

When do people use query suggestion? A query suggestion log analysis

Makoto P. Kato · Tetsuya Sakai · Katsumi Tanaka

Received: 6 September 2012 / Accepted: 5 December 2012 / Published online: 3 January 2013
© Springer Science+Business Media New York 2012

Abstract Query suggestion, which enables the user to revise a query with a single click, has become one of the most fundamental features of Web search engines. However, it has not been clear what circumstances cause the user to turn to query suggestion. In order to investigate when and how the user uses query suggestion, we analyzed three kinds of data sets obtained from a major commercial Web search engine, comprising approximately 126 million unique queries, 876 million query suggestions and 306 million action patterns of users. Our analysis shows that query suggestions are often used (1) when the original query is a rare query, (2) when the original query is a single-term query, (3) when query suggestions are unambiguous, (4) when query suggestions are generalizations or error corrections of the original query, and (5) after the user has clicked on several URLs in the first search result page. Our results suggest that search engines should provide better assistance especially when rare or single-term queries are input, and that they should dynamically provide query suggestions according to the searcher's current state.

Keywords Query suggestion · Query log analysis · Web search

1 Introduction

Although continuing advances in Web search technologies provide efficient and effective information access to users, users are still finding it difficult to express their information needs in the form of queries. A study on Excite query logs reported that 52.0, 39.6, and

M. P. Kato (✉) · K. Tanaka
Department of Social Informatics, Graduate School of Informatics, Kyoto University, Yoshida
Honmachi, Sakyo, Kyoto 606-8501, Japan
e-mail: kato@dl.kuis.kyoto-u.ac.jp

K. Tanaka
e-mail: tanaka@dl.kuis.kyoto-u.ac.jp

T. Sakai
Microsoft Research Asia, No. 5, Dan Ling Street, Haidian District, Beijing 100080, China
e-mail: tetsuyasakai@acm.org

44.6% of users (in 1997, 1999 and 2001, respectively) modified their queries (Spink et al. 2002). More recently, a study on Dogpile¹ query data from 2005 (Jansen et al. 2009) and one on the AOL query logs from 2006 (Huang and Efthimiadis 2009) have reported that approximately 28 % of queries were modified.

One of the reasons why queries are modified so often may be that queries are often *informational*: Broder (2002) reported that about half of the AltaVista queries he analyzed were informational, as opposed to *navigational* (e.g. looking for a particular home page) and *transactional* (e.g. trying to buy a ticket). Using this query taxonomy, Jansen et al. (2008) automatically classified Dogpile queries and found that around 80 % of them were informational. With informational queries, users look for various aspects of information on a particular topic, and therefore may try out different query terms. Moreover, the users may not know the effective query terms that can retrieve the required information, as they may lack detailed knowledge on that topic. Interactive and exploratory search systems (Marchionini 2006; White et al. 2006) may help the user formulate effective queries.

As a simple form of interactive search, query suggestion has become one of the most fundamental features of commercial Web search engines. Given a list of query suggestions, the user can simply click on one of them to initiate a new search. Providing effective query suggestions to the user is very important for helping the user express his information need precisely so that he can access the required information. Hence, many studies on effective query suggestion based on clickthrough and query session data have been reported (Anagnostopoulos et al. 2010; Baeza-Yates et al. 2004; Beeferman and Berger 2000; Boldi et al. 2008; Cao et al. 2008; He et al. 2009; Ma et al. 2008; Mei et al. 2008; Song and He 2010). However, how people use query suggestions is not well understood. We believe that analysis of this kind is necessary in order to effectively assist the user depending on situations.

The objective of the present study is to clarify when and how query suggestions are used by Web search users. Whether a user uses a query suggestion is basically dependent on three factors: an input query, presented query suggestions, and the user's actions before he uses query suggestion. Our approach is to examine real data by considering all of these three factors. Thus, we analyzed three kinds of data sets obtained from a major commercial Web search engine, comprising approximately 126 million unique queries, 876 million query suggestions and 306 million action patterns of users. Our analysis shows that query suggestions are often used

1. when the original query is a rare query;
2. when the original query is a single-term query;
3. when query suggestions are unambiguous;
4. when query suggestions are generalizations or error corrections of the original query; and
5. after the user has clicked on several URLs in the first search result page.

Findings 1 and 2 come from our analysis of query data; findings 3 and 4 come from our analysis of query suggestion data; and finding 5 comes from our analysis of user action data.

Our results suggest that search engines should provide better assistance especially when rare or single-term queries are input, and that they should dynamically provide query suggestions according to the searcher's current state, for example, whether s/he is on the first search result page, whether s/he has clicked on several URLs so far, and so on.

¹ <http://www.dogpile.com/>.

The remainder of this paper proceeds as follows. Section 2 briefly surveys prior art in query suggestion and manual query reformulation. Section 3 introduces some definitions of terminology used in this paper, and Sect. 4 describes three kinds of data sets obtained from a commercial Web search engine. Section 5 presents the results of our analyses, and Sect. 6 discusses some practical implications of our findings. Finally, Sect. 7 concludes this paper and discusses future work.

2 Related work

This section briefly surveys prior art in query suggestion and manual query reformulation. Query suggestion, a common feature in commercial Web search engines, enables the user to revise a query with a single click by providing him with a small list of clickable candidates. Whereas, manual query reformulation refers to the action of the user who modifies his query by hand (e.g. by typing or copy-and-paste). Hence, in our view, *query reformulation* subsumes both query suggestion and manual query reformulation.

2.1 Query suggestion

Recently, various kinds of methods have been proposed for generating query suggestions, many of which use clickthrough or query session data. Clickthrough data consist of queries and URLs that were clicked within the result pages (*clicked URLs*), while query session data consist of queries, their timestamp, and session identifier.

One approach to generating query suggestions is clustering queries based on their clicked URLs (Baeza-Yates et al. 2004; Beeferman and Berger 2000; Cao et al. 2008). Given a query, query suggestions are selected from the query cluster to which the original query belongs. One of the earliest work by Beeferman and Berger incorporated the co-occurrence of clickthrough to cluster queries, and suggested queries in a cluster to which an input query belongs (Beeferman and Berger 2000). The quality of the query suggestions was evaluated by the clickthrough rate on the live Lycos search engine. Another approach is incorporating a random walk or hitting time in a query and clicked URL bipartite graph (Ma et al. 2008; Mei et al. 2008; Song and He 2010). He et al. (2009) showed the advantage of query suggestion based on user query sequences over that based on query pairs. Other studies also utilized query session data (Anagnostopoulos et al. 2010; Boldi et al. 2008; Song et al. 2012). In contrast to the aforementioned studies that focused on providing *new* queries, some studies tackled the query suggestion problem by merely substituting or stemming terms (Dang and Croft 2010; Jones et al. 2006; Kraft and Zien 2004; Wang and Zhai 2008). A few recent studies tackle the problem of diversifying query suggestions by diversifying their top-returned search results (Ma et al. 2010; Song et al. 2011). Some other work is introduced in Sect. 4.3 of Silvestri's book (Silvestri 2010).

