What Affects Word Changes in Query Reformulation During a Task-based Search Session?

Jiepu Jiang
Center for Intelligent Information Retrieval
College of Information and Computer Sciences
University of Massachusetts Amherst
jpjiang@cs.umass.edu

Chaoqun Ni
School of Library and Information Science
Simmons College
chaoqun.ni@simmons.edu

ABSTRACT

This paper performs an analysis on the influence of different factors on users' choices of specific word changes in query reformulation during a search session. We study three types of word changes: whether to remove or retain a word in the current query; whether or not to add a brand-new word to the query; whether or not to reuse a word (included in previous queries, but removed in the current query). Three types of factors are examined: session-level factors measuring task and user characteristics; query-level factors related to past user activities in a session; word-level factors for the characteristics of the examined word and its relation to the current query and search results. Statistical analysis suggests that: word-level factors strongly influence all three types of word changes; query-level factors only show a clear influence on retaining or removing a word; task-level factors exhibit limited direct influence on all three types of word changes. Analysis also disclose reasons for different word changes: users remove a word to stop exploring a subtask, or to correct bad performing queries; they look for related, unused words from recently viewed result summaries and add to queries; reusing a word usually indicates reverting from a subtask to the main task or another subtask.

Keywords

Query reformulation; search session; context; task.

1. INTRODUCTION

Some simple information needs such as finding home pages may be satisfied by a single query and one click, but it usually requires multiple searches to solve more complex tasks. The reasons vary. For example, sometimes it is the user who employs a divide and conquer strategy, using each query to deal with a part of the task [3]. Sometimes it is the user's limited knowledge about the problem that makes search and query formulation difficult [4]. For whichever reason, a complex search problem usually involves more than one query (a search session).

The activity to modify an existing query and perform a new search is called *query reformulation*. Previous work [1, 5, 11, 12, 26] studied patterns of reformulation at query level, such as adding, deleting, and replacing words. These patterns provide insights on how query reformulations look like, but they also have a few limitations:

- They do not look into specific words, e.g., why users add or remove a specific word instead of others.
- They are patterns for the *outcome* of users' query reformulations, but are not necessarily indicative of users' decisions in query reformulation or the reasons for reformulation.

In contrast, we focus on individual and specific word changes in query reformulation. All vocabulary differences in query reformulation can be decomposed into three types of word changes: to retain or remove a word in the current query; to add a brand-new word to the next query or not; whether or not to reuse a word that was involved in previous queries, but removed in the current query. These word changes stand for finer-grained decisions compared with the query-level reformulation patterns.

We examine the influence of three categories of factors on word changes: session-level factors measuring task and user characteristics; query-level factors related to past user activities in a session; word-level factors for the characteristics of the examined word and its relation to the current query and search results.

2. RELATED WORK

2.1 Query Reformulation

Query reformulation, in the scope of this paper, refers to the activity of formulating a query that is different from an existing one. We focus on the difference of the two queries.

Previous work characterized patterns of reformulation at query level [1, 5, 11, 12, 26]. Some are characterized from the lexical aspect, for example: adding words, removing words, replacing words by synonyms, spelling correction, stemming, case change, and using acronyms. Some are concerned with syntactic differences, for example: punctuation, reordering words, and using search operators. Some patterns may imply users' intents, for example: specification, generalization, and subtopic change. These patterns are not exclusive of each other. For example, one can reformulate a more specific query (specification) either by adding words, or replacing words with more specific ones.

CHIIR '16, March 13-17, 2016, Carrboro, NC, USA
© 2016 ACM. ISBN 978-1-4503-3751-9/16/03…$15.00
DOI: http://dx.doi.org/10.1145/2854946.2854978

In contrast to previous work, we look into changes in query reformulation at word level—the unit of analysis in our study is a specific word, and whether users will remove, retain, or add the word in query reformulation. This is also relevant to previous work on choices of words in query reformulation and interactive relevance feedback.

Spink et al. [30] studied five sources of query terms in mediated online searching. Among the sources they examined, question statement is similar to task description in our study, and we also consider relevance feedback as a source for new terms. In addition, the content of search results is also an important source of knowledge for query reformulation [18]. Yue et al. [33] examined possible sources of query words in collaborative search. Some of them may also be applied to other types of searches, including users' past queries and viewed search results. Another source we examined is query suggestions displayed on the SERP. Kelly et al. [17] compared term and query suggestions, where users reported that query suggestions provide ideas for manually formulating queries; Jiang et al. [13] reported that before query reformulation, searchers viewed task description and query suggestions more frequently.

The word changes we examined in this paper were rarely studied in previous work from the user's perspective. However, some work built technical solutions for contextual search and query suggestion based on these word changes. Guan et al. [10] separately considered added, retained, and removed words in relevance feedback; Dang et al. [8] generated synthesized query suggestions by considering similar patterns.

2.2 Search Session and Search Task

We study query reformulation in the context of a *search session*—a period that the user searches consecutively for the same search task. This definition is ideal. Practically a user may perform multiple tasks in an interleaved way [29]. In such case, one needs to identify and concatenate queries for each unique task [14] to obtain such sessions.

Search task [31] is a widely studied factor influencing user interaction in a search session. Li et al. [19] classified task from six facets: source, task doer, time, products, process, and goal. We examine the influence of task goal and product on word changes. Other ways of characterizing search tasks exist [16, 32], but we do not consider them in this study.

Much previous work studied search behavior variation in sessions with different task goals [6, 7, 13, 20, 24], products [6, 7, 13, 20, 24], complexity [6, 7, 24], and at different stages [22, 23], and etc. However, few studied the influence of tasks on query reformulation, especially on word changes in query reformulation. Liu et al. [21] compared the frequencies of using different reformulation patterns in different tasks.

3. APPROACH

3.1 Dependent Variables

We use S for a session, q_i for the ith query of a session, and $q_k \rightarrow q_{k+1}$ for the reformulation from q_k to q_{k+1}. We do not consider query reformulations where any of the two queries are query suggestions. When discussing $q_k \rightarrow q_{k+1}$, we call q_k *the current query* and q_{k+1} *the next query* or *the next query.* We consider a query as *a set of words* and ignore word sequence. We also do not consider multiple occurrences of the same word in a query, because this only happens in 2 out of 388 queries in the collected data.

We decompose a user's decisions in $q_k \rightarrow q_{k+1}$ into the following two types of events:

- For each word in q_k, to decide whether to retain or remove the word in the next query (q_{k+1}).
- For each word in C, to decide whether or not to add the word to the next query (q_{k+1}).

