Examining the Effectiveness of Real-Time Query Expansion

Ryen W. White[1,2]
Microsoft Research
One Microsoft Way
Redmond, WA USA 98052
ryenw@microsoft.com

Gary Marchionini
School of Information and Library Science
University of North Carolina
Chapel Hill, NC USA 27599
march@ils.unc.edu

Abstract

Interactive query expansion (IQE) (c.f. Efthimiadis, 1996) is a potentially useful technique to help searchers formulate improved query statements, and ultimately retrieve better search results. However, IQE is seldom used in operational settings. Two possible explanations for this are that IQE is generally not integrated into searchers' established information-seeking behaviors (e.g., examining lists of documents), and it may not be offered at a time in the search when it is needed most (i.e., during the initial query formulation). These challenges can be addressed by coupling IQE more closely with familiar search activities, rather than as a separate functionality that searchers must learn. In this article we introduce and evaluate a variant of IQE known as Real-Time Query Expansion (RTQE). As a searcher enters their query in a text box at the interface RTQE provides a list of suggested additional query terms, in effect offering query expansion options *while the query is formulated.* To investigate how the technique is used – and when it may be useful – we conducted a user study comparing three search interfaces: a baseline interface with no query expansion support; an interface that provides expansion options during query entry, and a third interface that provides options after queries have been submitted to a search system. The results show that offering RTQE leads to better quality initial queries, more engagement in the search, and an increase in the uptake of query expansion. However, the results also imply that care must be taken when implementing RTQE at the interface. Our findings have broad implications for how IQE should be offered, and form part of our research on the development of techniques to support the increased use of query expansion.

Keywords

Real-Time Query Expansion, query completion, query quality

[1] Corresponding author.

[2] This research was conducted while the first author was employed at the University of Maryland, College Park, MD 20742, USA.

1. Introduction

The quality of queries submitted to Information Retrieval (IR) systems directly affects the quality of search results generated by these systems (Croft & Thompson, 1987). For this reason the issue of how to improve search queries has been of great interest in IR research. One approach that has proven effective is training searchers to pose better queries by using thesauri (e.g., Sihvinen & Vakkari, 2004), or learning systematic search strategies (e.g., Bates, 1997). Since people are generally more concerned with solving their information problems than learning how to search, much research has been devoted to building system support for improving query quality. Techniques such as Relevance Feedback (RF) (c.f. Salton & Buckley, 1990) have been proposed as a way in which IR systems can support the iterative development of a search query using examples of relevant information. One way RF can help is by suggesting additional *query expansion* terms for query modification (Efthimiadis, 1996). This modification can occur interactively with searcher participation i.e., interactive query expansion (IQE), or automatically without searcher involvement i.e., automatic query expansion (AQE).

User studies (Koenemann & Belkin, 1996; Beaulieu, 1997) have shown that although terms selected during AQE benefit from the presence of statistical information inaccessible to searchers, searchers would still like to be in control of query expansion decisions. Koenemann & Belkin (1996) investigated the use and effectiveness of different levels of RF and query expansion with three experimental systems, ranging from "opaque" (an AQE system where term selection is hidden from the searcher), through "transparent" (an AQE system where terms are visible but not selectable by searchers), to "penetrable" (an IQE system where terms are visible and selectable by searchers)[3]. Their findings show that increasing the level of searcher control over query expansion term selection improves search effectiveness. Beaulieu (1997) carried out an investigation of three interfaces to IR systems: one offered AQE, and two offered IQE. The systems were not investigated through laboratory investigation as in the Koenemann and Belkin study, but through operational investigation: the systems were used as an interface to a university library catalogue. Beaulieu's findings show that, although an improved interface can increase the level of use of IQE and the effectiveness of term selection, this did not surpass AQE. Although recent work by Anick (2003) has demonstrated that some progress is being made in this area, it remains a problem how to get searchers to consistently employ IQE in operational environments.

[3] The "penetrable" interface includes a version of IQE that is similar in some respects to RTQE.

Harman (1988) demonstrated the potential effectiveness of IQE in a study conducted using simulated query expansion decisions. Other studies of IQE effectiveness have involved human subjects, and shown that it can be a worthwhile technique, but searchers may make poor expansion term selections (Magennis & Van Rijsbergen, 1997; Ruthven, 2003). As Ruthven (2003) suggests, the failure to realize the potential of IQE shown by Harman could be related to how it is presented at the interface. It is therefore important to investigate alternative ways of offering query expansion interactively.

We have developed a user interface mechanism that uses Pseudo-Relevance Feedback (PRF) (Jinxi & Croft, 1996) (i.e., assumes that all highly-ranked results are relevant), and offers expansion terms at query time. These terms are presented as a list very shortly (less than two seconds depending on network latency) after the searcher finishes typing the first term of their query, and updates after each term is typed. Searchers may either select a term or ignore the suggestions, and complete their query. This approach integrates IQE directly into query formulation, giving help at a stage in the search when it can positively affect query quality, and possibly supporting the development of improved expansion strategies by searchers. We call this Real-Time Query Expansion (RTQE).

Although similar techniques have already been implemented (e.g., Google Suggest[4]), there exists to our knowledge no study of how effective such techniques are for real searchers. In this article we describe a user study in which we compare RTQE with a no expansion baseline and a system offering IQE after a search has been performed. The study assesses the efficacy of these systems on four dependent variables: task completion time, searcher satisfaction, quality of the results, and quality of the query. These variables give multiple perspectives on the effectiveness of RTQE, and give us insight into the effect of RTQE on key aspects of the search process. The overarching aim of the study is to determine the circumstances under which RTQE performs well and when it performs poorly.

The remainder of this article is structured as follows. Section 2 describes the study, including a description of the RTQE approach. Section 3 describes the findings, and Section 4 discusses them and their implications. We conclude in Section 5.

[4] http://www.google.com/webhp?complete=1

2. Study

A laboratory-based within-subject user study was conducted. In this section we describe the main aspects of the study, beginning with the experimental systems.

2.1 Systems

Three systems were developed for this research study: a baseline system that provided no explicit support for query expansion, a PRF system that assumed top-ranked results were relevant and presents query expansion options in real-time as a searcher enters their query, and a third comparator system that also uses PRF but presented query expansion options after a retrieval has been initiated by the searcher. We begin by describing the RTQE approach.

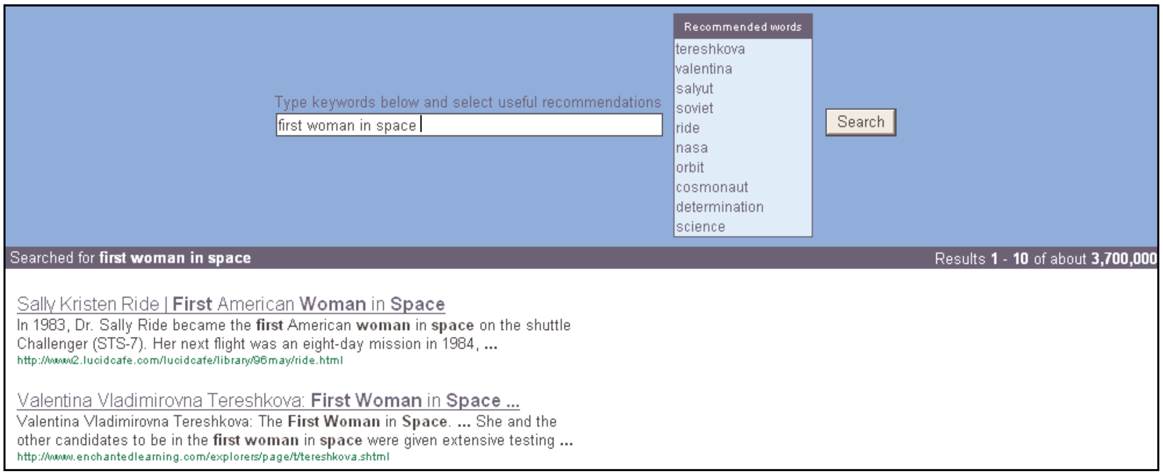
2.1.1 Real-Time Query Expansion

RTQE is embedded directly into the initial query entry screen and results display. It produces updated lists of potential query expansion terms as a searcher types words into the query entry box. When the spacebar is pressed between terms the component presents the current contents of the query box to Google,[5,6] and chooses the top ten query expansion terms from the surrogates (i.e., titles and query-relevant abstracts) of the top ten results. Ten documents was the maximum number of results that the Google API offered without having to perform multiple retrievals. Since these surrogates often contain query terms in one or more contexts within the document, they can be effective sources of terms for query expansion. The effectiveness of surrogate information for this purpose has already been demonstrated in previous work (Allan, 1995; Lam-Adesina & Jones, 2001). Terms are displayed in a "Recommended words" list situated between the query entry box and the search button. The number of feedback terms displayed is limited to ten to minimize cognitive load by keeping the list short enough to examine quickly. To append a term from the list of recommendations to the current query, the searcher can double-click the term in the list with the mouse pointer.[7] Figure 1 illustrates the component in action for one of this study's tasks. Also shown in the figure are the search results as they appear in all three systems used in this study. A maximum of ten results are displayed per page on the interfaces to these systems.