To our knowledge, only a few studies on the usage of query suggestion have been reported in the literature, and they are rather limited in scale. Kelly et al. (2010) investigated the effect of presenting the usage statistics of each query suggestion to the user. Their experiments used four topics, each with eight query suggestions. Each query suggestion was accompanied by information as to how many other people used that suggestion. Four of the query suggestions were frequently-used suggestions, and the remaining four were infrequently-used ones. They found that subjects were not influenced by these usage statistics. Kelly et al. (2009) studied the difference between *term suggestion* and query suggestion. A term suggestion system they developed enables the user to add terms to his

original query by clicking on each suggestion term. In their interactive information retrieval study with 55 subjects and 20 topics, subjects preferred query suggestion to term suggestion. White et al. (2007) studied the use of query suggestions and *destination suggestions* (suggesting Web pages at which users finish their search frequently). In their user study, 36 subjects were asked to use systems suggesting queries or destinations to conduct two types of tasks: known-item task (finding items of information for which the target was well-defined) and exploratory task (gathering background information on a topic or gathering sufficient information to make a required decision). Their finding is that systems presenting query suggestions were preferred for known-item tasks, while systems offering destinations were preferred for exploratory tasks. Kato et al. (2012) proposed a structured query suggestion interface and its back-end algorithm leveraging query log data. Through a task-based user study with 20 subjects and 20 topics, they demonstrated that query suggestion interfaces can affect the user's search performance and perception. However, while all of the above studies conducted user-based evaluations under control, how *real* search engine users, who search for their own purpose in daily life, use query suggestion has not been clarified.

There are also existing studies on search functionalities that are related to query suggestion but are different: namely, *interactive query expansion* (Anick 2003) and *query completion* (Amin et al. 2009; Kamvar and Baluja 2008; White and Marchionini 2007). Interactive query expansion is basically the same as the aforementioned term suggestion, but it appears that interactive query expansion has been replaced by query suggestion during the last decade. In contrast, query completion is as common a feature as query suggestion in current search engines and they probably complement each other: while query suggestion provides a static list of possible queries *given a complete initial query*, query completion aims to provide a dynamic list of possibilities *given a prefix string of an initial query*, often within the search query box.

2.2 Manual query reformulation

There are two prominent approaches to providing a taxonomy for manual query reformulation: lexical (Huang and Efthimiadis 2009; Teevan et al. 2007) and semantic categorization (Boldi et al. 2009; Jansen et al. 2009; Lau and Horvitz 1999; Rieh and Xie 2006).

Huang and Efthimiadis (2009) took a lexical categorization approach. They proposed a query reformulation taxonomy based on word removal, word addition, word substitution, abbreviation, and so on. Using the taxonomy, they applied a rule-based classifier to an AOL query log. One of the findings from their experiment was that the types of query reformulations that are likely to be effective depend on how *successful* the previous query was, where a successful query is one that is followed by at least one URL click. For example, after a successful query, a query reformulation by word substitution is also likely to be successful. Whereas, after an unsuccessful query (with no URL clicks), a query reformulation by spelling correction is likely to be successful.

On the other hand, Boldi et al. (2009) focused on a semantic categorization of manual query reformulations. Their categories were derived from previous work (Rieh and Xie 2006), and adopted four categories, i.e. *generalization*, *specialization*, *error correction*, and *parallel movement*. Generalization means that the reformulated query represents a broader concept than the original query does; specialization conversely means that the reformulated query represents a narrower concept; error correction means that the user is correcting a typographical error or trying an alternative spelling or uppercase/lowercase

variants; and parallel movement means that the reformulated query shares the same topic with the original query but focuses on a different aspect. They built a classifier trained by using labeled pairs of queries, and applied it to automatically classify a large query log. Their experiment based on a Yahoo! search query log from 2008 showed that the distributions of manual query reformulations over the four categories were: parallel movements (48–56 %), specializations (30–38 %), error corrections (5–10 %), and generalizations (4–10 %).

3 Definitions

This section provides definitions of the terminology used throughout the paper.

Queries and query suggestions A query is a set of terms that triggers a search and produces a search engine result page: it may be a manually-input query, or a query suggestion that has been clicked. However, we simply refer to the current query (often a manually-input initial query) as *query* and to the new query suggestion in the result page as *query suggestion*, *QS* or simply *suggestion*.

Qslist_shown and Qslist_used A query suggestion list (Qslist) is a *list* of query suggestions shown in a result page produced in response to a given query. For each query that has a QS list, *Qslist_shown* is defined as the number of times the Qslist was shown to the user in response to that query. In other words, *Qslist_shown* is the query frequency with its suggestions shown. As 87.3 % of user's queries are accompanied by a Qslist, *Qslist_shown* nearly equals the input frequency of queries. Whereas, *Qslist_used* is defined as the number of times the Qslist was utilized by the user, i.e. *at least one query suggestion* in the Qslist was clicked by the user. Note that these statistics are collected for each query, not for individual query suggestions.

3.1 Query CTR (clickthrough rate)

Query CTR is one of the main metrics used in this paper. We define Query CTR as:

$$\text{Query CTR} = \frac{\text{Qslist_used}}{\text{Qslist_shown}}.$$

Query CTR can be interpreted as the probability that a user uses at least one of the query suggestions given in response to a query. By regarding the existence of clicks in the Qslist as success and lack of clicks as failure, we can assume that *Qslist_used* follows a binomial distribution, with *Qslist_shown* as the number of independent experiments. Under this assumption, Query CTR as defined above approximately follows a Gaussian distribution. Hence, for average Query CTR, we conduct significance testing at 99 % and compute confidence intervals at 99 % using a Student's *t* distribution. Confidence intervals are shown together with Query CTRs in this paper.

3.2 QS_shown and QS_clicked

QS_shown is the number of times a *particular query suggestion* for a query is shown to the user, and *QS_clicked* is the number of times that query suggestion is actually clicked by the user. Note that *QS_shown* and *QS_clicked* are computed for every ⟨query, query suggestion⟩ pair, while *Qslist_shown* and *Qslist_used* are computed for every query.

3.3 Suggestion CTR

Suggestion CTR is another main metric used in this paper. We define Suggestion CTR as:

$$\text{SuggestionCTR} = \frac{\text{QS_clicked}}{\text{QS_shown}}$$

Suggestion CTR is the probability that a user uses a suggestion when it is shown in response to a query. A confidence interval of average Suggestion CTR is obtained in the same way as Query CTR.

3.4 Action patterns

An *action* represents a user’s click on a particular area in the search engine result page. As shown in Fig. 1, **Result** represents a click on a search result; **Page** represents a click to move to another result page (e.g. “next” and “previous”); **Ads** represents a click on a sponsored site link; and **QS** represents a click on a query suggestion. **Other** represents a click on another area such as *deep link* (a link to a certain page on a website, instead of that website’s main page), and some features specific to a search engine.

An *action pattern* is a sequence of actions within a single query session, e.g. **Result** → **Ads** → **Result**, where “→” denotes a transition from one action to another. In our definition of *query session*, every query reformulation initializes a query session, so a **QS** action usually terminates an action pattern. That is, if **QS** appears in an action pattern, it is usually the last action in that pattern. (Exceptions occur when the user clicks a query suggestion but returns to the original result page by clicking the “back” button.)