Here C stands for a candidate set of words in the user's mind being considered for adding to the next query or not. In this paper we consider two types of words added to the next query. The first type is words that are brand-new in the session, i.e., none of the previous queries included the word. We refer to this case as *adding a new word* and use C_{new} for its candidate set. The second type is words that were involved in past queries (from q_1 to q_{i-1}), but removed in q_i. We refer to this case as *reusing a word* and use C_{reuse} for its candidate set. We separately consider these two types of words because they may stand for different intentions of users. Any added words should belong to either type. We discuss choices of the candidate sets in Section 6.

Any vocabulary changes in a query reformulation can be categorized into the two types of events. Words in the current query are either retained or removed in the next query. Words in C_{new} and C_{used} are either added or not.

Formally, we study three binary dependent variables:

- For $w \in q_k$, $\mathrm{Y}_{\mathrm{rmv}}(w, q_k, q_{k+1}) = 1$ if the user removes w in $q_k \rightarrow q_{k+1}$, or 0 otherwise.
- For $w \in C_{new}$, $\mathrm{Y}_{\mathrm{new}}(w, q_k, q_{k+1}) = 1$ if the user adds w to q_{k+1} in $q_k \rightarrow q_{k+1}$, or 0 otherwise.
- For $w \in C_{used}$, $\mathrm{Y}_{\mathrm{used}}(w, q_k, q_{k+1}) = 1$ if the user adds w to q_{k+1} in $q_k \rightarrow q_{k+1}$, or 0 otherwise.

3.2 Independent Variables

The purpose of this study is to analyze the influence of different factors on word changes, as characterized by the three dependent variables. Table 1 lists all independent variables considered in this paper. We divide them into three groups depending on their relations to the dependent variables.

3.2.1 Session-level Variables

Session-level variables measure factors related to the characteristics of the task and the searcher. Within a search session, every word change in each query reformulation shares the same influence from session-level variables.

We follow Li and Belkin's faceted task classification framework [19] and consider three characteristics of search tasks: the goal of a search task is either clear or amorphous (`goal`); the product of a search task is either factual information, or enhanced intellectual understanding of the user (`product`); user's self-rated familiarity on the task using a five-point Likert scale (`familiarity`).

In addition, we suspect that users' choices of word changes in a query reformulation may also depend on individual preference. Some searchers may prefer to add or remove words in general. In order to capture this factor, we compute the average number of added (`avg_num_add`) and removed words (`avg_num_rmv`) during a query reformulation in other sessions performed by the same searcher as surrogates for the users' preferences for adding or removing words.

Table 1: Independent variables for analyzing word changes in query reformulation.

Group	Variable	Explanation
session	goal	0 if the goal of the search task is clear, or 1 if it is amorphous [19].
	product	0 if the task looks for factual information, or 1 if for intellectual understanding [19].
	familiarity	User's self-rated familiarity regarding the topic of the task using a five-point likert scale.
	avg_num_rmv/add	Average number of removed/added words in other sessions by the same user.
query	q_length	Length of the current query (excluding stop words).
	q_duration	Time duration from the submission of the current query to that of the new query.
	q_clickpos	Position of the lowest ranked clicked results on the SERP, or 0 if no click.
	num_query	Number of submitted queries in the session (including the current query).
	num_click	Number of past clicks in the session (including clicks on the current query's SERP).
	duration	Time duration from the beginning of the session to the submission of the new query.
word	idf	IDF of the word in the ClueWeb09 collection.
	p(w|pastq)	Probability of the word appearing in past queries of the session.
	avg_jaccard	Average Jaccard similarity of the word with other words in the current query.
	#click_hasw	Number of clicked results whose title/snippet/URL contains the word.
	#skip_hasw	Number of skipped results whose title/snippet/URL contains the word.
	freq_w_suggest	Frequency of the word in query suggestions if users viewed the area, or 0 otherwise.

3.2.2 Query-level Variables

Query-level variables measure factors related to past user activities in a session. These factors apply to a query reformulation as a whole. Each word change in a query reformulation shares the same influence from query-level variables.

We suspect users' choices of word change in a query reformulation is directly influenced by the most recent search. We include variables for the characteristics of the current query, such as q_length, and user activities on its SERP, including q_duration and q_clickpos. In addition, we include variables for the time of a session when the reformulation happened, including num_query, num_click, and duration.

3.2.3 Word-level Variables

Word-level variables measure factors directly related to the word being examined. Different words in the same query reformulation can be affected differently by these variables.

idf indicates word specificity [28], i.e., whether the word is general or specific. p(w|pastq) is the probability that the word w was included in past queries of the session (from q_1 to q_i). We use p(w|pastq) to measure the centrality of a word to the task. Example 1 shows queries in a session, where the task is to find information on the symptoms and treatments of depression. "Depression" is included in every query and expresses the main theme of the task, which is unlikely to be removed in query reformulation.

Example 1	
1	**depression** symptoms
2	**depression** definition
3	**depression** treatment
4	**depression** treatment cost

avg_jaccard measures the connection between the word being examined and other words in the current query using their co-occurrences in documents. Here we define the Jaccard similarity of two words as that between the sets of documents containing each word. When examining a word w, we calculate its Jaccard similarity with each word that is not w in the current query, and use the mean value as an independent variable (avg_jaccard). If there is no other words in the query, we set the value to the mean value of avg_jaccard in other query reformulations.

In addition, we examine variables measuring occurrences of the word in results displayed on the current query's SERP. We separately consider clicked results (#click_hasw) and skipped results (#skip_hasw). Here we use *skipped results* to refer to those that users viewed their summaries but did not click on. We rely on eye movement to determine skipped results. We separately examine different elements of result summaries, including their titles, snippets, and URLs.

Moreover, we measure occurrences of words in query suggestions displayed on the current query's SERP and viewed by the user (freq_w_suggest). We consider the screen area for query suggestions as a whole. If the SERP provides query suggestions and we observed the user's eye fixations on that area, the variable's value is set to the frequency of the word in all query suggestions. Otherwise, we set its value to 0. We do not consider users' visual attention for individual query suggestions due to the limited accuracy of our device in tracking eye fixations on small items.

3.3 Analysis Approach

We examine the influence of independent variables on the dependent variables using hierarchical (multilevel) logistic regression, a technique that models variables with more than one variance component [9]. More specifically, it deals with a regression model with binary outcome (dependent variable), and varying coefficients of independent variables. This study performs analysis using SPSS 23.

We use hierarchical logistic regression instead of vanilla logistic regression because the observations in this study are nested—we examine multiple word changes nested within a query reformulation, and there can be multiple query reformulations nested within a session as well. In such case, the observations are not independent of each other, because some of them share the same contexts at the query and/or session levels. This violates the independence assumption to apply vanilla logistic regression. In contrast, hierarchical models can handle such issues. We use a hierarchical model with three levels to study word changes (level 1) during a query reformulation (level 2) in a search session (level 3).