[5] Using the Google Java API (http://www.google.com/apis).

[6] We conducted experiments on the Web since this was the search environment that we felt subjects would be most familiar with, and allowed us to create search tasks on a broad range of topics.

[7] Since a searcher may want to add multiple suggested terms, adding a term does not lead to an immediate update in the "Recommended words" list. All updates occur only once the searcher has pressed the spacebar.



Type keywords below and select useful recommendations

first woman in space

Searched for **first woman in space**

Results **1** - **10** of about **3,700,000**

Sally Kristen Ride | **First** American **Woman** in **Space**

In 1983, Dr. Sally Ride became the **first** American **woman** in **space** on the shuttle Challenger (STS-7). Her next flight was an eight-day mission in 1984, **...**

http://www2.lucidcafe.com/lucidcafe/library/96may/ride.html

Valentina Vladimirovna Tereshkova: **First Woman** in **Space ...**

Valentina Vladimirovna Tereshkova: The **First Woman** in **Space**. **...** She and the other candidates to be in the **first woman** in **space** were given extensive testing **...**

http://www.enchantedlearning.com/explorers/page/t/tereshkova.shtml

Figure 1. Term suggestions in real-time at the interface. The list of "Recommended Words" updates after each query word is typed in the text box. In this example searcher has just pressed the spacebar.[8]

RTQE is in some respects similar to the "penetrable" RF interface used in the Koenemann & Belkin study described earlier. That interface provided an opportunity to manipulate the output of the RF component before the query was used in retrieving a new set of documents. That is, the execution of the query was interrupted and searchers were presented with a list of suggested terms prior to the continuation of the query evaluation. The "penetrable" interface was based on explicit RF, where searchers needed to mark documents to provide the relevance information used to generate terms, and searchers needed to explicitly indicate when they were ready to perform their search. RTQE sacrifices user control over these activities in favor of an interface mechanism that integrates term selection more naturally into the query formulation process.

Query expansion terms in RTQE are selected from the top ten titles and abstracts using a PRF approach based on simple term weighting techniques. As is standard in PRF all top-ranked search results (in our case meta-information about these results) are assumed to be relevant in some way. A list of all non-query terms in these information sources is created. Commonly occurring "stop words" are removed and each remaining term is scored based on the number of titles and abstracts it appears in, we call this a *surrogate frequency*. For example, a term occurring in all ten document titles and abstracts would receive a surrogate frequency of 20. Then, we count the frequency with which each term is juxtaposed on either side of a query term in document surrogates. This gives insight into how often a term co-occurs with a query term,

[8] First woman in space: Soviet cosmonaut Valentina Tereshkova.

and hence whether it would be a reasonable candidate for query expansion. Terms are ranked based on the product of this co-occurrence frequency and the surrogate frequency.

Although alternative approaches to PRF are possible (e.g., a search engine can use logs of all queries to extract related terms that often co-occur in previous queries), the approach we adopted was felt to be a simple and fast way of scoring terms, and is not dependent on having access to search engine logs or prior relevance judgments. Query syntax supported included the separation of terms with the space character, and the concatenation of terms to form multi-term phrases such as "Boston marathon." Since Google is used as the underlying search engine, terms (and phrases) in the resultant query are ANDed together when performing a new search. The average time for a new set of terms to be generated following the release of the spacebar was 1.8 seconds.[9] Whilst the average typing speed of some computing professionals is over 150 words per minute, for many computer users this value is around 20 words per minute for composition (Karat *et al.*, 1999), an average of one word every three seconds. Although query formulation may be more sporadic than other forms of composition this does suggest that the latency may not have a drastic effect on searcher performance. We now describe the three systems used in this study.

2.1.2 Baseline

Baseline does not offer explicit support to searchers in formulating their query statements. In response to a searcher-defined query the system presents the top ten titles, abstracts and URLs in a ranked list retrieved from the Web. The system is effectively a masked interface to the Google search engine. We chose to mask the engine in an effort to reduce potential bias caused by subjects' previous experiences; subjects were never told that they were interacting with Google.

2.1.3 RealTime

RealTime implements the approach described in Section 2.1.1. It presents searchers with a list of ten candidate query expansion terms that updates after each query term has been typed. Once the search button is pressed the system operates in the same way as *Baseline*. That is, it retrieves the top ten results from Google and presents their surrogates in a ranked list. The list of recommended expansion terms appears next to the query entry box on the results page (initially with the same terms as when the search button was pressed), and updates as searchers reformulate query statements in the same way as on the initial query entry screen.

[9] Approximately 98% of this average was related to the response time of the Google Java API (on average 1.7 seconds per query).

2.1.4 Retrospective

Retrospective uses the same query expansion weighting technique as described in Section 2.1.1, but does not offer real-time support as searchers type. Instead, this system provides the top ten expansion terms in a list next to the query box (in the same way as *RealTime*) *after retrieval has been performed*. This means that the searcher must wait for the results of a retrieval to see the query expansion options, potentially slowing them down. However, they get to see these options in the context of the search results, allowing them to potentially make more informed selections. For this reason we refer to this approach as "retrospective" expansion, since the searcher can look back over the result list before making query modification decisions.

The only difference between *RealTime* and *Retrospective* was *when* the query expansion support was offered. *Baseline* differed from *RealTime* and *Retrospective* because it did not offer any explicit support for query formulation.

2.2 Subjects

A total of 36 subjects were recruited between the campuses of the University of Maryland at College Park (UMD), and the University of North Carolina at Chapel Hill (UNC). To facilitate data collection and diversity, 18 subjects were recruited from each campus and were compensated financially for their participation. Recruitment took take place via flyers and postings to email lists. Table 1 shows subject characteristics at each of the sites.

Table 1. Subject demographic information from each site.

Site	**UMD**	**UNC**
Age (range and mean)	19-33 [26 years]	19-56 [28 years]
Gender	10 males, 8 females	13 males, 5 females
Search experience	5-10 [8.3 years]	4-25 [9.3 years]
Computer use frequency	Daily	Daily
Search frequency	Daily	Daily

Subjects were both undergraduate and graduate students from a range of nine different majors. Given that our subjects were drawn from the student population at both sites, we felt that it would be difficult to find novice and expert searchers; most of those who participated have were familiar with information technology and search systems. For this reason there was no division of the subject groups based on levels of search experience.

2.3 Tasks

Since the type of task may also influence the effectiveness of query expansion (Beaulieu, 1997), we made task type an independent variable in this study. We developed two known-item retrieval type tasks and two open-ended, exploratory type tasks for each condition that were rotated between systems and subjects. Figure 2 shows examples of the two task types.

Known-item task
You are doing some research for a term paper you are writing and need to find the name of the first woman to travel in space and her age at the time of her flight.

Exploratory task
You are about to depart on a short-tour along the west coast of Italy. The agenda includes a visit to the country's capital, Rome, during which you hope to find time to pursue your interest in modern art. However, you have recently been told that time in the city is limited and you want information that allows you to choose a gallery to visit.

Figure 2. Examples of known-item task and exploratory task.

The exploratory tasks were phrased in the form of simulated work task situations (Borlund, 2000), i.e., short search scenarios that were designed to reflect real-life search situations and allow subjects to develop personal assessments of relevance. The known-item search tasks required subjects search for particular pieces of information (e.g., an email address, a name, a date or time). The exploratory search tasks required subjects to gather information on a particular topic to allow them to perform some action (e.g., help a friend construct a letter of complaint, decide on an art gallery to visit). We began with tasks used in previous work (White, 2004), and for the known-item searches, carefully adapted them to current Web conditions to insure parallelism in the average number of clicks required to reach relevant information (on average 1-2 clicks per query). The exploratory search tasks required much more work by searchers and there were too many variants to try to insure similar optimal click patterns for those tasks. The two types of tasks had differing levels of *a priori determinability*, and therefore different levels of complexity (Byström & Järvelin, 1995; Bell & Ruthven, 2004). In the classification described by Byström and Järvelin the known-item task would represent a *normal decision task*, and the exploratory task represents a *genuine decision task*. The description of required task inputs (what information is necessary for searching), processes (how to find the required information), and outcomes (how to recognize the required information) in the task statements we provided to subjects were more uncertain in the exploratory tasks. Subjects were asked to write down answers and notes during each task. These were coded by the experimenters for later analysis.