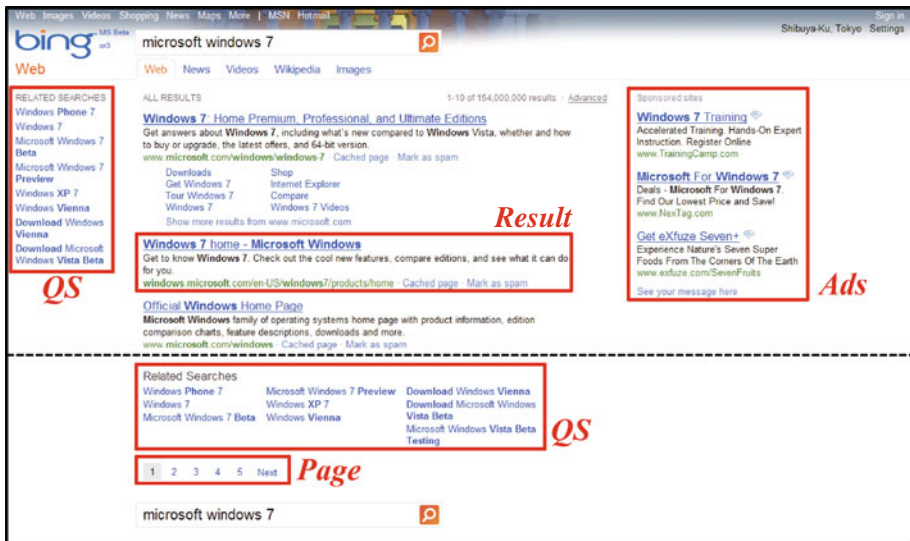


Fig. 1 Actions recorded in log

Table 1 Examples of the **QUERY** data

Query	Qslist_shown	Qslist_used
Japanese restaurant	2,078	252
Live soccer scores	1,260	326
Alzheimer's disease	806	110
What is online marketing	487	5

3.5 Transition ratio

Let $N(x)$ and $N(x \rightarrow y)$ represent the frequency of action x and that of transition $x \rightarrow y$. A conditional probability $P(y|x)$ is defined as $N(x \rightarrow y)/N(x)$, and a prior probability $P(x)$ is defined as $N(x)/N$, where N is the total number of all the actions.

We define the transition ratio from x to y as $P(y|x)/P(y)$. Thus, if this value is >1 , it means that the user is more likely to move from x to y than to move to y unconditionally.

4 Data

We analyzed three types of logs recorded from May 2nd to 8th, 2010 in Microsoft's Bing search engine.² The first log is a query log (**QUERY**) that includes *Qslist_shown* and *Qslist_used* statistics for each query. The second log is a query suggestion log (**SUGGESTION**) that consists of pairs of a query and query suggestion, *QS_shown* and *QS_clicked* statistics for each query suggestion. The third log is an action pattern log (**ACTION**), which contains action patterns per query.

In addition, we collected a query session log (**SESSION**) from October 1st to 10th, 2009 through Microsoft Internet Explorer. This log was used as a supplementary material to show the statistics of user's manual query reformulations and to compare them with those of query reformulations via query suggestion.

The present study thus depends on data from one particular search engine. However, we shall try to observe trends that we believe are general (i.e. they do not depend on Bing's specific features), and also clearly report our methods of analysis so that other research institutions will be able to conduct similar analysis with their own search engines.

4.1 Query log

Our query log (**QUERY**) consists of records whose fields are *query*, *Qslist_shown*, and *Qslist_used*. Some examples of the **QUERY** data are shown in Table 1. The table shows, for example, that the query suggestion list for "Japanese restaurant" was shown to users 2,078 times, of which 252 instances resulted in at least one click. The statistics of the **QUERY** data are shown in Table 2. The *total average Query CTR* ($Qslist_used/Qslist_shown$) is 0.0459, and the average length of queries is 2.673 words.

4.2 Query suggestion log

Our query suggestion log (**SUGGESTION**) contains records whose fields are *query*, *query suggestion*, *QS_shown*, and *QS_clicked*. Some examples of the **SUGGESTION** data are

² <http://www.bing.com/>.

Table 2 Statistics of the **QUERY** data

No. of unique queries	126,496,393
No. of unique queries with Qslist	110,477,998
No. of unique queries with Qslist used	9,013,618
Total Qslist_shown	393,987,833
Total Qslist_used	18,090,921
Total avg. query CTR	0.0459
Avg. length	2.673

shown in Table 3. For example, the query suggestion “Harry Potter Films” was shown to users 68,174 times, of which 190 instances resulted in a click. The statistics of the **SUGGESTION** data are shown in Table 4. The *total average Suggestion CTR* (QS_clicked/QS_shown) is 0.00632. The average query suggestion length is 2.921 words, which is slightly longer than the average query length shown in Table 2 (2.673 words).

4.3 Action pattern log

Our action pattern log (**ACTION**) contains records whose fields are *query*, *action pattern*, and *count*. Some examples of the **ACTION** data are shown in Table 5, and the statistics of the **ACTION** data are shown in Table 6. As shown in the table, the average action pattern length is 1.209 actions. Note that the **ACTION** data contains records for queries with at least one action. Thus, the average length in Table 6 is taken over these records, excluding queries without an action.

4.4 Query session log

Our query session log (**SESSION**) contains triplets comprising a session id (a unique user session id), a query and a timestamp (time when the query was input). Some examples of the **SESSION** data are shown in Table 7, and the statistics are shown in Table 8. Note that this log was collected before the other data sets were recorded and is used as a supplementary material for comparing query reformulations via query suggestion with manual query reformulations.

5 Analysis

This section discusses our five main findings that we mentioned in Sects. 1, 5.1 and 5.2 discuss the types of query that are likely to be followed by a use of query suggestion.

Table 3 Examples of the **SUGGESTION** data

Query	Query suggestion	QS_shown	QS_clicked
Harry Potter	Harry Potter films	68,174	190
Harry Potter	Harry Potter cast	68,174	58
Harry Potter	Harry Potter and the deathly hallows	68,174	13
Harry Potter	Harry Potter and the Sorcerer’s stone	68,174	7
Harry Potter	Daniel Radcliffe	2,189	3

Table 4 Statistics of the **SUGGESTION** data

No. of unique queries	110,805,265
No. of unique suggestions	876,537,724
Total QS_shown	3,572,451,604
Total QS_clicked	18,935,221
Total avg. suggestion CTR	0.00632
Avg. length	2.905

Table 5 Examples of the **ACTION** data

Query	Action pattern	Count
France travel	Result → QS	90
France travel	Other → Result	36
France travel	Page	25
France travel	Result → Ads → Result	23

Table 6 Statistics of the **ACTION** data

No. of unique queries	105,366,789
No. of action patterns	306,820,632
No. of unique action patterns	247,457
Avg. length	1.209
No. of Result	277,552,243
No. of Page	18,875,313
No. of Ads	18,437,315
No. of QS	16,823,503
No. of Other	39,391,980

Table 7 Examples of the **SESSION** data

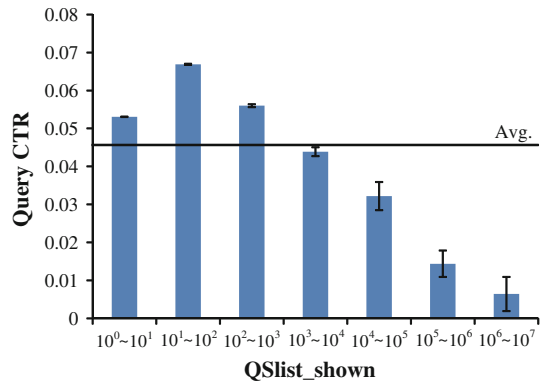
Session ID	Query	Timestamp
15749578	Microsoft	2009/10/01 19:22:01
15749578	Microsoft windows	2009/10/01 19:25:47
15749578	Microsoft windows 7	2009/10/01 19:25:59
24398712	Windows 7	2009/10/01 19:22:11

Table 8 Statistics of the **SESSION** data

No. of queries	22,212,088
No. of unique queries	11,216,354
No. of sessions	774,096

Sections 5.3 and 5.4 discuss the types of query suggestions that are likely to be clicked. Finally, Sect. 5.5 discusses the search context that is likely to be followed by a use of query suggestion.