However, it should be noted that our approach does not consider another type of dependency issue in word changes—decisions on one word may depend on those on other words. This is a limitation of our approach.

4. DATA

4.1 User Study

We use data from a previous user study [13] to analyze the proposed questions. The purpose of the user study was to compare search activity patterns in sessions of different types of tasks. The experiment controlled two characteristics of search tasks in Li et al.'s framework [19]: goal (specific or amorphous) and product (factual or intellectual). We did not consider tasks with a mixed goal and other types of product (e.g., image, mixed product) in Li et al.'s framework. Different combinations of task goal (two levels) and product (two levels) define four types of tasks. This is identical to the settings of the TREC session tracks [15] and is similar to many related studies [13, 20, 22, 24].

The experiment used a 2×2 within-subject design. Each subject performed four tasks of different types using an experimental search system. We employed 20 formal subjects and divided them into 5 groups. We assigned different tasks to each group to increase task diversity. Subjects in the same group performed the same 4 tasks, and we rotated task sequence using a Latin square. These tasks were developed by and used in the TREC 2012 session track [15]. In total, we collected 80 sessions from 20 unique subjects on 20 tasks.

The experimental search system provides modified Google search results. It redirects user queries to Google and returns the "10-blue links" and query suggestions (related searches). It removes other results such as sponsored links and direct answers. The system displays search results in the same way Google would display, e.g., the highlight of query terms are retained. The system records user search activities, including queries and clicks. In addition, we collected searchers' eye-movement on the screen using a Tobii 1750 eye-tracker. In this study, we determine that a user viewed a result summary or query suggestions if we observed an eye fixation on the corresponding area on the screen. We set the minimum duration of an eye fixation to 100 milliseconds, a common value adopted in many previous studies of web search behaviors using the same series of eye-tracker.

During experiments, the subjects were first introduced to the experimental search system and a training task. Then, they worked on four formal tasks. After finishing two formal tasks, they took a 10-minute break. For each task, they spent about 10 minutes to search information to solve the task. After each task, they answered post-search questionnaires (we only use their self-rated task familiarity in this study) and judged relevance of results. We required all the subjects to be English native speakers (to reduce the influence of language efficiency) and to have a perfect eyesight without glasses or lens (to ensure accuracy of eye tracking). More details were introduced in a previous article [13].

4.2 Dataset Statistics

The collected dataset includes 80 sessions and 388 requests in total. Among them, 39 are query suggestions, and 36 are turning to different pages of results for the same query. After removing these 75 requests, the rest of the dataset includes 313 queries formulated by the searchers and 203 query reformulations where both queries are formulated by the searchers. The average length of a query is 3.37 words excluding stop words (we use the stop words list in Indri). Among the 203 query reformulations, 158 removed at least one word from the current query, and 186 added at least one word to the next query. On average searchers removed 1.56 words and added 1.68 words during a query reformulation.

5. REMOVING OR RETAINING A WORD

This section reports results for whether to remove or retain a word in query reformulation ($\mathrm{Y_{rmv}}$). As discussed in Section 3.1, we examine each word in the current query (q_i) when studying $q_i \rightarrow q_{i+1}$. This is to assume that users consider each word in the current query and make decisions on whether to remove or retain the word in the next query. This yields 687 observations of $\mathrm{Y_{rmv}}$ from 203 query reformulations, where 304 (44%) are positive ($\mathrm{Y_{rmv}} = 1$).

Table 2 reports results for hierarchical logistic regression, where $\mathrm{Y_{rmv}}$ is the dependent variable. exp(B) > 1 suggests a positive influence of the independent variable on $\mathrm{Y_{rmv}}$, and exp(B) < 1 indicates a negative one. Model 1 includes only session-level variables and constant. Model 2 further includes query-level variables. Model 3 includes all variables.

We transform some variables by taking natural log values to make them linear, including: `q_length`, `q_duration`, and `avg_jaccard`. We examine multicollinearity using the variance inflation factor (VIF). A value greater than 5 suggests a cause for concern, and a value greater than 10 indicates a serious collinearity problem [25]. We exclude `avg_num_add` due to its high correlation with `avg_num_rmv` ($r = 0.85$). Similarly, we remove `#click_url_hasw` because of its correlation with `#click_title_hasw` ($r = 0.86$) and `#skip_url_hasw` ($r = 0.83$ with `#skip_title_hasw`). All variables included in Table 2 satisfy VIF < 5.

5.1 Influence of Session-level Variables

Session-level variables show certain influence on whether to remove or retain a word in query reformulation. Model 1 explains the collected data significantly better than a baseline model including only constant ($p < 0.001$ by the Omnibus test). However, the magnitude of change in -2 log likelihood is small, indicating only a mild influence.

Among these variables, only the user's average number of removed words in other sessions (`avg_num_rmv`) consistently shows a significant positive effect in all three models—users are more likely to remove a word in a session if they removed words more frequently in other sessions. This suggests that a user's overall preference to remove a word may affect their decisions on removing or retaining a word in a specific session. Some users may prefer to remove words in query reformulations in general, and they are likely to do so in a specific session. But it is unclear whether such preference is related to other factors, e.g., search expertise.

Task product (`product`), goal (`goal`), and topic familiarity (`familiarity`) have no significant effects in any models. This indicates that the examined task characteristics may not directly affect removing or retaining a word.

5.2 Influence of Query-level Variables

Query-level variables also show certain influence on $\mathrm{Y_{rmv}}$. After including query-level variables, Model 2 significantly improves over Model 1 ($p < 0.001$). Over half of the query-level variables show significant effects in both Model 2 and Model 3. This suggests that removing or retaining a word

Table 2: Results for hierarchical logistic regressions: $\mathbf{Y_{rmv}}$ as dependent variable.