2.4 Hypotheses

Four clusters of experimental hypotheses were devised that drove our investigation. The clusters are related to four different measures of search activity (time, satisfaction, quality of results, and quality of queries). Since we assume that different systems may be more or less useful for different task types, we pose separate hypotheses for the known-item tasks (*k*) and exploratory tasks (*e*).

2.4.1 Task Completion Time

There is a learning curve associated with new interface technology that users must overcome before they can become comfortable using it. The time to complete a search task on a system can give insight into the utility of the system and the nature of the search task. Since the tasks were held constant and rotated between subjects and systems, we used the time to complete the task as one way of determining the usefulness of the query expansion support offered by the systems. We devised the following two hypotheses.

$H_{t(k)}$**: For known-item searches, *Baseline* leads to faster task completion times than *RealTime*, which in turn is faster than *Retrospective*.** We posited that subject familiarity with traditional ranked-list style result interfaces such as *Baseline* would override any potential benefit that query expansion in *RealTime* or *Retrospective* could offer.

$H_{t(e)}$**: For exploratory searches, *RealTime* leads to faster task completion times than *Retrospective*, which in turn is faster than *Baseline*.** We posited that subjects attempting exploratory tasks would benefit from query formulation support, and that helping searchers select these terms may speed up their search. We felt that the earlier introduction of IQE in *RealTime* would lead to faster task completion than in *Retrospective*, where searchers must submit a query before they received assistance. We also felt that offering any query support at all would be better than none offered in the *Baseline* system.

2.4.2 Satisfaction

Satisfaction is a complex construct that is best assessed with several probes rather than a single value. We considered satisfaction as a group of four factors: usability, effectiveness, engagement, and enjoyment. The former two factors are common to many usability studies and were each assessed with six statements with accompanying 5-point Likert scales (e.g., Usability:

Learning to operate this system was easy for me; Effectiveness: *I find this system useful for finding information*) The latter two factors are associated with cognitive flow and were each assessed with four 7-point semantic differential scales (e.g., Engagement: *How you felt while using the retrieval system*: *absorbed intensely / not absorbed intensely*; Enjoyment: *Using the retrieval system: enjoyable / not enjoyable*).

The entire satisfaction assessment was kept to a single page but took substantial time to complete. Since there was a demographic questionnaire at the beginning of the experiment and a final questionnaire at the end of the experiment, we asked subjects to complete the system satisfaction questionnaire after they had finished using each system rather than each pair of tasks. Therefore, our satisfaction measures are generally not broken out by search task.[10] We posed the following hypothesis (that was tested for each of the four satisfaction factors).

***H_s*: Subjects are most satisfied with *RealTime*, then *Retrospective*, and least satisfied with *Baseline*.** This was derived from our belief that: subjects would prefer systems offering query support was over those that did not, and of those that offered such support, they would rather have it during the initial formulation of their query.

2.4.3. Result Quality

Result quality was assessed differently for the known-item and exploratory tasks. In known-item tasks, where assessment can be objective, it was assessed based on the final answers that subjects obtained, e.g., for one of the study's known-item tasks "The Green Bay Packers won Super Bowl I on January 15 1967." would have constituted a correct answer. In exploratory searches subjects were expected to take more notes and produce answers that could not be assessed objectively. We therefore judged result quality in exploratory tasks by evaluating the highest quality list of top-ten documents they obtained throughout each task.

Subjects attempted two known-item tasks on each system. For the first known item-task (an email lookup task in all cases) subject answers were scored as 0 (incorrect) or 1 (correct). For the known-item tasks attempted second each had two- part answers, and were scored as 0 (both parts incorrect), 1 (one part correct), or 2 (both parts correct). A total known-item answer correctness

[10] The only exceptions to this are subjects' system preferences for each task type, elicited in the final questionnaire.

score was therefore computed for each subject for each system. These objective scores were computed by the experimenters at each site based on the correct answers for each of the tasks.

Subjects attempted two exploratory tasks on each system. For some time IR research has focused on evaluating the quality of results retrieved by search systems (Sparck-Jones, 1981). There are two commonly used measures of result quality that have emerged from this research: *precision*, the ratio of relevant retrieved documents to the total number of documents retrieved, and *recall*, the ratio of relevant retrieved documents to the total number of (known) relevant documents. As many have noted, searchers on the Web tend to be more concerned with precision than recall, typically scanning only the top ten ranked results for appropriate documents (Spink *et al.*, 2002). For this reason we use *precision at 10 retrieved documents* as the objective measure of result quality for exploratory tasks. As an estimate of result quality we employed a panel of two judges who independently assessed the quality of every set of results. During the assessment process judges resubmitted the queries posed by subjects to the Google search engine and evaluated the quality of the top ten results obtained.[11] To do this we first judged the precision at 10 for all of each subject's queries, and then selected the highest precision value for all of their queries for each task. We selected the best precision value for each task because subjects posed many queries and some of these queries were meant to focus on specific aspects of the task (some subjects were observed to use a building block search strategy, others posed very specific queries to contextualize or clarify details after finding the relevant answers). We then took the mean of the best precision values for the two exploratory tasks as the overall results quality score for each subject-task-system triple. We tested two hypotheses for result quality. We now describe these hypotheses and the rationale for selecting them.

***$H_{RQ(k)}$*: The quality of search results is the same in all three systems.** We posited that the query expansion support offered by *RealTime* and *Retrospective* would be of limited benefit to subjects for known-item tasks. Our feeling was that subjects would be able to extract the terms they needed from the task descriptions, and may not use the query expansion. If this suspicion was borne out, the search results from all three systems would be of similar quality.

[11] Our judges carried out these new retrieval runs immediately after our experiments were completed. This minimized the likelihood that the contents of the result lists would change due to new indexing, improving the reliability of the result quality assessments.

$H_{RQ(e)}$**: The quality of the search results is highest on *RealTime*, then *Retrospective*, and lowest on *Baseline*.** For exploratory tasks, where subjects may be more likely to require query support, we posited that giving the support during query formulation leads to better quality search results than giving support after query formulation, and in turn than providing no support at all. We felt that the inclusion of query support at an earlier stage in the search may allow searchers to build a better quality initial query, leading to better quality queries (and hence results) in later searches.

Given that the search engine is a constant factor in this study, the quality of the search results are inevitably linked to the quality of the query submitted to search system. Query quality and result quality gave us two probes into the effectiveness of RTQE. In the next subsection we describe how we established query quality.

2.4.4. Query Quality

Query quality is a complex construct that is dependent on many factors such as the searcher's knowledge about the need, search experience, system experience, and the mapping between the need and the information source. By assigning search tasks, an experiment both gains comparability across subjects and collapses some of the query creation variability at the cost of natural motivation and setting for searchers. Taylor (1968) described four levels of information need in natural settings (visceral, conscious, formalized, and compromised), and assigning search tasks removes the variability of searchers mapping across the first three levels and reduces it to the mapping between the formalized (assigned task statement) and the compromised (the query actually posed to a system) levels. Although there have been some efforts to have experts judge the quality of queries, we know of no direct ways of assessing query quality (e.g., Wildemuth & Moore (1995) had librarians assess queries posed by medical students and identified missed opportunities but no consistent relationships between queries and effectiveness).[12] As an estimate of query quality we employed a panel of two judges who independently assessed the quality of every query expressed for all subjects using a 5-point scale. The judges met with one of the experimenters and discussed ways to assign values. The basic agreement was to examine the task, conduct a search, and then identify the key concepts in the task to use as basis for judging the subject queries. The judges then coded queries for one task together to establish a common rating scheme. They then independently assessed the queries for each of the tasks.

[12] An alternative could be to use language models to predict query performance (Cronen-Townsend *et al.*, 2002).

Table 2 shows the rating scheme used by the judges for the known-item topic shown in Figure 1, where the goal was to find the name of the first woman in space, and her age at the time of flight. The structure of the scheme was the same for all tasks: query-quality score to be assigned to the query, the number of unique concepts applied in the query, and the relevant concepts that need to be mentioned for that score to be assigned. The relevant concepts varied according to the task.