Fig. 2 QSlist_shown versus Query CTR



5.1 Query suggestion for rare queries

In this section, we show that the popularity of queries input by users is negatively correlated to the clickthrough rate of query suggestions presented by a search engine. Figure 2 shows the average Query CTR (i.e. QSlist_used divided by QSlist_shown) against seven bins of QSlist_shown, with confidence intervals, based on the **QUERY** data. Recall that Query CTR represents the popularity of a query suggestion list as a whole, and that QSlist_shown basically represents query input frequency as was mentioned in Sect. 3. The average CTRs for the range 10^0-10^3 are significantly higher than those for the range 10^4-10^7 .³ Thus, it can be observed that *query suggestions are often used when the original queries are rare queries*.

One possible explanation for the above finding is that, while Web search engines are already effective for popular queries (as search engine companies can leverage a lot of user feedback such as clickthroughs for these queries), they are not effective enough for rare queries and this makes the user turn to query suggestion. Indeed, Downey et al. (2008) reported that Web search engines are less effective for rare queries, and that users who have issued a rare query often try to reformulate it rather than to click URLs in the result page. Our finding seems to be in line with theirs, as query suggestion is a means of query reformulation.

5.2 Query suggestion for single-term queries

This section focuses on another feature of user's input queries, the query length, that affects the clickthrough rate of query suggestions. Figure 3 shows the average Query CTR against query length (i.e. number of query terms), again with confidence intervals, based on the **QUERY** data. The QSlists of single-term queries have an average CTR that is statistically significantly higher than those of other length queries.⁴ That is, *query suggestions are often used after a single-term query*. As for the other query lengths, the average Query CTRs are more or less similar to one another.

³ We conducted Welch's *t* test with Bonferroni correction. There are significant differences in all the bin pairs from the ranges 10^0-10^3 and 10^4-10^7 . The maximum *p* value was obtained between the ranges 10^0-10^1 and 10^6-10^7 : $t(12) = 31.8$, $p < 0.001$.

⁴ Conducting Welch's *t* test with Bonferroni correction, we found significant differences between single-term queries and others. The maximum *p*-value was found between single-term queries and 12-term queries: $t(60880) = 28.2$, $p < 0.001$.

Fig. 3 Query length versus Query CTR

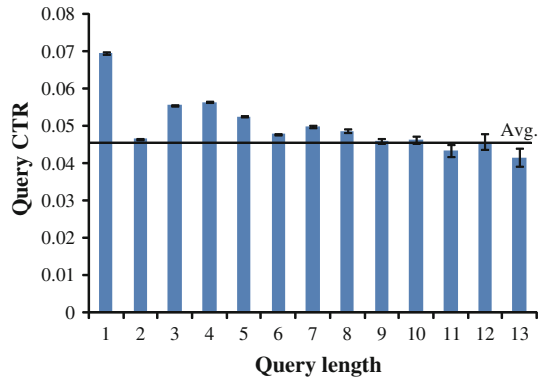


Table 9 Examples of single-term queries

Query	Query suggestion	QS_shown	QS_clicked
Hotmail	Sign in to Hotmail	965,273	4,778
Hotmail	Open my Hotmail account	965,273	1,124
News	CNN news	50,358	1,739
News	Fox news	50,358	1,394
g	Gmail	180,671	193
y	Yahoo.com	101,209	554
y	YouTube videos	101,209	159

Note that the examples are from the **SUGGESTION** data

Table 9 shows some examples of single-term queries. The queries “hotmail” and “news” are underspecified, and require some specifications to identify the target information. We also observed interesting single-character queries such as “g” and “y”. Some users probably utilize those single-characters as a shortcut for navigational queries (e.g. “y” for Yahoo.com), though it is not conclusive as their Suggestion CTRs are low.

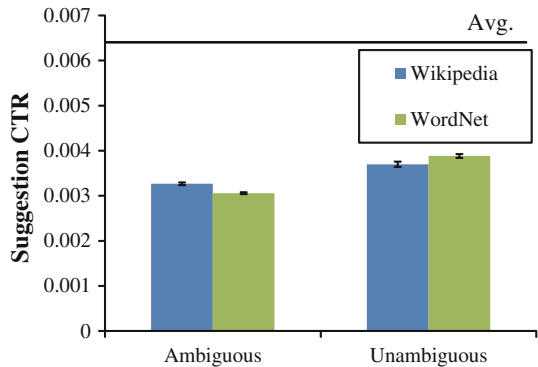
The fact that the CTR is high for single-term queries is quite intuitive. When a user issues a single-term query, he may be feeling that he cannot fully express his information need.⁵ In such a case he may rely on query suggestion. Moreover, for single-term queries, the search results may often be poor. Experiments on document retrieval with a vector space model showed that longer queries for the same intent achieved higher average precision in most of the cases (Cui et al. 2003). Disappointed users may also turn to query suggestion. The findings discussed in this section may also hold for the other languages such as Chinese, Japanese, and Korean, since terms in a Web search query are separated with a whitespace like English queries.

5.3 Unambiguous query suggestions

The previous sections showed that the Query CTR varies with the query frequency and query length. We now examine the properties of *query suggestions* themselves (not the

⁵ The findings observed by Phan et al. (2007) are supportive for our hypothesis. Their experimental result showed a strong correlation between the query length and degree of specificity of an information retrieval request.

Fig. 4 Query suggestion ambiguity versus Suggestion CTR



original query, but ones presented by a search engine) that affect the usage of query suggestion. We first hypothesized that the degree of ambiguity of a query suggestion may affect its popularity, and decided to analyze our **SUGGESTION** data from this viewpoint. One available approach to identifying ambiguous queries is learning query ambiguity models by using query features such as clickthrough and query session data (Song et al. 2010). This approach can be applied to most of the Web queries, while the precision is not yet perfect. Another approach is based on well-structured corpus and thesaurus. Sanderson (2008) used Wikipedia and WordNet to predict the ambiguity of queries. The online encyclopedia, Wikipedia, now provides over 2 million articles on broad topics, and disambiguates an ambiguous entity name by showing a so-called *disambiguation page*. Thus, a query that has a disambiguation page in Wikipedia is likely to be ambiguous. Similarly, if a query has multiple senses in WordNet, it can be regarded as ambiguous. In this study, we follow Sanderson's simple approach.

We downloaded a snapshot of the Wikipedia English language pages on January 15th, 2011, and installed WordNet (version 3.0). Query suggestions that are titles of Wikipedia articles and words included in WordNet were extracted from the **SUGGESTION** data, and were separated into ambiguous and unambiguous queries by using Sanderson's method (2008).⁶

Figure 4 shows the average *Suggestion CTR* for ambiguous and unambiguous query suggestions. Note that we are now discussing per-suggestion CTRs rather than those of entire query suggestion lists. First, it can be observed that the average *Suggestion CTR*s of query suggestions that match Wikipedia or WordNet are much lower than the total average *Suggestion CTR*. This suggests that Wikipedia and WordNet entries are often not useful as query suggestions. Second, it can be observed that the average *Suggestion CTR* for unambiguous suggestions is significantly higher than that for ambiguous ones, both for Wikipedia and for WordNet.⁷ Thus, *users often use unambiguous query suggestions*. This is also intuitive, as a query suggestion with multiple senses may not look promising to the user.

Next, we also classified the *original queries* into ambiguous and unambiguous ones using Sanderson's method, to further drill down the results. Figure 5 shows the average

⁶ Sanderson's approach can be taken even in non-English languages: the Wikipedia-based method can be applied where a sufficient number of Wikipedia articles are available, and WordNet-based method can work by using alternatives to WordNet in a target language, e.g. Japanese WordNet (Isahara et al. 2008)

⁷ Welch's *t* test was conducted to test the significant difference: $t(33061859) = 30.4$, $p < 0.001$ and $t(11360559) = 30.0$, $p < 0.001$, respectively.