Step	Variable Name	Model 1		Model 2		Model 3			
		exp(B)		exp(B)		exp(B)		95% CI	
1	`constant`	**0.236**	***	**0.039**	***	**0.068**	***	-	-
	`product (intellectual)`	0.974		0.682		0.825		0.545	1.250
	`goal (amorphous)`	1.001		0.926		0.929		0.638	1.353
	`familiarity`	1.018		1.064		1.085		0.917	1.283
	`avg_num_rmv`	**2.123**	***	**1.626**	**	**1.559**	*	1.063	2.285
2	`q_length (log)`			**2.067**	***	**1.675**	*	1.117	2.511
	`q_duration (log)`			**1.334**	**	**1.570**	***	1.236	1.993
	`q_clickpos`			1.042		**1.130**	**	1.038	1.230
	`num_query`			**1.213**	***	**1.165**	***	1.077	1.261
	`num_click`			0.953		0.959		0.897	1.026
	`duration`			**0.905**	*	**0.885**	*	0.804	0.974
3	`idf`					0.928		0.845	1.019
	`p(w|pastq)`					**0.216**	***	0.123	0.378
	`avg_jaccard (log)`					**0.851**	*	0.749	0.966
	`#click_title_hasw`					**0.734**	**	0.590	0.912
	`#click_snippet_hasw`					0.946		0.767	1.165
	`#skip_title_hasw`					0.921		0.797	1.065
	`#skip_snippet_hasw`					0.916		0.813	1.032
	`freq_w_suggest`					0.941		0.844	1.049
	−2 Log Likelihood (baseline 943.3)	917.3		877.1		802.5			
	Omnibus Tests of Model Coefficients	$p < 0.001$		$p < 0.001$		$p < 0.001$			

*, **, and *** indicate statistical significance at 0.05, 0.01, and 0.001 levels, respectively.

can be influenced by past user interaction in the session.

The length of the current query (`q_length`) shows a significant positive effect—users are more likely to remove a word in relatively longer queries. This is not surprising considering longer queries are more likely to include words that are unnecessary for the task.

Results also suggest that users are more likely to remove a word if they spent a relatively longer time on the current query (`q_duration` shows a significant positive effect) and clicked on results on the current query's SERP (`q_clickpos` shows a significant positive effect). This indicates a situation that the current query retrieved some relevant information (such that users would spend time on examining the results, instead of quickly reformulating to the next query without clicking). We further examined the collected data and found that this is probably because the majority of the removal happened when users had (successfully) finished exploring a facet of the task (a subtask) and switch to another subtask. Section 5.4 discusses more details.

In addition, results suggest that users are more likely to remove a word if they submitted many queries in a session (a significant positive effect of `num_query`), but they are less likely to do so with the time goes by in a session (a significant negative effect of `duration`). The two trends are seemingly conflicting with each other, since the time spent in a session (`duration`), unsurprisingly, has a moderate correlation with the number of issued queries (`num_query`) ($r = 0.52$). But the two variables may also stand for two different factors.

Submitting more queries does not necessarily mean a better search progress, because querying and SERP examination themselves provide limited relevant information. Submitting many queries may even indicate limited search performance, e.g., previous studies reported that users will compensate limited search performance by searching more frequently [2, 27]. The number of clicks (`num_click`), in contrast, may better indicate the amount of relevant information acquired in a search session, even though not all clicked results are relevant. In the collected data, `duration` better correlates with `num_click` ($r = 0.61$), while `num_query` only slightly correlates with `num_click` ($r = 0.33$).

As such, we believe results for `num_query`, `num_click`, and `duration` in Table 2 indicate two possible factors for whether to remove or retain a word in query reformulation. On the one hand, users are more likely to remove a word when the session has limited search effectiveness, which is supported by the significant positive effect of `num_query` and its connection with limited search performance reported in previous studies [2, 27]. On the other hand, searchers are less likely to remove a word after acquiring more relevant information, as suggested by the negative effect of `duration` and `num_click`. Note that in Table 2, `num_click` shows no significant effect, but this is in fact influenced by the relatively high correlation between `num_click` and `duration` ($r = 0.61$). After excluding `duration`, `num_click` shows a significant negative effect (exp(B) = 0.917, $p = 0.005$).

5.3 Influence of Word-level Variables

Word-level variables show a strong influence on whether to remove or retain a word in query reformulation. After including word-level variables, Model 3 significantly improves over Model 2 ($p < 0.001$). The −2 log likelihood reduces by 74.6 compared with Model 2. The magnitude is greater than that for the combination of session and query-level variables (66.2). This suggests word-level variables are more salient factors than session and query-level variables for removing or retaining a word in query reformulation.

The frequency of using a word in past queries of the same session (`p(w|pastq)`) has a significant negative effect on removing the word—users are less likely to remove a word that appeared frequently in previous queries. As we discussed in

Table 3: Mean values (S.E.) of variables for removed and retained words (post-hoc analysis).

Variables	Removed	Retained	
`#click_title_hasw`	0.63 (0.06)	1.03 (0.08)	***
`#click_snippet_hasw`	0.97 (0.08)	1.27 (0.09)	**
`#click_url_hasw`	0.46 (0.05)	0.78 (0.06)	***
`#skip_title_hasw`	1.17 (0.09)	1.75 (0.09)	***
`#skip_snippet_hasw`	2.02 (0.11)	2.49 (0.11)	**
`#skip_url_hasw`	0.82 (0.07)	1.30 (0.08)	***
`#unclick_title_hasw`	2.69 (0.16)	3.85 (0.15)	***
`#unclick_snippet_hasw`	4.25 (0.15)	5.15 (0.13)	***
`#unclick_url_hasw`	2.03 (0.13)	2.91 (0.13)	***
`freq_w_suggest`	0.35 (0.08)	0.64 (0.10)	*

*, **, and *** indicate significance at 0.05, 0.01, and 0.001 levels, respectively, by a two-tail Welch's t-test.

Section 3.2.3, this is probably because words repeatedly used in a session indicate the main theme of the task, which is less likely to be excluded in queries.

Results also suggest that users are more likely to remove a word that does not co-occur frequently with other words in the current query (`avg_jaccard` shows a significant negative effect on Y_{rmv}). This usually happens when the word is off-topic, overspecific, or misspelled. In these cases, the word co-occurs with other query words only in a limited number of (if any) documents, causing a low value of `avg_jaccard`.

In addition, results for `#click_hasw` and `#skip_hasw` suggest that searchers are more likely to remove a word in query reformulation if the word appeared less often in the current query's results. Although only `#click_title_hasw` shows a significant negative effect in Table 3, this is affected by the correlations among these variables ($r = 0.77$ between `#click_title_hasw` and `#click_snippet_hasw`, and r=0.68 between `#skip_title_hasw` and `#skip_snippet_hasw`). After we removed `#click_title_hasw` and `#skip_title_hasw`, both `#click_snippet_hasw` (exp(B) = 0.798, $p = 0.010$) and `#skip_snippet_hasw` (exp(B) = 0.876, $p = 0.003$) show significant negative effects. Similarly, `#skip_title_hasw` will show a significant negative effect (exp(B) = 0.861, $p = 0.006$) if we remove `#skip_snippet_hasw`.