Table 2. Query-quality rating scheme used in example known-item task.

Query-quality score	Number of applied concepts in query	Concepts
1	0	unrelated terms
2	2	woman + space
3	3	woman + space + first
4	4	woman + space + first + age (or year, date of birth) / valentina tereshkova (background information that a subject has about task topic) + age
5	5	woman + space + first + age (or year, date of birth) + flight

There was a high Pearson correlation between the sets of ratings obtained from the two judges for the known-item tasks (r = .60) and good Pearson correlation for the exploratory tasks (r = .43). Weighted Cohen Kappa values for inter-rater reliability were computed for each task type (M = .51 for known-item, M = .35 for exploratory). Although the usual threshold for reliability is .7, the 5-point range of scores somewhat mitigates this stringent threshold which is based on dichotomous scales and we argue that these reliabilities reflect adequate estimates of query quality. There were a total of 1174 queries. The mean query quality score for each subject-task-system triple was then used to compare query quality across the three systems and the two types of tasks. The hypotheses were:

***$H_{QQ(k)}$*: The quality of the query generated on *RealTime* exceeds that on *Retrospective*, which equals *Baseline*.** We felt that for known-item tasks the query generated by RTQE would exceed the queries generated after retrieval has been performed. This was derived from our belief that subjects would generally build their queries around the initial query submitted. Therefore, a poor quality initial query may lead to poor later queries. The ordering of the systems in the hypothesis was based on the predicted impact on the initial query. *RealTime* provides support during the generation of this initial query, *Retrospective* and *Baseline* do not. Since known-item searches are generally short, we posited that subjects would not use the IQE options in *Retrospective*, meaning that it would be no better for improving query quality than *Baseline*.

***$H_{QQ(e)}$*: The quality of the query is highest on *RealTime*, then *Retrospective*, and lowest on *Baseline*.** The rationale for the ordering of systems in the hypothesis is much the same as that for the known-item search tasks. *RealTime* provides support during the generation of the initial query, *Retrospective* provides support after the query has been formulated (which may be beneficial beyond the first query iteration), and *Baseline* provides no support. We felt that the early introduction of IQE in *RealTime* would improve query quality. Also, since we felt that subjects were more likely to use the IQE options in *Retrospective* for exploratory tasks than known-item, we predicted that query quality would be higher in *Retrospective* than *Baseline*.

2.5 Experimental Design

The experiment used a within-subjects design across three systems with two kinds of search task. Subjects attempted two known-item and two exploratory tasks on each system. The order in which systems were used and search tasks attempted was carefully counterbalanced for system and task order according to a Latin square experimental design (6 combinations of the three experimental systems). A common experimental procedure was deployed between the two sites. The same questionnaires were used at each location and a common experimental script was used when introducing subjects to the experimental methodology to improve consistency. Questionnaires used Likert scales, semantic differentials, and open-ended questions to elicit subject opinions (Busha & Harter, 1980). System logging was used to record subject interaction.

2.6 Procedure

Subjects were run independently except for three pairs of subjects who were run concurrently at separate workstations at the UNC site. Each experimental session ran for up to two hours. The procedure we adhered to was as follows:

1. Upon arrival, subjects were given an overview of the study in written form that was read aloud to them by the experimenter. Subjects were also asked to read and sign a consent form.
2. Subjects then completed a short demographic questionnaire focusing on search experience and aspects of computer use.
3. For each of the three interface conditions:
    a. Subjects were given a short explanation of interface functionality lasting around 2 minutes.

   b. Subjects were asked to perform two known-item searches. They were limited to 5 minutes for each of these searches and could move on to the next search if they finished before the allotted time had expired.
   c. Subjects were then given two exploratory search tasks and were allotted up to 10 minutes for each task.
   d. Upon completion of the four searches on this interface, they were asked to complete a short questionnaire about their experience.
4. After attempting the 12 searches on the three interfaces subjects answered a final questionnaire that focused on comparing their experiences with the three interfaces.
5. Subjects were thanked and given $20 for their participation.

In the next section we present the findings of our study.

3. Findings

In this section we use the data derived from the experiment to test our hypotheses about the effectiveness of RTQE for known-item and exploratory searches. The four dependent variables are: task completion time; satisfaction with search system; quality of results; and quality of queries. Statistical analysis is conducted using parametric statistical testing at $\underline{p} < .05$ unless otherwise stated. M and SD denote the mean and standard deviation respectively. We present our findings per hypothesis.

3.1 Task Completion: Time / Result Quality

The attainment of high quality search results, and doing so in a short period of time, are related goals within the completion of a search task. For this reason we consider them together in the same section of our analysis. We analyze data related to hypotheses $H_{t(k)}$ and $H_{t(e)}$ on task completion time and $H_{RQ(k)}$ and $H_{RQ(e)}$ for results quality.

3.1.1 Task Completion Time

During the experiment all subject-system interaction events were automatically recorded in interaction log files with associated time stamp information. Task completion time was measured as the time in seconds from the first interaction event (i.e., the first keystroke of the first query) to the last subject interaction of any kind. Figure 4 shows the mean average task completion times for each system and task type.

The results suggest that subjects were fastest on *Baseline* for the known-item tasks, and fastest on *RealTime* for the exploratory tasks. A one-way repeated measure Analysis of Variance (ANOVA) to test for significant differences in task completion times between systems within each type task.[13] There were no significant differences in completion times between any systems for known-item tasks F(2, 142) = .530, ns, or for exploratory tasks F(2, 142) = .617, ns.

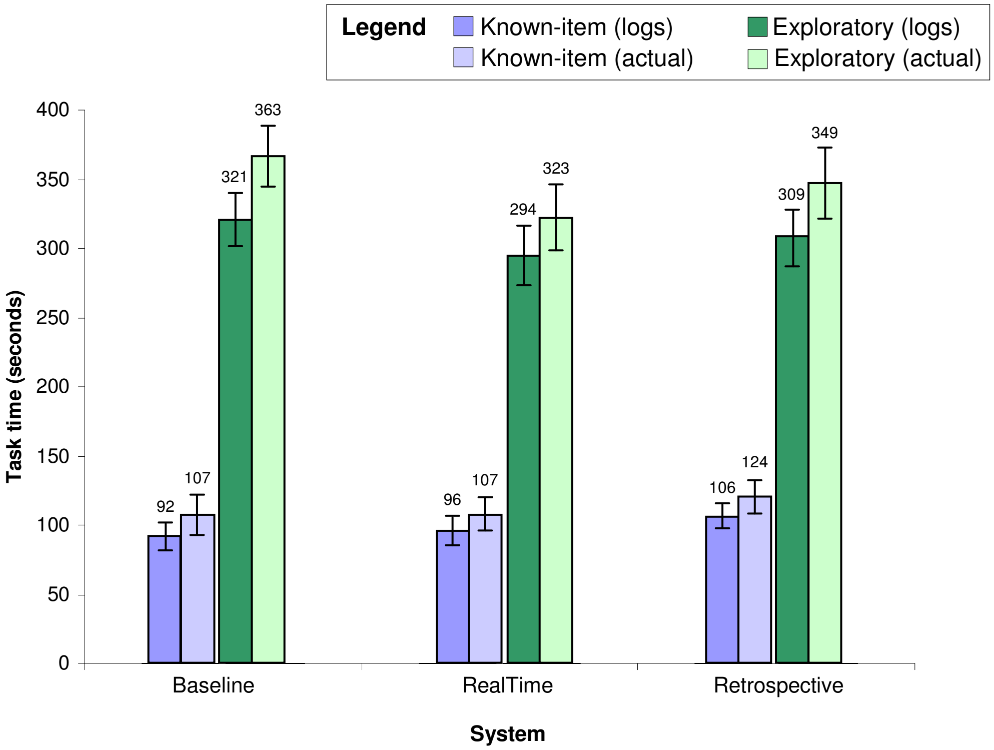


Figure 4. Mean average task completion times (± *SEM*).