Fig. 5 Suggestion CTRs for ambiguous and unambiguous queries, and ambiguous and unambiguous suggestions. “X–Y” along the horizontal axis means average suggestion CTR of Y query suggestions when X queries are input

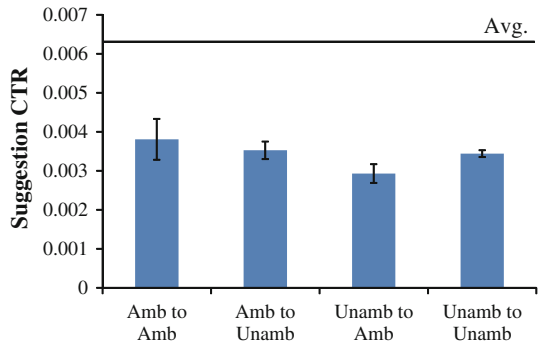


Table 10 Types of query-suggestion pairs and their definition

Type	Definition
Specialization (S)	$X \rightarrow X \cup Y$
Generalization (G)	$X \cup Y \rightarrow X$ (or Y)
Parallel movement (P)	$X \cup Y \rightarrow X \cup Z$ where $ X \geq \max(X \cup Y , X \cup Z)/2$
Weak parallel movement (W)	$X \cup Y \rightarrow X \cup Z$ where $ X < \max(X \cup Y , X \cup Z)/2$
Error correction (C)	$X \rightarrow X'$ where $Lev(X, X') \leq \theta$
New (N)	other than those above

Suggestion CTR for four cases: for example, “Amb to Unamb” means that the original query was classified as ambiguous, while the query suggestion was unambiguous. It can be observed that the average Suggestion CTR for “Unamb to Amb” is significantly lower than the other cases.⁸ That is, it is highly unlikely for a user who entered an unambiguous query to click on an ambiguous query suggestion.

5.4 Query suggestions for generalization and error correction

The previous section examined whether query suggestions had multiple *senses*. In this section, we examine another hypothesis about the properties of query suggestions, namely, that the users seek particular *types* of query reformulations, such as specialization and generalization. We automatically classified query-suggestion pairs into six query reformulation types based on a lexical categorization approach. In this study, we consider the reformulation types shown in Table 10, based on previous work by Boldi et al. (2009). Let X , Y , and Z denote nonempty sets of query terms, and $|X|$ denote the number of query terms in X . We define *specialization* (S) as a transition from a query represented by X to that represented by $X \cup Y$; *generalization* (G) as a transition from $X \cup Y$ to X (or Y); *parallel movement* (P) and *weak parallel movement* (W) as a transition from $X \cup Y$ to $X \cup Z$ (if $|X| \geq \max(|X \cup Y|, |X \cup Z|)/2$, then parallel movement; otherwise weak parallel

⁸ Welch’s t test with Bonferroni correction was conducted to test the significant difference. There are significant differences between “Unamb to Amb” and the others—“Amb to Amb”, “Amb to Unamb”, and “Unamb to Unamb”: $t(49453) = 3.92, p < 0.001, t(249,229) = 4.66, p < 0.001$, and $t(163324) = 5.14, p < 0.001$, respectively.

Table 11 The fraction of query-suggestion types in the **SUGGESTION** and **SESSION** data as well as those reported by Boldi et al

Type	% in SUGGESTION	% in SESSION	% in Boldi et al. 's
Specialization (S)	3.6	28.0	30.1
Generalization (G)	0.7	7.2	9.5
Parallel movement (P)	46.0	23.5	55.5
Weak parallel movement (W)	48.9	24.1	
Error correction (C)	0.8	17.1	5.0

movement); and *error correction* (C) as a reformulation where the Levenshtein distance (Lev) between a query and reformulated query is $< \theta$ (2 in this study). All other reformulations are classified as *new* (N). For example, given “microsoft windows 7” as a query, “microsoft windows 7 update” is a specialization, “microsoft windows” is a generalization, “microsoft windows 8” is a parallel movement, and “microsoft office” is a weak parallel movement. Using the rules shown in Table 10, we could classify approximately 80 % of query-suggestion pairs into five types other than *new*. Note that, in contrast to the original definitions of Boldi et al., we classify query reformulation types solely based on how the query and the suggestion overlap with each other.

Table 11 shows the fraction of query reformulation types based on the **SUGGESTION** data, and that based on the **SESSION** data: note that the latter data set contains not only query suggestions but also manually reformulated queries. In addition, we also show the classification results by Boldi et al. for comparison. It can be observed that, while the distribution of our **SESSION** data is similar to that of Boldi et al. in that there are about 30 % of specializations and 50 % of (weak) parallel movements, the distribution of our **SUGGESTION** data is quite different: specializations are rare and about 95 % are (weak) parallel movements. Note that this distribution for **SUGGESTION** data is impression-based: below, we shall discuss the distribution over reformulation types based on CTR.

Figure 6 shows the average Suggestion CTR for each of the six reformulation types. It can be observed that generalizations and error corrections are most likely to be used, and the specialization and parallel movement types had higher average Suggestion CTRs than the total average. Whereas, the other types, i.e. weak parallel movement and *new*, had lower average Suggestion CTRs than the total average.

By comparing Fig. 6 with the aforementioned distribution for the **SESSION** data shown in Table 11, we can observe that the popular query reformulation types are different between query suggestions and manual query reformulations. First, while generalizations and error corrections are “popular” query suggestions (in the sense that they are often clicked), these two types are actually the least frequent in the **SESSION** data. Second, while (weak) parallel movements and *new* query reformulations are frequent in the **SESSION** data, these reformulation types are not popular as query suggestions. Third, query suggestions for generalization are used more frequently than ones for specialization, whereas query reformulations for specialization are more frequent than ones for generalization in the **SESSION** data. One possible explanation for this discrepancy would be that the use of query suggestion and manual query reformulations are two very different modes of information exploration. Another would be that the current quality of query suggestions limits the users’ information seeking behavior. Note that, from the search engine’s point of view, generating generalizations and error corrections tend to be easier than generating

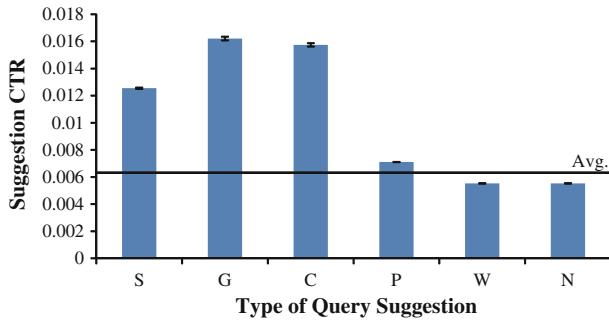


Fig. 6 Type of query suggestion versus Suggestion CTR

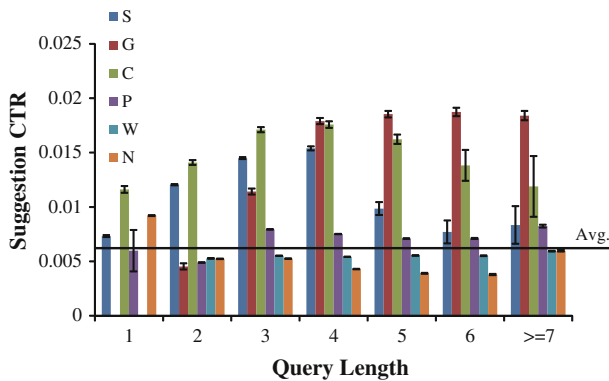


Fig. 7 Type of query suggestion and length of query versus Suggestion CTR

query suggestions of other types. Possibly, it might be the case that the users are not satisfied with the quality of the latter, and are forced to choose generalizations and error corrections. The overall conclusion of this section is that *query suggestions are likely to be used for generalization and error correction* rather than specialization, (weak) parallel movement, and new.