Results from post-hoc analysis also confirms this finding. Table 3 compares occurrences of the removed and retained words in different result elements. In addition to the clicked and skipped results, we also examine *unclicked* results—any results displayed on the SERP that users did not click on, regardless of whether or not they viewed the results. Table 3 clearly suggests that removed words appear significantly less often in the current query's results compared with the retained words. This applies to all clicked, skipped, and unclicked results, and is consistent among different elements of the results. Therefore, we do not hope to over-emphasize the significant effect of word occurrences in clicked result titles (`#click_title_hasw`) in Table 2. It seems in general users are more likely to remove words that appeared less often in the retrieved results. This may happen when the word is ineffective (cannot retrieve any results containing the word when combining with other query words), or when the word is not central to the main theme of the task.

5.4 Qualitative Analysis

To better interpret results in Table 2, two authors of this article manually examined all 304 cases of word removal and divided them into two groups: removing with satisfaction (SAT remove), and removing due to dissatisfaction (DSAT remove). The Cohen's κ is 0.72. They further discussed the disagreements in annotation and came into a final decision. 199 (65%) removed words are classified as SAT remove, and 105 (35%) are DSAT remove. SAT remove is more frequent than DSAT remove in the collected data.

SAT remove stands for the case that a word has successfully served its purpose in a query and is removed afterwards. It usually happens when users finished exploring one facet of the task (a subtask) and plan to switch to another. Example 2 shows queries, the number of clicks, and the time spent on each query in a session. Removed words are highlighted. Removed words in the third, fourth, fifth, and sixth queries were labeled SAT removes. These words all indicate different subtopics related to sunspot, the main theme of the task. The user clicked on some results and spent relatively longer time on each of these queries, indicating they might have acquired some useful information, and the removed words might have successfully served their purposes.

Example 2			#click	time
1	what are sunspots	-	1	93s
2	are sunspots a **new phenomena**	DSAT	0	14s
3	when were sunspots **first observed**	SAT	2	103s
4	are sunspots **random** or **patterned**	SAT	2	150s
5	sunspots and **earths climate**	SAT	1	142s
6	sunspots **11 year cycle**	SAT	1	94s
7	sunspots magnetic fields	-	1	-

In contrast, DSAT remove stands for the case that a word did not fulfill the searcher's purpose of using it in the query. The query usually retrieves low quality results, such that the user did not click on any results and spent only a short duration on the SERP. In the second query of Example 2, *new* and *phenomena* were labeled DSAT removes. The searcher showed a similar intent in the second and the third queries, but did not click on any results for the second query. We examined the second query's SERP and found that the retrieved results seem not useful—none include both *new* and *phenomena* in the summaries. Therefore, the user rephrased the query and removed the two words in the next query.

The greater popularity of SAT remove in the collected data explains why results in Table 2 suggest that users tend to remove a word when they spent a relatively longer time on the current query (`q_duration`) and clicked on its results (`q_clickpos`). However, we did also observe a substantial number of DSAT removes (35%). This is concealed by the positive effects of `q_duration` and `q_clickpos`.

5.5 Summary

To summarize, results in this section suggest that users remove a word mainly for two reasons. Firstly, the word has already fulfilled its purpose and becomes less useful (SAT remove)—users need to remove the word in query reformulation in order to move forward in the session. This is supported by: users spent a longer time on the current query (`q_duration`) and clicked on results on the current query's SERP (`q_clickpos`) before they remove a word; removed words appeared less frequently, but still quite often, in the retrieved results (`#skip_snippet_hasw`; searchers are less likely to remove a word if it is related to the main theme of the task (`p(w|pastq)`). Secondly, the word caused limited search performance in the current query (DSAT remove). This conflicts with the positive effects of `q_duration` and `q_clickpos`, but is supported by the qualitative analysis. DSAT remove is also consistent with the observed effects of `avg_jaccard_log` and `#skip_snippet_hasw`. Moreover, removing a word may also be influenced by individual preference in query reformulation (`avg_num_rmv`), query length (`q_length`), overall search performance of the session (`num_query`), and the amount of acquired relevant information in the session (`duration` and `num_click`).

6. ADDING A WORD

6.1 Candidate Sets

For a query reformulation $q_i \rightarrow q_{i+1}$, we examine two types of added words: brand-new word ($\mathrm{Y_{new}}$) that was not used in any previous queries; used words ($\mathrm{Y_{reuse}}$) that was included in past queries (from q_1 to q_{i-1}), but removed in the current query (q_i). We separately study these two cases, because they may stand for different intentions of users.

We can observe all added words in query reformulations, but it is difficult to determine the words that users considered but did not added. We refer to the set of words that users consider for whether or not to add to the new query as *candidate set*. In this study, we implement the candidate sets for $\mathrm{Y_{new}}$ and $\mathrm{Y_{reuse}}$ as follows:

The candidate set for $\mathrm{Y_{reuse}}$ includes all words in previous queries (from q_1 to q_{i-1}) excluding those in the current query (q_i). This is to assume that during a query reformulation, users reconsider each previously used word that was excluded in the current query, and decide whether or not to reuse it in the next query. For each session, we apply this rule to extract $\mathrm{Y_{reuse}}$'s candidate set for query reformulations starting from $q_2 \rightarrow q_3$. On average the candidate set for $\mathrm{Y_{reuse}}$ includes 4.4 words. This yields 668 observations of $\mathrm{Y_{reuse}}$ among 151 query reformulations, where 53 (7.9%) are positive ($\mathrm{Y_{reuse}} = 1$).

The candidate set for $\mathrm{Y_{new}}$ includes words from three sources: task description, result summaries (both titles and snippets) viewed by the user, and query suggestions viewed by the user. Here we only consider result summaries and query suggestions displayed on the current query's SERP. By definition, the candidate set for $\mathrm{Y_{new}}$ excludes words from q_1 to q_i. This is to assume that in a query reformulation, users consider whether or not to add new words related to the task (in task description) and those they recently viewed in result summaries and query suggestions. We extract candidate sets for each query reformulation. On average the candidate set has 71.0 words. In total, we extract 14,413 observations from 203 query reformulations, where 152 (1.05%) are positive ($\mathrm{Y_{new}} = 1$).

As Table 4 shows, $\mathrm{Y_{new}}$'s candidate set (all three sources) covers about half (54.9%) of the observed added new words. Task description and viewed result summaries are the major two sources of added new words. In contrast, query suggestions viewed by the users include only 1.3% of the added new words. In this study, we restrict our scope to this candidate set when examining $\mathrm{Y_{new}}$. We do not consider words from clicked result web pages, because at the time of the user experiment [13], we did not store a copy of the visited web pages for that moment. This is a limitation of our study.

6.2 Models

We estimate models for $\mathrm{Y_{new}}$ and $\mathrm{Y_{reuse}}$ separately. Table 5 reports the results. Similarly, Model 1 includes only session-level variables and constant, Model 2 further includes query-level variables, and Model 3 uses all variables. We only report coefficients of variables for Model 3 due to limited space. For Model 1 and Model 2, we only report changes in -2 log likelihood (LL) and results for the Omnibus tests.