In Figure 4 we also show the average "actual" task time recorded at one of the experimental sites (UMD). As well as the task time as it appears in the system logs, the experimenter at that site recorded using a stopwatch the full time taken to complete a search task (i.e., the time taken from when the subject began reading the task description until the subject informed the experimenter that they felt they were finished). This allows us to additionally analyze task completion time based on 18 subjects' perceptions of completeness rather than as extracted from the interaction logs. The additional time reported for the stopwatch-recorded times were generally due to time spent by subjects reading the task description. This analysis revealed that the time taken to reach

[13] We did not test for the effects of task type on task completion time since tasks had different allotted times (i.e., 300 seconds for known-item tasks, and 600 seconds for exploratory tasks).

task completion increases, and leads to significant differences between the systems for both task types F(2, 71) = 3.98, p = .023. Tukey post hoc tests revealed the difference between *RealTime* and *Retrospective* was statistically significant. Other differences were not significant. Thus, it is apparent that reading/thinking/planning before typing a query is affected by RTQE.

3.1.2 Result Quality

As described in Section 2.4.3 we analyzed the result quality for the known-item and exploratory search tasks in different ways. In this section we describe findings for each task type separately.

KNOWN-ITEM TASKS

The result quality scores for each subject on each system were averaged to give a final score for known-item tasks on each system. The maximum score that could be obtained on any system was 3 (i.e., both known-item tasks correct for every subject). The mean average score for *RealTime* (M = 2.75, SD = .65) was higher than *Baseline* (M = 2.69, SD = .58) and *Retrospective* (M = 2.67, SD = .72). However, a one-way repeated measures ANOVA found no effect of experimental system on result quality in known-item tasks F(2, 142) = .142, ns. Therefore, as expected, no system outperformed the others in helping subjects' obtain correct answers for known-item searches.

EXPLORATORY TASKS

The mean of the best precision at 10 values for every exploratory task was determined and one-way repeated measures ANOVA was computed for each exploratory task-system pair. The mean precision for the *RealTime* (M = 4.24, SD = 2.56) was higher than the mean for *Retrospective* (M = 4.14, SD = 2.57), and both were higher than the mean for *Baseline* (M = 4.04, SD = 2.70).[14] These differences were not statistically significant F(2, 142) = .421, ns.

3.1.3 Summary

In this section we presented findings related to task completion time and result quality. Task completion times from the interaction logs do not support hypotheses $H_{t(k)}$ and $H_{t(e)}$, but the relatively low times for *RealTime* are promising given the earlier concerns about time delay during query entry. The inclusion of data about the actual task completion times gathered at one

[14] Values for precision at 10 for initial query alone were different (M: *RealTime*: 4.40, *Baseline*: 4.17, *Retrospective*: 3.94), although not significantly so F(2,142) = .81, ns.

experimental site rather than the system-recorded task completion times showed that *RealTime* led to faster task completion than *Baseline*. As predicted, there was no difference in results quality for known-item tasks. The presence of any query expansion support was unable to influence the values of precision at 10 documents retrieved. Further analysis of retrieval effectiveness beyond the top ten documents may yield differences, but in a Web-based study, where searchers generally demonstrate an unwillingness to browse beyond the first result page, it is the quality of top-ranked results that may be truly important.

3.2 Satisfaction

In this section we analyze data related to hypotheses H_s for each of the four satisfaction factors. As described in Section 2, we measured this satisfaction with a variety of Likert scales and semantic differentials. Table 3 presents the mean average 5-point Likert scale responses to attitude statements about the effectiveness and usability of the systems, and mean average 7-point semantic differential responses about subjects' levels of engagement and enjoyment with each of the experimental systems. In all scales, values nearer to 1 reflect a higher level of agreement with the attitude statement, or a more positive differential response. The most positive responses for each measure are shown in bold.

Table 3. Satisfaction responses (Mean and standard deviation).

Measure		Baseline		RealTime		Retrospective	
		M	SD	M	SD	M	SD
5-point scale	Effectiveness	**2.42**	**1.11**	2.46	1.07	2.64	1.24
	Usability	**2.20**	**1.17**	2.35	1.20	2.32	1.24
7-point scale	Engagement	3.13	1.33	**2.99**	**1.38**	3.45	1.47
	Enjoyability	3.37	1.36	**3.31**	**1.51**	3.79	1.49

The next four subsections describe the findings for each of these measures in more detail, and provide the results of statistical analyses.

3.2.1 Effectiveness

Subject opinion on system effectiveness was measured using six Likert scales. The values for "Effectiveness" in Table 3 are the combined mean (and standard deviation) of responses for all of these six statements for each system. One-way repeated measures ANOVA yielded significant differences between systems in regard to overall feelings of effectiveness F(2, 430) = 3.48, p =

.032. Post hoc comparisons using a Tukey Test revealed that *Baseline* was perceived as being significantly more effective than *Retrospective*. Other system differences were not significant.

3.2.2 Usability

Subject opinion on system usability was measured using six Likert scales. The values for "Usability" in Table 3 are the means (and standard deviations) of these six statements for each system. A one-way repeated measures ANOVA of subject responses yielded no significant differences between systems in regard to overall feelings of usability $\underline{F}(2, 430) = 1.78$, $\underline{ns}$.

3.2.3 Engagement

Subject opinion on their level of engagement when using each system was measured using four semantic differentials: The values for "Engagement" in Table 3 are means (and standard deviations) of these four differentials for each system. A one-way repeated measures ANOVA of subject responses yielded significant differences between systems in regard to overall feelings of engagement $\underline{F}(2, 286) = 10.79$, $\underline{p} < .001$. Post hoc comparisons using the Tukey Test revealed that *RealTime* led to significantly higher feelings of engagement than *Baseline*, and *Baseline* led to significantly higher feelings of engagement than *Retrospective*.

3.2.4 Enjoyability

In a similar way to the level of engagement, subject opinion on how much they enjoyed their searches when using each of the systems was measured on four semantic differentials. The values for "Enjoyability" in Table 3 are the means (and standard deviations) of these differentials for each system. A one-way repeated measures ANOVA yielded significant differences between systems in regard to overall feelings of enjoyment $\underline{F}(2, 286) = 7.76$, $\underline{p} = .0005$. Post hoc comparisons using the Tukey Test revealed that *RealTime* led to significantly higher feelings of enjoyment than *Baseline*, and *Baseline* led to significantly higher feelings of enjoyment than *Retrospective*.

3.2.5 Preferences

After using all three systems, subjects were asked to indicate which of the three systems they preferred for a variety of criteria: learnability, ease of use, helpful for known-item tasks, helpful for exploratory tasks, and overall. The proportion of responses assigned to each system for each criterion is shown in Table 4.

Table 4. Subject preferences (values are percentages, rounded to nearest point).

	Baseline	RealTime	Retrospective	No preference
Easier to learn	**47**	11	17	25
Easier to use	**42**	25	22	11
Known-item	**72**	22	0	6
Exploratory	11	**44**	39	6
Overall	28	31	**36**	6

A one-way independent measures ANOVA revealed significant effects for each of the criteria except "Easier to use" and "Overall" all $\underline{F}(3,105) \geq 3.92$, all $\underline{p} \leq .011$. Post hoc Tukey Tests [15] revealed that *Baseline* was easier to learn than *RealTime* and *Retrospective*, and preferred over all systems for known-item tasks. Tukey Tests also revealed that *RealTime* and *Retrospective* were preferred over *Baseline* for exploratory tasks.

3.2.6 Comments

Subjects were asked to describe what they liked and disliked about each system in two open-ended questions in the final questionnaire. We now summarize subject comments. Unique subject identifiers are shown next to the each of the comments.

BASELINE

Subject comments included that *Baseline* was "easy to use" (S20), "easy to learn" (S18), "familiar" (S26), and "straightforward" (S17). However subjects felt they "had to define queries clearly" (S26), and that "[missing support] made it difficult to narrow the search" (S28). Subjects were most familiar with this system but they did not like having to formulate queries, or be responsible for the search strategy they selected.

REALTIME

Subject comments included that *RealTime* "offered words (paths) to go down that I might not have thought of on my own" (S22), and was "a real time-saver by providing me w/suggested words as I typed" (S15). However, they also felt it was "slow to react" (S24), and "slow right now but will certainly be useful once the speed increases" (S12). *RealTime* received positive comments, although numerous subjects commented negatively on the one or two second delay after pressing the spacebar before the list of candidate query expansion terms updated.

[15] We do not report pair-wise differences between the systems and the "No preference" responses.

Retrospective

Subject comments included that in *Retrospective* "if results were bad, I could look at suggested terms" (S14), and that it "helped me clarify my search if I didn't get what I wanted the first time" (S21). However, subjects also remarked "I already found correct sites by the time the suggested words were provided" (S1), the system "gave help when I didn't really need it (i.e. after query)" (S18), and "the help wasn't available until I had committed to clicking search" (S14). Comments on *Retrospective* suggest that subjects felt offering IQE after the query has been submitted was perhaps not as helpful as doing so during query formulation.