To examine the distribution of query suggestions over reformulations types more closely, Fig. 7 shows the average Suggestion CTR by the original query length for each reformulation type. When single-term queries are input, query suggestions of error correction and new types are more likely to be used. Query suggestions of the new type for single-term queries include several cases: acronym expansion (e.g. “NATO” to “North Atlantic Treaty Organization”), whitespace insert (e.g. “crosslingual” to “cross lingual”) and related queries similar to parallel movement (e.g. “Windows” to “Mac OS”). When an input query consists of multiple terms less than 5 terms, query suggestions are more frequently used for specialization and error correction as the query contains more terms. Given a query containing more terms, the user is more likely to use query suggestions for generalization.

5.5 Query suggestion after several URL clicks

We now discuss in what kind of search contexts query suggestions are likely to be used. Table 12 shows the transition ratio for each pair of actions based on the ACTION data.

Table 12 Transition ratio matrix

	To					
	Result	Page	Ads	QS	Other	End
From						
Result	0.18	1.52	0.44	1.29	0.49	2.80
Page	1.31	18.2	0.55	1.25	0.40	0.75
Ads	0.87	0.54	1.04	1.75	0.64	2.15
QS	0.12	0.42	0.19	0.02	0.21	3.04
Other	0.40	0.38	0.34	0.86	0.86	2.64
Start	2.86	0.61	2.66	1.75	2.58	–

Bold values indicate values >1

As was defined in Sect. 3, transition ratio from x to y reflects how likely y is followed by x . For example, the table shows that $P(\mathbf{Page}|\mathbf{Result})/P(\mathbf{Page}) = 1.52$, meaning that the action **Page** is more likely to occur after the action **Result** than to occur unconditionally. Values >1 are highlighted in bold. Note that the transition ratio $P(\mathbf{End}|\mathbf{QS})/P(\mathbf{End})$ is very high because a click on a query suggestion usually creates a new action pattern (for a new query).

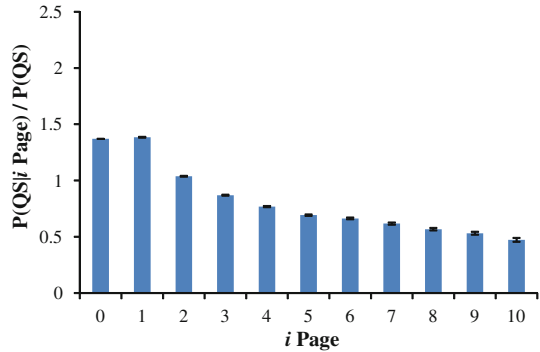
Before focusing on the action **QS**, we discuss the other actions in the transition ratio matrix. It can be observed that the **Page** to **Page** transition has the highest transition ratio (18.2). It indicates that the user tends to move from one result page to another, for example, by clicking on the “next page” button repeatedly, without taking any other actions in between. The **Start** row also shows that the user is unlikely to start an action pattern with **Page** (0.61): he is more likely to start with clicking on a URL (**Result**) or an advertisement (**Ads**). Whereas, the **Result** row shows that the user is unlikely to click on two URLs one after the other (0.18), and is likely to move to another result page (**Page**), use query suggestion (**QS**), or abandon search (**End**) after **Result**.

As for the action **QS**, it can be observed from the column **QS** in Table 12 that **QS** often follows **Start** (1.75) and **Ads** (1.75). That is, query suggestion is often used immediately after the first search result page is presented to the user or immediately after a click on an advertisement. The reason why query suggestion is used immediately after the search result page is presented may be that the user is disappointed in the search result quality or that the user finds a promising query suggestion in the left panel and clicks on it regardless of the current result quality.

The analysis so far discussed transitions from one action to another, but not those from a sequence of actions to a new action. In order to investigate what action sequences proceed query suggestion, we extend the notion of transition ratio to handle transitions to **QS** from an action sequence containing exactly i **Page** actions (and possibly some other actions) and from an action sequence containing exactly j **Result** actions (and possibly some other actions). We denote these transition ratios by $P(\mathbf{QS}|i \mathbf{Page})/P(\mathbf{QS})$ and $P(\mathbf{QS}|j \mathbf{Result})/P(\mathbf{QS})$, respectively.

Figure 8 shows the transition ratio for **QS** conditioned by the action sequence containing i **Page** actions. For example, the value at $i = 0$ indicates the transition ratio for **QS** at the first search result page, and the value at $i = 1$ indicates that at the second search result page (i.e. after 1 **Page** action). The trend is quite intuitive, in that the more pages the user sees, the less likely he is to turn to query suggestion. When the user is examining the

Fig. 8 Ratio of $P(QS|i \text{ Page})$ to $P(QS)$



first or the second search result page, the transition ratio to **QS** is high. However, after the third result page, the probability of turning to query suggestion gradually decreases. Since the user has already examined several pages, he may be reluctant to turn to query suggestion at this point and to start examining a new ranked list.

Figure 9 shows the transition ratio for **QS** conditioned by the action sequence containing j **Result** actions (i.e. URL clicks). Interestingly, there appear to be at least two peaks in this figure, one around $j = 2$ and another around $j = 8$. Figure 10 breaks down this phenomenon by visualizing the transition probability $P(QS|N \text{ Page}, M \text{ Result})$, which represents the probability of the action **QS** after an action sequence $* \rightarrow p_1 \rightarrow p_2 \rightarrow * \rightarrow \dots \rightarrow * \rightarrow p_N \rightarrow \star \rightarrow r_1 \rightarrow r_2 \rightarrow \star \rightarrow \dots \rightarrow \star \rightarrow r_M \rightarrow \star$, where p_i is the action **Page**, r_j is the action **Result**, $*$ represents any actions excluding **Page** or no action, and \star represents any actions excluding **Page** and **Result**, or no action. For example, $P(QS|1 \text{ Page}, 2 \text{ Result})$ is the probability of transition to **QS**, after one **Page** action (i.e. goes to the second search result page) and two **Result** actions (i.e. URL clicks). Note that the “1 Page” and “2 Page” curves are incomplete due to data sparsity. It can be observed that the bimodal phenomenon shown in Fig. 9 arises from the “0 Page” curve (i.e. after 0 **Page** action) in Fig. 10. That is, the bimodal phenomenon represents the user behavior for the very first search result page: *the user tends to click on either two or eight URLs in the first result page and then use query suggestion.*

The above bimodal phenomenon may be a search engine dependent result. For Bing, the default number of URLs per page is 10, and query suggestions are placed on the top-left side of a result page as well as below a list of URLs, as shown in Fig. 1 as of May, 2010.