Table 4: Percentage of new words in query reformulations found in different sources.

Source	Percentage
Task description	43.3%
Result summaries viewed by the user	26.7%
Query suggestions viewed by the user	1.3%
All three sources	54.9%
Result titles viewed by the user	15.2%
Result snippets viewed by the user	23.5%

For $\mathrm{Y_{new}}$, we exclude `p(w|pastq)` from independent variables because by definition, all words in the candidate sets should not appear in previous queries. For both $\mathrm{Y_{new}}$ and $\mathrm{Y_{reuse}}$, we exclude `avg_num_rmv` to avoid serious multicollinearity issues (VIF ≥ 5), because it has a high correlation with `avg_num_add` ($r = 0.85$ for $\mathrm{Y_{new}}$ and $r = 0.91$ for $\mathrm{Y_{reuse}}$). Similarly, we exclude `click_url_hasw` ($r = 0.71$ with `click_title_hasw`) and `skip_url_hasw` ($r = 0.87$ with `skip_title_hasw`) in $\mathrm{Y_{reuse}}$. The variables included in the reported models all satisfy VIF < 5.

6.3 Influence of Session-level Variables

Results in Table 5 suggest that session-level variables have no significant influence on adding a word or not in query reformulation. This is consistent for both adding a new word ($\mathrm{Y_{new}}$) and reusing a word ($\mathrm{Y_{reuse}}$). None of the session-level variables show any significant effects on $\mathrm{Y_{new}}$ or $\mathrm{Y_{reuse}}$ in Model 1, Model 2, or Model 3. In addition, for both $\mathrm{Y_{new}}$ and $\mathrm{Y_{reuse}}$, Model 1 cannot explain the collected data significantly better than baseline models involving only constant ($p = 0.073$ for $\mathrm{Y_{new}}$ and $p = 0.122$ for $\mathrm{Y_{reuse}}$). This indicates that the included task and user characteristics may not directly affect users' decisions on whether or not to add a word in query reformulation.

6.4 Influence of Query-level Variables

Results show that the query-level variables have limited influence on adding a word or not in the next query. This applies to both adding a new word ($\mathrm{Y_{new}}$) and reusing a word ($\mathrm{Y_{reuse}}$). As Table 6 shows, only one query-level variables shows a significant effect on adding new words ($\mathrm{Y_{new}}$) in Model 3, and none have any significant effects on reusing ($\mathrm{Y_{reuse}}$). For $\mathrm{Y_{new}}$, Model 2 significant outperforms Model 1 at 0.05 level, but the magnitude of change in -2 log likelihood is small. For $\mathrm{Y_{reuse}}$, Model 2 does not significantly improve over Model 1 ($p = 0.172$). Even taking into account the limited sample size for $\mathrm{Y_{reuse}}$, we believe results suggest limited influence of the query-level variables on adding a new word and reusing a word in query reformulation.

The length of the current query (`q_length`) shows a significant negative effect on adding a new word ($\mathrm{Y_{new}}$) in Model 3. This is not surprising because longer queries are usually more specific. Adding a new word can make it overspecific.

6.5 Influence of Word-level Variables

Word-level variables show relatively strong influence on whether or not to add a new word to the next query ($\mathrm{Y_{new}}$),

Table 5: Results from hierarchical logistic regressions: $\mathbf{Y_{new}}$ and $\mathbf{Y_{reuse}}$ as dependent variables.

Variable Name	$\mathbf{Y_{new}}$: Adding a New Word				$\mathbf{Y_{reuse}}$: Reusing a Word			
	exp(B)		**95% CI**		**exp(B)**		**95% CI**	
`constant`	**0.048**	***	-	-	**0.006**	**	-	-
`product (intellect)`	0.916		0.618	1.357	1.031		0.337	3.153
`goal (amorphous)`	0.801		0.557	1.153	1.155		0.403	3.313
`familiarity`	1.059		0.909	1.233	0.847		0.566	1.267
`avg_num_add`	1.317		0.899	1.930	2.703		0.799	9.143
`q_length (log)`	**0.615**	*	0.426	0.890	0.913		0.487	1.712
`q_duration (log)`	1.051		0.824	1.339	1.386		0.878	2.188
`q_clickpos`	1.031		0.961	1.106	0.902		0.731	1.114
`num_query`	1.009		0.925	1.101	1.067		0.946	1.205
`num_click`	0.958		0.898	1.023	0.980		0.821	1.168
`duration`	0.952		0.867	1.046	0.851		0.681	1.064
`idf`	1.092		0.990	1.203	1.020		0.860	1.209
`p(w|pastq)`	-		-	-	**4.663**	*	1.003	21.68
`avg_jaccard (log)`	**1.395**	***	1.202	1.619	0.965		0.740	1.259
`#click_title_hasw`	**2.399**	***	1.688	3.410	1.693		0.458	6.253
`#click_snippet_hasw`	**0.688**	*	0.494	0.957	2.253		0.891	5.694
`#click_url_hasw`	1.281		0.957	1.717	-		-	-
`#skip_title_hasw`	**1.442**	*	1.045	1.990	0.824		0.443	1.530
`#skip_snippet_hasw`	**0.752**	*	0.575	0.985	1.364		0.821	2.268
`#skip_url_hasw`	0.775		0.548	1.097	-		-	-
`freq_w_suggest`	0.317		0.049	2.058	3.152		0.810	12.27
−2 LL & Omnibus Tests, Model 1	1686.2 → 1677.6		$p = 0.073$		370.3 → 363.0		$p = 0.122$	
−2 LL & Omnibus Tests, Model 2	1677.6 → 1666.8		$p = 0.036$ *		363.0 → 356.3		$p = 0.172$	
−2 LL & Omnibus Tests, Model 3	1666.8 → 1598.1		$p < 0.001$ ***		356.3 → 339.5		$p < 0.030$ *	

*, **, and *** indicate differences are significant at 0.05, 0.01, and 0.001 levels, respectively.

and they also show a clear influence on reusing words in query reformulation ($\mathrm{Y_{reuse}}$). After including the word-level variables, Model 3 for both $\mathrm{Y_{new}}$ and $\mathrm{Y_{reuse}}$ significantly improve over Model 2 ($p < 0.001$ for $\mathrm{Y_{new}}$, and $p < 0.030$ for $\mathrm{Y_{reuse}}$). This indicates that the word-level variables are more salient factors for adding a word in query reformulation compared with the session and query-level variables.