3.2.7 Usage

As well as using the explicit measures of subject satisfaction described in this section so far, we can also utilize usage information as an implicit measurement of subject satisfaction. From the interaction logs we extracted details of the proportion of recommended expansion terms that were added by subjects when using *RealTime* and *Retrospective*.[16] In Table 5 we present the proportion of queries that involved the addition of at least one system generated query expansion term. We present results for all tasks of each type, and just for those tasks that involved at least one instance of query reformulation. 55% of tasks attempted on *RealTime* and 62% of tasks attempted on *Retrospective* had more than one query iteration.[17] "Overall" percentage is included for reference only, and is not part of the statistical analysis.

Table 5. Use of query expansion (values are percentages, rounded to nearest point).

	RealTime		Retrospective	
	All tasks	Tasks with reformulation	All tasks	Tasks with reformulation
Known-item	34	49	20	39
Exploratory	54	63	36	47
Overall	44	57	28	44

We applied two-way ANOVA for "All tasks" and for "Tasks with reformulation" separately.

All Tasks

There was a significant effect of task type F(1,71) = 9.04, p = .004 and experimental system F(1,71) = 7.08, p = .010 on the number of recommended query expansion terms added. That is, more recommended query expansion terms were added for exploratory searches, and significantly

[16] *Baseline* did not offer direct query expansion support.

[17] *RealTime*: Known-item: 46%, Exploratory: 64%; *Retrospective*: Known-item: 50%, Exploratory: 74%

more recommended terms were added when using *RealTime*. There was no significant interaction between task type and system and the number of recommended terms added F(1,71) = .16, ns. Higher task complexity led to an increase in the use of query expansion, and an increase in the number of terms drawn from *RealTime*.

TASKS WITH REFORMULATION

There was no significant effect of task type F(1,164) = 2.01, ns, or experimental system F(1,164) = 2.90, ns on the number of recommended terms added by subjects. There was no significant interaction between task type and system and number of terms added F(1,164) = .16, ns. Task complexity had less impact on the use of query expansion for tasks with reformulation, although the results followed a similar trend to those for "All tasks."

Subjects remarked during the experiment that on some occasions they may have typed in a recommended term or a semantically similar variant of it rather than moving their hand from the keyboard to select a term explicitly with the mouse pointer. The findings on usage we reported in this section are therefore a lower bound of query expansion usage.

3.2.8 Summary

In this section we presented results for several aspects of satisfaction: usefulness, usability, engagement, and enjoyment. Subjects were generally more satisfied with *Baseline* with respect to effectiveness and usability, and were generally more satisfied with *RealTime* with respect to engagement and enjoyment. Our hypothesis (H_S) was therefore supported for engagement and enjoyment, but not for ease of learning and use. It is not that *RealTime* had low usability or learnability, but rather subject familiarity with *Baseline* led to it being preferred on these two aspects. Subjects preferred *Baseline* for know-item tasks, and the two query expansion systems for exploratory tasks. They were found to actually use the query expansion functions in a third of the known item and more than half of the exploratory tasks.

3.3 Quality of Queries

The aim of query expansion is to help searchers formulate better quality queries that may be used to retrieve better search results. In this section we analyze data related to hypotheses $H_{QQ(k)}$ and $H_{QQ(e)}$ for query quality. For all of the 1174 queries posed, a query quality score was determined on a 5-point scale by two different judges. To minimize the differences in quality estimates by the two judges, the mean of the two judgments was taken as the overall query quality for each query. We also analyzed the composition of queries in two further ways: query iterations, and unique query terms used.

3.3.1 Judged Queries

During composition of the initial query, the interface displayed a text box for query entry, a "search" button to initiate the retrieval operation, and in RealTime a list of recommended words updated using RTQE. In all systems subjects could not see search results until they had composed their query and clicked "search." We analyzed data relating to the mean average query quality ratings for the initial query alone (since it allowed us to isolate the RTQE) and all queries are shown in Table 6.

Table 6. Query quality (mean average, standard deviation).

	Baseline		RealTime		Retrospective	
	Initial Query	All Queries	Initial Query	All Queries	Initial Query	All Queries
Known-item	2.14	1.88	**1.86**	1.84	2.07	**1.82**
Exploratory	2.01	1.75	**1.70**	1.72	1.99	**1.65**
Overall	2.07	1.81	**1.78**	1.77	2.03	**1.73**

We applied two-way ANOVA for "All Tasks" and for "Tasks with reformulation" separately.

INITIAL QUERY

There was a significant effect of system on initial query quality F(2,142) = 12.37, p < .001. Post hoc comparisons using a Tukey Test reveal significant differences between *RealTime* and the other systems for each of the task types. There were no statistically significant effects between the task types F(1,71) = 3.26, ns, and no interaction effects between task and system and initial query quality F(2,142) = .19, ns. RTQE appears to lead to better quality initial queries.

ALL QUERIES

There were no statistically significant differences across the three systems F(2,142) = .95, ns. There were statistically significant effects between the task types F(1,71) = 4.98, p = .029, with lower means on the known-item tasks. There were no interaction effects between task and system F(2,142) = .24, ns. The presence of the additional information in search results appears to help subjects using *Retrospective* and *Baseline* achieve a level of quality that is statistically indistinguishable from that of *RealTime*.

3.3.2 Query Iterations

Query expansion can be helpful for query reformulation, therefore, we investigated the number of query iterations per task.

For the known-item tasks, subjects composed a total of 432 queries ranging between 1 and 10 queries per task (M = 1.96, SD = .77). For the exploratory tasks, subjects posed a total of 742 queries ranging between 1 and 21 queries per task (M = 3.43, SD = 1.92). A two-way repeated measures ANOVA revealed no significant effects of systems F(2,142) = 1.06, ns. There were statistically significant effects found between the task types F(1,71) = 12.53, p < .001. There was also a large range of total queries expressed by individual subjects. Subjects posed an average of 32.6 queries for all 12 tasks (M = 2.60, SD = 1.61), ranging from one subject who posed 16 queries for all tasks and one who posed 60 queries for all tasks. There was no significant interaction between task type and system on number of query iterations F(2,142) = .83, ns.

3.3.3 Unique Query Terms

As an additional measure of query quality we used interaction logs to count the number of unique query terms per query (effectively the query length) and per search task (all queries from each subject on that task). In Table 7 we present the mean query term count per query, and per task, averaged across task type.

Table 7. Mean average unique query term count.

	Baseline		RealTime		Retrospective	
	Query	Task	Query	Task	Query	Task
Known-item	4.25	4.90	4.23	5.21	4.21	5.10
Exploratory	4.12	6.15	4.21	6.53	4.41	6.82
Overall	4.17	5.53	4.26	5.87	4.26	5.96

The queries our subjects submitted were typically longer than standard Web search queries, which tend to be 2-3 terms in length (Spink *et al.*, 2002). This may be because they were provided with short textual task descriptions from which they could extract terms for use in devising their query statements. Analysis of the query logs generated during our study supported this belief, and showed that 88% of the terms used in the initial query for known-item searches, and 78% of the terms used in initial queries for exploratory searches also occurred in the task descriptions.

QUERY

Two-way independent measures ANOVA revealed no significant effect of tasks F(1,1152) = .09, ns or systems F(2,1152) = 1.18, ns on the number of unique terms per query. There was no significant interaction between task type and system and unique query terms per query F(2,1152) = 1.00, ns. Analysis of the number of unique terms per query revealed no differences between experimental systems or search tasks.

TASK

The results show that there are fewer unique query terms for the known-item searches, and a two-way repeated measures ANOVA revealed a significant effect of task type F(1,71) = 18.65, p < .001 but not systems F(2,142) = 1.08, ns on the number of unique terms per search task. There was no significant interaction between task type and system F(2,142) = .30, ns. Analysis of the number of unique query terms per task revealed that subjects submitted more unique query terms for the exploratory tasks.