Fig. 9 Ratio of $P(QS|j \text{ Result})$ to $P(QS)$

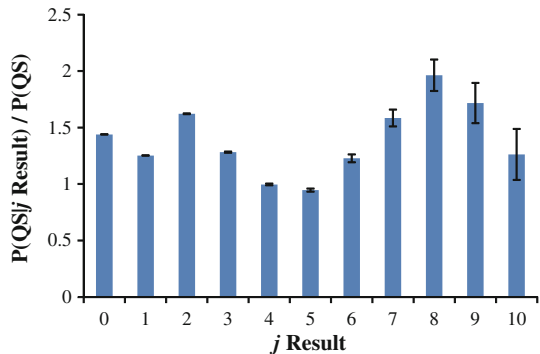
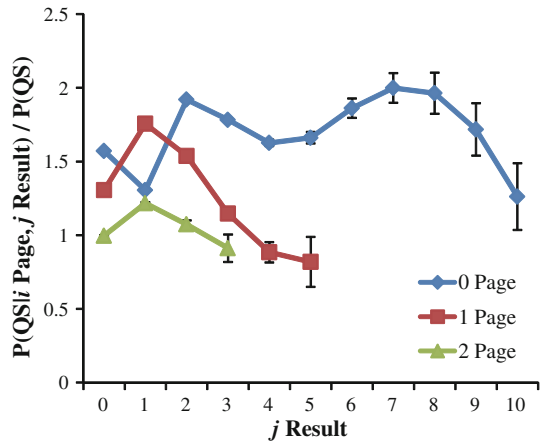


Fig. 10 Ratio of $P(\text{QSLi Page}, j \text{ Result})$ to $P(\text{QS})$. Note that the ratios for $i = 1$ and $j \geq 6$ and $i = 2$ and $j \geq 3$ are not shown due to data sparsity



Thus, the two peaks may represent a mixture distribution of two classes of users: those who use query suggestions on the top-left side after examining the first few URLs, and those who use query suggestions after examining most of the URLs in the result page. Other major search engines such as Google and Yahoo! provide query suggestions at different places, so we might obtain different results from these search engines.

Another interesting, though inconclusive, observation from Fig. 10 is the behaviour difference between the first and the second search result page ($i = 0$ and $i = 1$). After one URL click ($j = 1$) on the first search result page (see the “0 Page” curve), the user is unlikely to use query suggestions; whereas after one URL click on the second search result page (see the “1 Page” curve), the user is most likely to use query suggestions.

6 Discussions and implications

We believe that the findings reported in this paper are useful for improving the Web search user experience. Below, we discuss some important problems that we believe should be tackled.

6.1 Processing queries

Our first finding based on the **QUERY** data is that query suggestion is often used after a rare query. This has a useful implication for research in query suggestion, as query suggestion algorithms exploit user feedback (i.e. clicks) but rare queries may lack enough feedback data despite the large amount of traffics. Specifically, we argue that query suggestion research should focus on handling sparse user feedback data and on mining resources other than clicks.

Song and He (2010) have already tackled the problem of generating query suggestions for rare queries. They used the top search results for rare queries as *pseudo-relevant* documents together with clickthrough data. Dang and Croft (2010) explored the use of anchor text data instead of a query log for query suggestion. Hence their method may be advantageous for rare queries. We should see more work on query suggestion along these directions.

Our second finding based on the **QUERY** data is that query suggestion is often used after a single-term query. There are at least two possible situations where the user may

input a single-term query that later requires query suggestion: (1) the user's information need is still vague, and (2) the user's information need is relatively clear, but he either cannot or does not express it precisely. Research in exploratory search (Marchionini 2006; White et al. 2006, 2007) should help these problems to some extent. However, a simple suggestion to current Web search engines would be to provide a richer query suggestion experience for single-term queries compared to that for longer queries. For example, in response to a single-term query, a search engine could provide a larger and more diversified list of query suggestions to the user. The layout of the first search result page (i.e. where to show the suggestion list) could also be changed depending on whether the query is a single-term or not. It is possible that such a change will boost the Suggestion CTR for single-term queries even more. Moreover, the frequency of single-term queries accounts for a substantial percentage of the total query frequency, i.e. 22.69 % according to the **QUERY** data. Therefore, the improvement of query suggestions for single-term queries should be important for Web search engines.

6.2 Generating query suggestions

Our first finding based on the **SUGGESTION** data is that ambiguous (i.e. multi-sense) query suggestions are less likely to be used than unambiguous query suggestions. Therefore, Web search engines should provide query suggestions to help the user disambiguate his query. This is of great importance because ambiguous queries constitute a significant part of Web search queries. An analysis on a Web query log showed that about 16 % of Web queries are estimated to be ambiguous (Song et al. 2009). Diversifying Web search results is a sensible approach to tackle ambiguous queries in the absence of any knowledge of the user's context or preferences (Agrawal et al. 2009; Santos et al. 2010), while another is to present unambiguous and diversified query suggestions (Ma et al. 2010; Song et al. 2011).

Our second finding based on the **SUGGESTION** data is that query suggestion is often used for generalization and error correction. This is particularly interesting in that it is different from the situation with manual query reformulation. As was mentioned in Sect. 5.4, the difference may suggest that query suggestion and manual query formulation are two very different query exploration and formulation processes, or that the quality of the suggestions for specialization and parallel movement is unsatisfactory. One possibility to change this situation would be to improve the presentation interface for specialization and parallel movement (Kato et al. 2012). Alternatively, to improve the quality of these types of query suggestions, contextual information such as the user's search history and locations may be effectively utilized (Cao et al. 2008; He et al. 2009). Also, the present study suggests that it may be useful to present query suggestions of different reformulation types based on the original query length: According to Fig. 7, short queries often require specialization, while long queries often require generalization.

6.3 Interacting with the user

Our main finding based on the **ACTION** data is that query suggestion is often used after the user has clicked on several URLs in the first search result page. In addition, about 36 % of query suggestion usages happen just after a URL click according to the **ACTION** data. These statistics suggest that the choice of query suggestions depends on what kind of actions the searcher has taken so far. Dynamic generation of query suggestions based on previous user actions is probably an important research direction.

Our results also show that as the users dig deeper into the search result (by examining many result pages), they naturally abandon query suggestion. This suggests that showing a query suggestion list of fixed size in every search result page may not be a good idea: in later pages, the user may benefit more from fewer query suggestions and more URLs.

7 Conclusions

In this paper, we investigated when and how query suggestions are used by Web search users. We analyzed three kinds of data sets obtained from a commercial search engine, and obtained five main findings. According to our analysis, query suggestions are often used (1) when the original query is a rare query, (2) when the original query is a single-term query, (3) when query suggestions are unambiguous, (4) when query suggestions are generalizations or error corrections of the original query, and (5) after the user has clicked on several URLs in the first search result page. These results suggest, for example, that it is important for researchers to tackle the problems of providing good query suggestions for rare or single-term queries, and of dynamic generation of query suggestions based on previous user actions.

Although we investigated a large amount of data and clarified the usage of query suggestion, there are some limitations in our analysis. First, while we tried to observe general trends for search engines, not all of our results may generalize to search engines other than Bing (e.g. see the discussion of the bimodal phenomenon). Second, we have not assessed the quality of query suggestions, i.e. whether suggestions are relevant to the original query, and whether suggestions really satisfied the user's information need. The relevance of query suggestion can be assessed through tracking users' actions after they have used query suggestion. Third, our analysis was based on simple automatic classification of queries and suggestions. Although such classification methods can have the advantage of greater reproducibility, it deserves further exploration to manually classify queries and suggestions in terms of their topic and intent (e.g. navigational and informational) and to examine their effects on query suggestion usage. Finally, the effect of query suggestion presentation order was ignored in our analysis. In the future, we plan to investigate the usage of query suggestion with data sets including user information (e.g. user identifiers), to propose a query reformulation taxonomy specifically designed for query suggestion classification, and to improve query suggestion functionality based on our insights.

Acknowledgments This work was supported in part by the following projects: Grants-in-Aid for Scientific Research (Nos. 24240013 and 22-4687) from MEXT of Japan.