Results suggest that users are more likely to add new words that co-occur frequently with existing words in the current query (`avg_jaccard` shows a significant positive effect on $\mathrm{Y_{new}}$). This is not surprising because words with low `avg_jaccard` values are more likely off-topic and may retrieve low quality results.

Results also show that users are more likely to add a new word to the next query if it appeared frequently in result titles they viewed on the current query's SERP, regardless of whether or not they clicked on the results (both `#click_title_hasw` and `#skip_title_hasw` show significant positive effects on $\mathrm{Y_{new}}$). On the contrary, users are less likely to add a new word if it appeared frequently in the result snippets they viewed on the current query's SERP (both `#click_title_hasw` and `#skip_title_hasw` show significant negative effects on $\mathrm{Y_{new}}$). Post-hoc analysis (Table 6) also agrees with these trends. The new words added to the next query ($\mathrm{Y_{new}} = 1$) appeared in significantly more clicked ($p < 0.01$) and skipped result titles ($p < 0.05$) compared with other words in the candidate sets ($\mathrm{Y_{new}} = 0$). The added new words also appeared in significantly fewer skipped result snippets ($p < 0.05$), although the difference is not significant for clicked result snippet.

Table 6: Mean values (S.E.) of variables for new words added and not added to the next query.

Variables	$\mathbf{Y_{new} = 1}$	$\mathbf{Y_{new} = 0}$	
`#click_title_hasw`	0.243 (0.054)	0.079 (0.003)	**
`#click_snippet_hasw`	0.270 (0.058)	0.286 (0.005)	
`#click_url_hasw`	0.224 (0.050)	0.108 (0.004)	*
`#skip_title_hasw`	0.257 (0.056)	0.186 (0.004)	*
`#skip_snippet_hasw`	0.487 (0.074)	0.605 (0.006)	**
`#skip_url_hasw`	0.191 (0.047)	0.221 (0.006)	

*, **, and *** indicate significance at 0.05, 0.01, and 0.001 levels, respectively, by a two-tail Welch's t-test.

This indicates that the occurrences of a new word in result titles may play a more important role on users decisions of whether or not to add the word to the next query. This is probably because result titles are more eye-catching than other elements on the SERP, as most current search engines show result titles using a larger font size than other SERP elements. However, as Table 4 shows, we can only locate 15.2% of the added new words in result titles, in contrast to 23.5% in snippets and 26.7% in summaries (both titles and snippets). This indicates result snippets still provide valuable ideas for new words in query reformulation, but the new words do not necessarily appear more often than other words in result snippets. In fact, both results in Table 5 and Table 6 suggest that they appear less often in snippets compared with other words that were not added to the next query. This is probably because result snippets are noisy, including both relevant and off-topic words.

In contrast, only `p(w|pastq)` (how frequently the word was used in past queries of the session) shows a significant effect on reusing a word ($\mathrm{Y_{reuse}}$). None of the other word-level variables show any significant effects. This indicates that while reusing a word, users usually simply reuse the word associated to the main theme of the task.

We manually examined the cases of reusing a word in the collected data. We found that reusing a word in query reformulation mostly happens when users revert from a subtask to the main task, or to another subtask. The following table shows an example. The user first explored differences in dehumidifiers in the first two queries, and then switched to look for information related to hygrometer in the third and the fourth queries. After finishing exploring hygrometer, the user reverted back to continue to explore dehumidifier and thus reused the word **dehumidifier** in the fifth query. This example explains why users are more likely to reuse words related to the main theme of the task (with high `p(w|pastq)` values).

No.	Query
1	differences in dehumidifier
2	differences in dehumidifier 500 sq ft room
3	hygrometer
4	hygrometer amazon
5	**dehumidifier** ACH

6.6 Summary

To summarize, results in this section suggest that adding a brand-new word ($\mathrm{Y}_{\mathrm{new}}$) and reusing a word ($\mathrm{Y}_{\mathrm{reuse}}$) stand for different intentions of users. Such distinctions were not identified in previous studies. Users exploit highly related, unused words from the results they viewed and include them into the next query, as suggested by the positive effects of `avg_jaccard`, `#click_title_hasw`, and `#skip_title_hasw`. In contrast, they reuse a word when reverting from a subtask to the main task or another subtask, as suggested by the positive effect of `p(w|pastq)` and manual analysis.

Compared with removing or retaining a word, results show that adding a word is less likely influenced by the session and query-level variables, as suggested by the limited effects of session and query-level variables on both $\mathrm{Y}_{\mathrm{new}}$ and $\mathrm{Y}_{\mathrm{reuse}}$. This indicates that users may make decisions on whether or not to add a word mostly based on local factors that are directly related to the word itself. Such decisions are less likely influenced by task characteristics or past user activities in the session.

7. CONCLUSION

7.1 Findings and Implications

Using both hierarchical logistic regression and qualitative analysis, this paper provides insights on how different factors may affect specific choices of word changes in query reformulations during a task-based search session. This advances the state-of-the-art understanding of query reformulation from query-level patterns [1, 5, 11, 12, 26] to word-level and finer-grained users decisions related to specific words.

Results suggest that word-level variables may strongly influence all three types of word changes. This is not surprising because word-level variables are word-specific, while other two types of factors are not. In contrast, query-level variables only show certain influence on removing or retaining a word, but limited influence on adding and reusing a word. This indicates that removing or retaining a word may more likely be affected by past user interaction and situation in a session, while adding and reusing a word may not. Session-level variables exhibited limited *direct* influence on all types of word changes. However, we believe they still influence word changes in a session in an *indirect* way. This is based on the fact that much previous work found that task type and user characteristics can affect search behavior patterns in a session [6, 7, 13, 20, 24]. Therefore, session-level variables may affect query and word-level variables and consequently influence word changes in query reformulation.

Comparing the three categories of factors, word-level factors show the strongest influence on all three types of word changes compared with the other two types of factors. Query-level factors show less influence, and session-level factors only exhibit limited direct influence. Results also suggest that these variables may influence different word changes in different ways. For example, word occurrences in result summaries show different patterns for all three types of word changes. This implies the different nature of these word changes. In addition, our analysis also help identify effective sets of features for predicting such word changes in an on-going search session. Such techniques may potentially help develop and evaluate interactive search systems.

Moreover, our study also discloses typical scenarios for different types of word changes. Users remove a word in query reformulation for two possible reasons. Firstly, it happens when users finished exploring a subtask and move forward to another. Secondly, it also happens when users try to correct bad performing queries (e.g., removing off-topic or ambiguous words). In our collected data, the former is more prevalent. In contrast, adding a new word usually happens in relevance feedback—users exploit related, unused words from recently viewed result summaries and add them to new queries. Reusing a word mostly happens when searchers revert back from a subtask to the main task, and the reused words are highly related to the main theme of the task.