3.3.4 Summary

In this section we presented results for several aspects of query quality. The results show that queries for the known-item tasks were generally of poorer quality than the exploratory tasks, and subjects generally submitted fewer queries for known-item tasks. *RealTime* led to the creation of better quality initial queries than *Baseline* or *Retrospective*, since it offered IQE before a retrieval had been performed, where subjects may lack knowledge about nature of search results. There were no differences in the number of unique query iterations between the three experimental systems. This is an interesting finding, as in the Koenemann and Belkin (1996) study described earlier, the "penetrable" (explicit) RF system, implementing IQE prior to use in generating a new set of results, led to subjects needing fewer query iterations than a baseline interface with no query formulation support. In that study perhaps the inclusion of explicit RF, together with the direct interruption of query execution, led to subjects spending more time engaged in search activities other than query formulation. The presence of search results appears to enhance query quality in *Baseline* and *Retrospective*, but not *RealTime*. When searchers are using RTQE they may be so engaged in its use as to ignore the other information available to them from meta-information about documents available in search engine result lists. An alternative explanation is that query quality on *RealTime* represents an upper bound, and the other systems are able to gain equivalence when subjects have additional information. Queries generated on *RealTime* and *Retrospective* had slightly more unique terms than *Baseline*, although not significantly more.

These systems may introduce alternatives not considered by subjects. Hypotheses $H_{QQ(k)}$ and $H_{QQ(e)}$ are partially supported by these findings, although not fully since they did not transcend all queries. In the next section we discuss the findings of our study and their implications for the deployment of RTQE.

4. Discussion and Implications

The effectiveness of RTQE appears to depend on a number of factors. It appears more useful for exploratory tasks and early in a search task (i.e., before the first set of results has been displayed), when searcher needs may be most uncertain. This is in line with the findings of an earlier study by Fowkes and Beaulieu (2000), who showed that searchers are more likely to use IQE when information needs are vague, little relevant information is being retrieved, or when the search task is complex or difficult. When other sources of information, such as result lists become available to searchers then systems implementing post-retrieval query reformulation appear to perform equally well. It seems that the additional information available to searchers obviates the potential additional benefits of RTQE. It is worth considering that in this study the subjects were fairly sophisticated searchers and posed fairly long queries and sometimes used advanced strategies than might mitigate the value that query expansion can add for novice searchers.

Despite its effectiveness, a lack of uptake has been one reason that IQE has not been as widely deployed in IR systems (Dennis *et al.*, 1998). However, given this concern, there have been very few user studies that have considered this issue. Beaulieu *et al.*'s (1997) studies revealed reluctance for searchers to take advantage of IQE, suggesting that the additional task of judging expansion terms is itself a difficult one which searchers will avoid. Using a dual-task technique to measure cognitive load, Bruza *et al.* (2000) showed that improved result set relevance from using terminological feedback does indeed come at the cost of increased cognitive load in evaluating the feedback. Anick (2003) suggested that usability tests conducted at AltaVista prior to external release of a query expansion tool indicated that many users did not even notice feedback terms when embedded within an already textually full results page. In a log-based study following that usability study Anick found that uptake of IQE as implemented in AltaVista at the time (i.e., with clickable hyperlinks), was approximately 14% of all sessions, and 25% of all sessions involved some sort of query reformulation.[18] The system implemented in that study was a PRF system that displayed terms following query submission, and closely resembled

[18] This should also be compared with uptake rates of 11% and 19.5% in the two interfaces offering IQE investigated in the study by Beaulieu (1997) described in the introduction to this article.

Retrospective. The usage of query expansion in *Retrospective* was higher than reported in Anick's study (approximately double the amount in both cases). This may be related to the algorithm used to select query expansion terms, and the fact that our study was conducted in a laboratory setting where subject attention was on the query expansion techniques. Other reasons could be that Anick's measurement was from estimated sessions, which can be difficult to detect from interaction logs (our findings on IQE uptake are from tasks with well-defined start and stop points), or that many Web searches are navigational and do not include query refinement.

In this study we investigated an alternative way of displaying the expansion terms, during query formulation, and compared it against a more traditional approach of displaying them alongside the search results (similar to the approach used by Anick). Subjects tended to accept query expansion terms more often with *RealTime* than with *Retrospective*, even though the terms are scored using the same algorithm. Since the only difference in these systems was the timing of the query support, it is likely the increased uptake in the study could be related to the alternative presentation technique, and subjects' increased sense of engagement when using it. This is a promising finding, although it will need to be verified in an operational setting where searchers are less aware of their participation in an experiment.

Showing query expansion terms before searchers have seen any search results has the potential to speed up their searching, but it can also lead them down incorrect search paths. For example, in Figure 1, the correct answer to the question "Who was the first woman in space?" is Soviet cosmonaut Valentina Tereshkova. However, in response to the query "first woman in space" Google returns Sally Kristen Ride as the top result. As our approach uses PRF, that document is assumed to be as relevant as those containing information on Soviet cosmonauts, and explains why the term "ride" appears in the recommended terms shown in Figure 1. Since searchers are unable to predict the effect of adding a query expansion term, they may add erroneous terms such as "ride" in this case, that may lead them in the wrong direction. This problem is more acute in the initial query where searchers have not yet viewed any of the retrieved results. Beaulieu and Jones (1998) referred to this and similar issues as relating to the *functional visibility* of an interface using query expansion. They suggested that the searcher be aware of what options are available at any stage (including query modification options) but they also be aware of the *effect* of these options. For example, in this case a preview of the results that would be obtained, perhaps appearing on mouse hover over an IQE would allow the searcher to make more rapid assessments of term impact than repeated query reformulation. A lack of functional visibility is a problem not just with our approach but with all systems that implement PRF, and is more apparent for known-item searches where there may only be a single correct answer. Research in question answering (e.g., Dumais *et al.*, 2002) may be useful in selecting documents containing only the correct answers that could then subsequently be used for PRF.

The underlying retrieval algorithm in can also affect how IQE is used. Systems based on the vector-space and probabilistic models (such as the Okapi system mentioned in Beaulieu *et al.* (1997)) typically allow searchers to select multiple feedback terms per iteration. They do this since result quality of these systems generally improves with longer search queries. In contrast, in systems based on a Boolean model (such as many of the major Web search engines), searchers are expected to select one term at a time for refinement. They do this since the number of search results can drop dramatically for longer queries, and unless the query exactly matches the target document, result quality will also be adversely affected. As another aspect of functional visibility, searchers should be made aware of how the expanded queries will be handled by the search system.

We observed that many of the queries beyond the first iteration were simply syntactic variants of the initial search statement (e.g., spelling corrections), or used the initial query as a skeleton for elaboration. The initial query appears important in determining search success, particularly when PRF is employed (i.e., poor initial queries lead to poor search results). When RF is offered explicitly (Oddy, 1977) or implicitly (Kelly & Teevan, 2003) the searcher may often be given the option of correcting the system's internal representation of the information need, and stopping the "formalized" and "conscious" needs (Taylor, 1968) from diverging too greatly. However, in systems implementing PRF the only way that searchers can control this internal representation is through the selection of query terms. That is, they have no other way to tune what information the system regards as relevant. This reliance on the query may be appropriate when the information need is well-defined (as in the known-item searches), but when it is vague (as in the exploratory searches) then using the query as the only control mechanism is potentially problematic. When using *RealTime* searchers generally succeeded or failed in their performance of the task on the basis of the query term selections they made in the initial query-entry screen. For example, subjects looking for the name the first woman in space who selected RTQE terms such as "nasa", "ride" and "astronaut" from the recommended list of terms during their first query iteration were more likely to end up with the incorrect answer for the task. Since searchers cannot preview the documents that will be generated by the query terms they select the technique has the potential to lead searchers astray. Care must therefore be taken when implementing RTQE to ensure that where possible terms likely to mislead searchers are not chosen for recommendation to them, or the estimated effect of adding a term is made visible to searchers.

An analysis of query quality showed that RTQE improved the quality of initial queries for both known-item and exploratory tasks, making it potentially useful during the initiation of a search, when searchers may be in most need of support. This is an important finding. If the RTQE is capable of enhancing the quality of some queries, and improving search effectiveness measured via factors such as task completion times, *and does not having a detrimental effect on other aspects of search performance*, then there is a case for implementing it as a permanent feature of all search systems. A promising feature of RTQE is that it does not force searchers to use the technique, or indeed do anything beyond the scope of their normal search activities. This makes it very attractive to commercial search engine developers looking to support large numbers of searchers in an efficient way. For example, at the time of writing this article Google has recently introduced a variant of RTQE in its toolbar for use with Microsoft Internet Explorer, and such a facility has been available as a Firefox browser extension for some time.[19] Rather than using PRF, the toolbar component uses query logs and personal search histories to make recommendations about potential query completions. However, in a similar way to our implementation, inappropriate terms (in this case common misspellings) appear in the list suggested by the component, introducing potential query skew if they are not handled by the search system.