References

- Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *Proceedings of the WSDM 2009*, pp. 5–14.
- Amin, A., Hildebrand, M., Ossenbruggen, J., Evers, V., & Hardman, L. (2009). Organizing suggestions in autocompletion interfaces. In *Proceedings of the ECIR 2009*, pp. 521–529.
- Anagnostopoulos, A., Becchetti, L., Castillo, C., & Gionis, A. (2010). An optimization framework for query recommendation. In *Proceedings of the WSDM 2010*, pp. 161–170.
- Anick, P. (2003). Using terminological feedback for web search refinement: a log-based study. In *Proceedings of the SIGIR 2003*, pp. 88–95.

- Baeza-Yates, R., Hurtado, C., & Mendoza, M. (2004). Query recommendation using query logs in search engines. In *Workshops in current trends in database technology (EDBT 2004)*, pp. 588–596.
- Beeferman, D., & Berger, A. (2000). Agglomerative clustering of a search engine query log. In *Proceedings of the KDD 2000*, pp. 407–416.
- Boldi, P., Bonchi, F., Castillo, C., Donato, D., Gionis, A., & Vigna, S. (2008). The query-flow graph: model and applications. In *Proceedings of the CIKM 2008*, pp. 609–618.
- Boldi, P., Bonchi, F., Castillo, C., & Vigna, S. (2009). From Dango to Japanese cakes: Query reformulation models and patterns. In *Proceedings of the WI 2009*, pp. 183–190.
- Broder, A. (2002). A taxonomy of web search. *ACM SIGIR Forum* 36(2), 3–10.
- Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., et al. (2008). Context-aware query suggestion by mining click-through and session data. In *Proceedings of the KDD 2008*, pp. 875–883.
- Cui, H., Wen, J. R., Nie, J. Y., & Ma, W. Y. (2003). Query expansion by mining user logs. *IEEE Transactions on Knowledge and Data Engineering*, 15(4), 829–839.
- Dang, V., & Croft, B. W. (2010). Query reformulation using anchor text. In *Proceedings of the WSDM 2010*, pp. 41–50.
- Downey, D., Dumais, S., Liebling, D., & Horvitz, E. (2008). Understanding the relationship between searchers' queries and information goals. In *Proceedings of the CIKM 2008*, pp. 449–458.
- He, Q., Jiang, D., Liao, Z., Hoi, S., Chang, K., Lim, E. P., et al. (2009). Web query recommendation via sequential query prediction. In *Proceedings of the ICDE 2009*, pp. 1443–1454.
- Huang, J., & Efthimiadis, E. N. (2009). Analyzing and evaluating query reformulation strategies in web search logs. In *Proceedings of the CIKM 2009*, pp. 77–86.
- Isahara, H., Bond, F., Uchimoto, K., Utiyama, M., & Kanzaki, K. (2008). Development of the Japanese wordnet. In *Proceedings of the LREC 2008*, pp. 2420–2423.
- Jansen, B. J., Booth, D. L., & Spink, A. (2008). Determining the informational, navigational, and transactional intent of web queries. *Information Processing & Management*, 44(3), 1251–1266.
- Jansen, B. J., Booth, D. L., & Spink, A. (2009). Patterns of query reformulation during Web searching. *Journal of the American Society for Information Science and Technology*, 60(7), 1358–1371.
- Jones, R., Rey, B., Madani, O., & Greiner, W. (2006). Generating query substitutions. In *Proceedings of the WWW 2006*, pp. 387–396.
- Kamvar, M., & Baluja, S. (2008). Query suggestions for mobile search: Understanding usage patterns. In *Proceedings of the CHI 2008*, pp. 1013–1016.
- Kato, M. P., Sakai, T., & Tanaka, K. (2012). Structured query suggestion for specialization and parallel movement: Effect on search behaviors. In *WWW 2012*, pp. 389–398.
- Kelly, D., Gyllstrom, K., & Bailey, E. W. (2009). A comparison of query and term suggestion features for interactive searching. In *Proceedings of the SIGIR 2009*, pp. 371–378.
- Kelly, D., Cushing, A., Dostert, M., Niu, X., & Gyllstrom, K. (2010). Effects of popularity and quality on the usage of query suggestions during information search. In *Proceedings of the CHI 2010*, pp. 45–54.
- Kraft, R., & Zien, J. (2004). Mining anchor text for query refinement. In *Proceedings of the WWW 2004*, pp. 666–674.
- Lau, T., & Horvitz, E. (1999). Patterns of search: Analyzing and modeling web query dynamics. In *Proceedings of the seventh international conference user modelling*, pp. 119–128.
- Ma, H., Yang, H., King, I., & Lyu, M. R. (2008). Learning latent semantic relations from clickthrough data for query suggestion. In *Proceedings of the CIKM 2008*, pp. 709–718.
- Ma, H., Lyu, M. R., & King, I. (2010). Diversifying query suggestion results. In *Proceedings of the AAAI 2010*, pp. 1399–1404.
- Marchionini, G. (2006). Exploratory search: From finding to understanding. *Communications of the ACM*, 49(4), 46.
- Mei, Q., Zhou, D., & Church, K. (2008). Query suggestion using hitting time. In *Proceedings of the CIKM 2008*, pp. 469–478.
- Phan, N., Bailey, P., & Wilkinson, R. (2007). Understanding the relationship of information need specificity to search query length. In *Proceedings of the SIGIR 2007*, pp. 709–710.
- Rieh, S. Y., Xie, H. I. (2006). Analysis of multiple query reformulations on the web: The interactive information retrieval context. *Information Processing & Management*, 42(3), 751–768.
- Sanderson, M. (2008). Ambiguous queries: Test collections need more sense. In *Proceedings of the SIGIR 2008*, pp. 499–506.
- Santos, R. L. T., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for web search result diversification. In *Proceedings of the WWW 2010*, pp. 881–890.
- Silvestri, F. (2010). Mining query logs: Turning search usage data into knowledge. *Foundations and Trends in Information Retrieval*, 4(1–2), 1–174.

- Song, R., Luo, Z., Nie, J. Y., Yu, Y. & Hon, H. W. (2009). Identification of ambiguous queries in web search. *Information Processing & Management*, 45(2), 216–229.
- Song, R., Dou, Z., Hon, H. W., & Yu, Y. (2010). Learning query ambiguity models by using search logs. *Journal of Computer Science and Technology*, 25(4), 728–738.
- Song, Y., & He, L. (2010). Optimal rare query suggestion with implicit user feedback. In *Proceedings of the WWW 2010*, pp. 901–910.
- Song, Y., Zhou, D., & He, L. (2011). Post-ranking query suggestion by diversifying search results. In *Proceedings of the SIGIR 2011*, pp. 815–824.
- Song, Y., Zhou, D., & He, L. (2012). Query suggestion by constructing term-transition graphs. In *Proceedings of the WSDM 2012*, pp. 353–362.
- Spink, A., Jansen, B. J., Wolfram, D., & Saracevic, T. (2002). From e-sex to e-commerce: Web search changes. *Computer*, 35(3), 107–109.
- Teevan, J., Adar, E., Jones, R., & Potts, M. A. S. (2007). Information re-retrieval: Repeat queries in Yahoo's logs. In *Proceedings of the SIGIR 2007*, pp. 151–158.
- Wang, X., & Zhai, C. X. (2008). Mining term association patterns from search logs for effective query reformulation. In *Proceedings of the CIKM 2008*, pp. 479–488.
- White, R. W., & Marchionini, G. (2007). Examining the effectiveness of real-time query expansion. *Information Processing & Management*, 43(3), 685–704.
- White, R. W., Kules, B., Drucker, S. M., & Schraefel, M. C. (2006). Supporting exploratory search. *Communications of the ACM*, 49(4), 36–39.
- White, R. W., Bilenko, M., & Cucerzan, S. (2007). Studying the use of popular destinations to enhance web search interaction. In *Proceedings of the SIGIR 2007*, pp. 159–166.