7.2 Limitations

Our study has a few limitations. Firstly, we study three types of word changes and each individual word change separately. This ignores dependencies among word changes—users' decisions to remove or add a word may depend on the removal or addition of another. Whereas analyzing such dependencies may enlarge the problem space exponentially—most notably it may require a substantially greater amount of data to examine these issues.

Secondly, the approach of generating candidate sets, especially that for adding a new word, is limited. This also resulted in a biased dataset for $\mathrm{Y}_{\mathrm{new}}$ and $\mathrm{Y}_{\mathrm{reuse}}$, where over 90% of the cases are negative. This may be one reason for the limited influence of the session and query-level variables in our analysis. However, we also believe that, as long as the influence of a variable is strong enough, it should still be able to show a significant effect. Yet we may have missed a few variables with slight or moderate effects on word changes.

Thirdly, it should be noted that the search tasks searchers performed in the user study are typically more complex than daily web search information needs. For example, no simple tasks such as navigational searches were included. Results and findings from this study should be generalized with cautious to other tasks with a substantially different nature.

8. ACKNOWLEDGMENT

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-0910884. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor. We thank the anonymous reviewers for their thoughtful and constructive comments.

9. REFERENCES

[1] P. Anick. Using terminological feedback for web search refinement: A log-based study. In *SIGIR ’03*, pages 88–95, 2003.

[2] L. Azzopardi. Modelling interaction with economic models of search. In *SIGIR ’14*, pages 3–12, 2014.

[3] M. J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online review*, 13(5):407–424, 1989.

[4] N. J. Belkin, R. N. Oddy, and H. M. Brooks. ASK for information retrieval: Part I. background and theory. *Journal of Documentation*, 38(2):61–71, 1982.

[5] P. Bruza and S. Dennis. Query reformulation on the internet: Empirical data and the hyperindex search engine. In *RIAO ’97*, pages 488–499, 1997.

[6] M. J. Cole, J. Gwizdka, C. Liu, R. Bierig, N. J. Belkin, and X. Zhang. Task and user effects on reading patterns in information search. *Interacting with Computers*, 23(4):346–362, 2011.

[7] M. J. Cole, C. Hendahewa, N. J. Belkin, and C. Shah. Discrimination between tasks with user activity patterns during information search. In *SIGIR ’14*, pages 567–576, 2014.

[8] V. Dang and B. W. Croft. Query reformulation using anchor text. In *WSDM ’10*, pages 41–50, 2010.

[9] A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models.* Cambridge University, 2006.

[10] D. Guan, S. Zhang, and H. Yang. Utilizing query change for session search. In *SIGIR ’13*, pages 453–462, 2013.

[11] J. Huang and E. N. Efthimiadis. Analyzing and evaluating query reformulation strategies in web search logs. In *CIKM ’09*, pages 77–86, 2009.

[12] B. J. Jansen, D. L. Booth, and A. Spink. Patterns of query reformulation during web searching. *Journal of the American Society for Information Science and Technology*, 60(7):1358–1371, 2009.

[13] J. Jiang, D. He, and J. Allan. Searching, browsing, and clicking in a search session: Changes in user behavior by task and over time. In *SIGIR ’14*, pages 607–616, 2014.

[14] R. Jones and K. L. Klinkner. Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs. In *CIKM ’08*, pages 699–708, 2008.

[15] E. Kanoulas, B. Carterette, M. Hall, P. Clough, and M. Sanderson. Overview of the TREC 2012 session track. In *the Twenty-First Text REtrieval Conference (TREC 2012) Proceedings*, 2012.

[16] D. Kelly, J. Arguello, A. Edwards, and W.-C. Wu. Development and evaluation of search tasks for IIR experiments using a cognitive complexity framework. In *ICTIR ’15*, pages 101–110, 2015.

[17] D. Kelly, K. Gyllstrom, and E. W. Bailey. A comparison of query and term suggestion features for interactive searching. In *SIGIR ’09*, pages 371–378, 2009.

[18] J. Koenemann and N. J. Belkin. A case for interaction: A study of interactive information retrieval behavior and effectiveness. In *CHI ’96*, pages 205–212, 1996.

[19] Y. Li and N. J. Belkin. A faceted approach to conceptualizing tasks in information seeking. *Information Processing & Management*, 44(6):1822–1837, 2008.

[20] C. Liu, N. J. Belkin, and M. J. Cole. Personalization of search results using interaction behaviors in search sessions. In *SIGIR ’12*, pages 205–214, 2012.

[21] C. Liu, J. Gwizdka, and N. J. Belkin. Analysis of query reformulation types on different search tasks. In *iConference 2010*, pages 477–485, 2010.

[22] J. Liu and N. J. Belkin. Personalizing information retrieval for multi-session tasks: The roles of task stage and task type. In *SIGIR ’10*, pages 26–33, 2010.

[23] J. Liu and N. J. Belkin. Personalizing information retrieval for multi-session tasks: Examining the roles of task stage, task type, and topic knowledge on the interpretation of dwell time as an indicator of document usefulness. *Journal of the Association for Information Science and Technology*, 66(1):58–81, 2015.

[24] J. Liu, M. J. Cole, C. Liu, R. Bierig, J. Gwizdka, N. J. Belkin, J. Zhang, and X. Zhang. Search behaviors in different task types. In *JCDL ’10*, pages 69–78, 2010.

[25] S. Menard. *Applied logistic regression analysis, Second Edition.* SAGE publications, 2002.

[26] S. Y. Rieh and H. Xie. Analysis of multiple query reformulations on the web: The interactive information retrieval context. *Information Processing & Management*, 42(3):751–768, 2006.

[27] C. L. Smith and P. B. Kantor. User adaptation: Good results from poor systems. In *SIGIR ’08*, pages 147–154, 2008.

[28] K. Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[29] A. Spink, M. Park, B. J. Jansen, and J. Pedersen. Multitasking during web search sessions. *Information Processing & Management*, 42(1):264–275, 2006.

[30] A. Spink and T. Saracevic. Interaction in information retrieval: Selection and effectiveness of search terms. *Journal of the American Society for Information Science*, 48(8):741–761, 1997.

[31] P. Vakkari. Task-based information searching. *Annual Review of Information Science and Technology*, 37(1):413–464, 2003.

[32] W.-C. Wu, D. Kelly, A. Edwards, and J. Arguello. Grannies, tanning beds, tattoos and NASCAR. In *IIIX ’12*, pages 254–257, 2012.

[33] Z. Yue, S. Han, D. He, and J. Jiang. Influences on query reformulation in collaborative web search. *Computer*, 47(3):46–53, 2014.