Baseline was preferred for its familiarity and lack of distraction. Subjects generally found that this interface was easiest to learn and easiest to use. The dynamics of *RealTime* made it more engaging and enjoyable. Despite subject preferences, many subjects suggested that RTQE should be offered faster than it was in *RealTime*. The implementation of RTQE in *RealTime* was obviously not ideal. The short time delay between the subject releasing the spacebar and the new set of expansion terms being generated frustrated some subjects, and surely had a negative impact on their perceptions of the technique and willingness to use it. This delay was attributable to the requirement that a new retrieval be performed each time a search was initiated. To address subject concerns it may be appropriate to implement caching for commonly submitted queries in any future versions of this component that we may implement.

[19] http://toolbar.google.com

The nature of the query expansion support offered did not appear to affect the number of query terms, or the number of query iterations. In fact, it was *Retrospective* that led on average to the highest query quality across all queries. This may be because the system provided two types of support: searchers were shown the ten query expansion terms, and they were shown the titles, abstracts, and URLs of the documents from which those terms were derived. The presence of this information may provide an additional source from which to choose terms, but perhaps more importantly, give practiced, motivated searchers a sense of the type of documents that their query retrieved, and a sense for the context within which query modification terms occur in the collection.

Our efforts to develop a measure of query quality yielded only a beginning set of criteria for quality that might be the basis for an automatable model. The use of facets to judge query quality tended to favor comprehensive queries that included all facets, whereas, popular topics lend themselves to search engine optimizations that are tuned to short queries more typically posed by the overall search population. Thus, query quality appears to be dependent on an array of contextual elements. Identifying and classifying these elements remains an important challenge for future work. Although the RTQE led to differences in query quality in the initial retrieval these differences were not apparent in the result quality ratings assigned by our panel of two judges. There are at least two possible explanations for this finding: the underlying search system may be insensitive to small differences in query quality that are detectable by human assessors, or the level of inter-rater agreement between our two judges was insufficient to consistently use as the basis for this analysis. Upon reflection we suspect that the true explanation involves some combination of the two we propose here. Certainly, today's search engines are continually tuned to events and user behavior and a high quality query one day may yield poor results at a different time. It may be prudent therefore to rerun aspects of the query quality and result quality analyses using a larger panel of human judges. We reserve this for future work, although the findings we report here are sufficiently dependable to suggest the value of the approach.

5. Conclusions

We have presented a study of a technique to support searchers in composing queries to submit to search systems. The technique, know as Real-Time Query Expansion offers query expansion terms to searchers as they enter queries, and updates following each term to reflect potential completions of the search query. Techniques such as this represent a step forward in the development of useable IQE techniques to assist searchers. The hypotheses we developed for this study were meant to assess the value of the component from multiple perspectives. Although not all of the hypotheses were borne out with statistically reliable evidence, the general patterns of the data suggest that RTQE increases the usage of query expansion, searchers will be engaged in using it, that it improves the quality of initial queries, and may lead to higher satisfaction. However, the technique has the potential to introduce query skew, and if this is to be implemented for large-scale use then care must be taken to implement it in such a way as to offer searchers some information about the predicted effect of their query expansion decisions. It is through techniques of this nature that the potential effectiveness of IQE can be realized, and problems with uptake overcome. This study has given us insight into the circumstances under which RTQE performs well, how searchers use it, and potential enhancements for the approach. The future of IQE may lie in techniques that couple query expansion more closely with searchers' normal information-seeking behaviors.

6. Acknowledgements

We would like to thank Thomas Ciszek for his help in collecting data at the UNC site, and Sanghee Oh and Laura Sheble, our query and results quality judges. We would also like to thank our experimental subjects whose comments and enthusiasm were much appreciated.

7. References

Allan, J. (1995). Relevance feedback with too much data. In *Proceedings of the 18th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 337-343.

Anick, P. (2003). Using terminological feedback for web search refinement: a log-based study. In *Proceedings of the 26th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 88-95.

Bates, M. (1979) Information search tactics. *Journal of the American Society for Information Science*, 30: 205-213.

Beaulieu, M. (1997). Experiments with interfaces to support query expansion. *Journal of Documentation*, 53(1): 8-19.

Beaulieu, M., Do, T., Payne, A., and Jones, S. (1997). ENQUIRE Okapi Project. *British Library Research and Innovation Report 17*.

Beaulieu, M. and Jones, S. (1998). Interactive searching and interface issues in the Okapi best match probabilistic retrieval system. *Interacting with computers*, 10(3): 237-248.

Bell, D. and Ruthven, I. (2004). Searchers' assessments of task complexity for web searching. In *Proceedings of the European Conference on Information Retrieval*, pp. 57-71.

Borlund, P. (2000). Experimental components for the evaluation of interactive information retrieval systems. *Journal of Documentation*, 56(1): 71-90.

Bruza, P.D., Dennis, S., and McArthur, R. (2000). Interactive internet search: keyword directory and query reformulation mechanisms compared. In *Proceedings of the 23rd Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 280-287.

Busha, C.H. and Harter, S.P. (1980). *Research methods in librarianship: Techniques and interpretation*. Library and information science series. New York: Academic Press.

Byström, K. and Järvelin, K. (1995). Task complexity affects information seeking and use. *Information Processing and Management*, 31(2): 191-213.

Croft, W.B. and Thompson, R.H. (1987). I^3R: A new approach to the design of document retrieval systems. *Journal of the American Society for Information Science*, 38(6): 389-404.

Cronen-Townsend, S., Zhou, Y., and Croft, W.B. (2002). Predicting query performance. In *Proceedings of the 25th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 299-306.

Dennis, S., McArthur, R. and Bruza, P. (1998). Searching the WWW made easy? The cognitive load imposed by query refinement mechanisms. In *Proceedings of the Third Australian Document Computing Symposium.*

Dumais, S., Banko, M., Brill, E., Lin, J., and Ng, A. (2002). Web question answering: is more always better? In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 291-298.

Efthimiadis, E.N. (1996). Query expansion. *Annual Review of Information Systems and Technology*, 31: 121-187.

Fowkes, H. and Beaulieu, M. (2000). Interactive searching behaviour: Okapi experiment for TREC-8. In *Proceedings of the IRSG 2000 Colloquium on IR Research.*

Harman, D. (1988). Towards interactive query expansion. In *Proceedings of the 11th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 321-331.

Jinxi, X. and Croft, W.B. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 4-11.

Karat, C.M., Halverson, C., Horn, D. and Karat, J. (1999). Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computer Systems*, pp. 568-575.

Kelly, D. and Teevan, J. (2003). Implicit feedback for inferring user preference. *SIGIR Forum,* 37(2): 18-28.

Koenemann, J. and Belkin, N.J. (1996). A case for interaction: A study of interactive information retrieval behavior and effectiveness. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computer Systems*, pp. 205-212.

Lam-Adesina, A.M. and Jones, G.J.F. (2001). Applying summarization techniques for term selection in relevance feedback. In *Proceedings of the 24th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1-9.

Magennis, M. and Van Rijsbergen, C.J. (1997). The potential and actual effectiveness of interactive query expansion. In *Proceedings of the 20th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 324-332.

Oddy, R. N. (1977). Information retrieval through man-machine dialogue. *Journal of Documentation,* 33(1): 1-14

Ruthven, I. (2003). Re-examining the potential effectiveness of interactive query expansion. In *Proceedings of the 26th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 213-220.

Salton, G. and Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4): 288-297.

Sihvinen, A. and Vakkari, P. (2004). Subject knowledge improves interactive query expansion assisted by a thesaurus. *Journal of Documentation*, 60(6): 673-690.

Sparck-Jones, K. (1981). *Information retrieval experiment*. Butterworths, London.

Spink, A., Jansen, B. J., Wolfram, D., and Saracevic, T. (2002). From E-Sex to E-Commerce: Web Search Changes. *IEEE Computer*, 35(3): 107-109.

Taylor, R. (1968). Question-Negotiation and Information Seeking in Libraries, *College and Research Libraries* 29: 178-194.

White, R.W. (2004). *Implicit feedback for interactive information retrieval*. Unpublished doctoral dissertation. University of Glasgow, Glasgow, United Kingdom.

Wildemuth, B. and Moore, M. (1995). End-user search behaviors and their relationship to search effectiveness. *Bulletin of the Medical Library Association*, 83(3): 294-304.