertainment* in fact viewed fewer (27.3 on average to 8; $p = 0.23$).

*Dynamic gain* users as well performing *Local Businesses* tasks switched from performing no slow queries in the first session to performing an average of one slow query in the second session. This increase was statistically significant ($p = 0.023$).

| | *Static Gain* | *Dynamic Gain* | Baseline |
|---|---|---|---|
| Reward (dollars) | 4.21 | 4.04 | 4.16 |
| Reward Variance (dollars) | 1.13 | 1.12 | 2.14 |

**Table 2: Reward means and variances by study condition**

There were no statistically significant differences within the *static gain* condition for these topics, but we note that the number of slow queries dropped from 1.17 to 1 ($p = 1$) for *Entertainment* and increased from 0.2 to 0.625 ($p = 0.31$) for *Local Businesses*. This behavior is actually consistent within the topics: as previously noted, the slow queries performed in for *Entertainment* also dropped for *dynamic gain* from 2 to 1, and slow queries increased from zero to 1 for *Local Businesses* in the *dynamic gain* condition.

The only statistically significant difference between tasks for the baseline condition was seen in *Local Businesses* users, where their result clicks per query dropped from 2.07 to 0.996 ($p = 0.027$).

The differences within the *dynamic gain* condition and the lack of many significant differences in other conditions seems to point to a stronger effect of adjustment to the system for users exposed to both the asynchronous capabilities and time-based gain [1].

We also found interesting differences in the way users progressed in these tasks, though the differences were not statistically significant. *Entertainment* sessions typically became shorter in duration in the second session across all conditions (994 seconds to 702.4 seconds on average), and *Local Businesses* sessions became longer (658.8 seconds to 893.3 seconds on average). In aggregate, the number of queries performed changed accordingly (8.3 to 5.4 for queries in session for *Entertainment* and 5.2 to 9.7 for *Local Businesses*). However, users examined more documents per query in *Entertainment* (2.6 to 2.8) and fewer in *Local Businesses* (3 to 2.5). Furthermore, the number of slow queries issued per session is concomitant with the number of regular queries by task and condition, however, we found that the number of highly relevant documents clicked increased during the second task compared to the first for both topics in the *dynamic gain* condition, which is the opposite of what was observed in other conditions. Thus, while users adjusted their behavior differently depending on the topic as they progressed through the experiment, users in the *dynamic gain* condition were able to consistently find the highly relevant documents regardless of other changes in interaction.

### 2.3.2 *Performance Robustness*

To measure performance, we used the reward earned per task by participants in each study condition. We present a histogram of these rewards in Figure 3 and summary statistics in Table 2. We performed Mann-Whitney U tests on these statistics by condition. However, none of these differences were statistically significant.

The *static gain* and *dynamic gain* conditions had the highest and lowest rewards respectively ($4.21 for the *static gain* and $4.04 for the *dynamic gain* condition). One possible explanation for this difference is that in the *static gain* condition, the system inserts all of the same documents that would have been inserted in the *dynamic gain* condition, but in the lower half of the ranking. As a result, *static gain* users gained access to these documents immediately, and seemed to be willing to look for them in the ranking. *Dynamic gain* users however had to wait to see these same results: in the *dynamic gain* condition, the system inserts documents into the ranking and improves the ranking of relevant documents steadily over the course of five minutes.

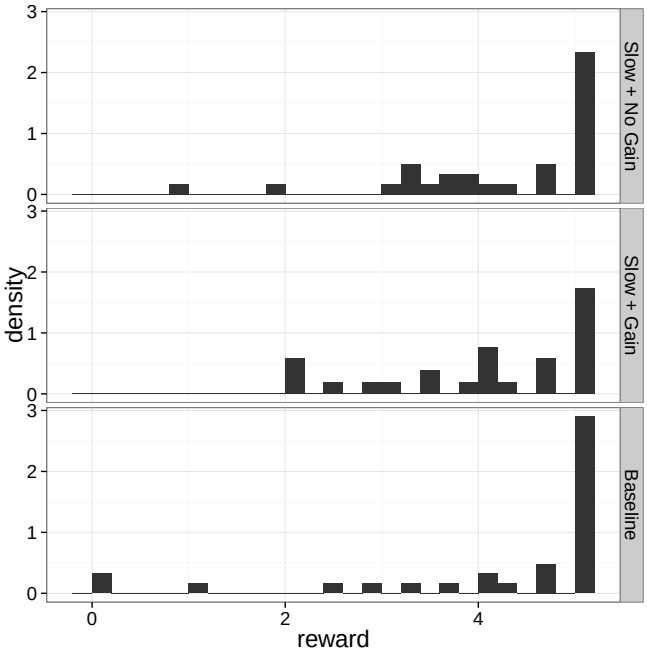


**Figure 3: Distribution of rewards by study condition.**

With that said, we also noticed that these two conditions had the smallest variances ($1.13 and $1.12 for *static gain* and *dynamic gain* respectively), compared to the baseline ($2.14). This gives the slow search conditions a greater degree of stability and predictability; for the baseline condition, especially in comparison to the *dynamic gain* condition, there is a greater risk, but also a potentially greater reward to using it for these multi-attribute tasks. Therefore, we believe that there is usefulness in both types of search; in fact, having traditional search as an option to additionally having slow search may indeed be a useful approach.

# 3. DISCUSSION AND FUTURE WORK

The results of our study show that slow search is a robust alternative to Web search for multi-attribute tasks, minimizing the worst case performance that one experiences in using the system. We also compared how users behaved in satisfying a particular information need when a topic is approached first versus when the same topic is approached second, and show that there are many behavioral characteristics that change depending on when the topic is attacked when users are exposed to slow search with a gain in quality. We believe that this shows that there is a period of acclimatization in this case, where users take some time to adjust to the capabilities of the system.

One limitation of this study is the length of time for which users are exposed to the system. Our study sessions were designed to give users a ten minute exploratory "training" period to probe the capabilities of the system, after which they would have thirty minutes to solve each of two problems. This may have in fact not been enough time for training, and in a future study, we will adjust the training period to reduce novelty effects.

# 4. REFERENCES

[1] C. L. Clarke and M. D. Smucker. Time well spent. In *Proceedings of the 5th Information Interaction in Context Symposium*, pages 205–214. ACM, 2014.

[2] M. Dörk, P. Bennett, and R. Davies. Taking our sweet time to search. In *Proceedings of CHI 2013 Workshop on Changing Perspectives of Time in HCI*, 2013.

[3] M. L. Mauldin. Retrieval performance in ferret a conceptual information retrieval system. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 347–355. ACM, 1991.

[4] J. Teevan, K. Collins-Thompson, R. W. White, and S. Dumais. Slow search. *Communications of the ACM*, 57(8):36–38, 2014.

[5] J. Teevan, K. Collins-Thompson, R. W. White, S. T. Dumais, and Y. Kim. Slow search: Information retrieval without time constraints. In *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*. ACM, 2013.