A Study on the Effects of Personalization and Task Information on Implicit Feedback Performance

Ryen W. White
Microsoft Research
One Microsoft Way
Redmond, WA 90852 USA
ryenw@microsoft.com

Diane Kelly
School of Information and Library Science
University of North Carolina
Chapel Hill, NC 27599 USA
dianek@email.unc.edu

ABSTRACT

While Implicit Relevance Feedback (IRF) algorithms exploit users' interactions with information to customize support offered to users of search systems, it is unclear how individual and task differences impact the effectiveness of such algorithms. In this paper we describe a study on the effect on retrieval performance of using additional information about the user and their search tasks when developing IRF algorithms. We tested four algorithms that use document display time to estimate relevance, and tailored the threshold times (i.e., the time distinguishing relevance from non-relevance) to the task, the user, a combination of both, or neither. Interaction logs gathered during a longitudinal naturalistic study of online information-seeking behavior are used as stimuli for the algorithms. The findings show that tailoring display time thresholds based on task information improves IRF algorithm performance, but doing so based on user information worsens performance. This has implications for the development of effective IRF algorithms.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information search and retrieval – *relevance feedback.*

General Terms

Experimentation, Human Factors

Keywords

Implicit feedback, Evaluation

1. INTRODUCTION

When Information Retrieval (IR) was first studied as an experimental discipline, automated search systems were incapable of supporting representations of users' information needs that extended beyond their initial textual queries. However, it has been understood for some time that these representations are only approximations of searchers' information needs [2, 22]. Since more complete representations generally lead to more precise search results, it can be advantageous to supplement textual queries with additional sources of information, such as enhanced queries created by users [15], term suggestions offered by systems [7], or documents users have found relevant [17].

Advances in information technology have facilitated the development of new interaction paradigms between humans and systems. It is now possible to leverage user-system interactions as additional information becomes available for use. For instance, eye-tracking, direct manipulation of on-screen components, and physiological measures, provide more information about interaction than once available. Currently, there are numerous efforts in disciplines that extend beyond IR to understand how such information can be used to improve human-machine interactions [e.g., 18]. In the area of IR, Implicit Relevance Feedback (IRF) [16] *algorithms* use information created as a byproduct of users' interactions with information to help users by customizing the support they offer. Support may include the recommendation of additional query expansion terms [26], or the automatic retrieval of new document sets based on characteristics of the interaction [1].

Typical studies of IRF have sought to determine whether a correlation exists between measures such as document selection and document display time, and document relevance [4, 8]. Whilst it is important to understand what measures can be accurate predictors of relevance, it is also important to understand what mediating factors, perhaps not immediately visible from information-seeking behavior, can influence the effectiveness of IRF in supporting users. Previous research has shown that factors such as task, user experience, and stage in the search can affect the utility of IRF [14, 25]. However, these studies have not looked at the effect of using such information to create IRF algorithms, and evaluate the resultant impact on retrieval performance.

In this paper we present a study of IRF algorithms. The aim of this study is not to add another finding about the reliability of IRF measures, but rather investigate the effect of using task and user information during IRF algorithm development. Additional information about users and tasks is used to tailor the relevance threshold for document display time adopted in four algorithms developed for this study. These algorithms vary the presence / absence of task and user information, and are compared with each other with the goal of determining which performs best and under what circumstances. The study uses interaction logs gathered during a naturalistic study of online information-seeking behavior as stimuli for the algorithms under test. To help perform the study and enhance repeatability we developed an automated evaluation framework.

The remainder of this paper is structured as follows. In Section 2 we describe the evaluation framework, and describe the study itself in Section 3. We present findings in Section 4, discuss them in Section 5, and conclude in Section 6.

CIKM'06, November 5–11, 2006, Arlington, Virginia, USA.
Copyright 2006 ACM 1-59593-433-2/06/0011...$5.00.

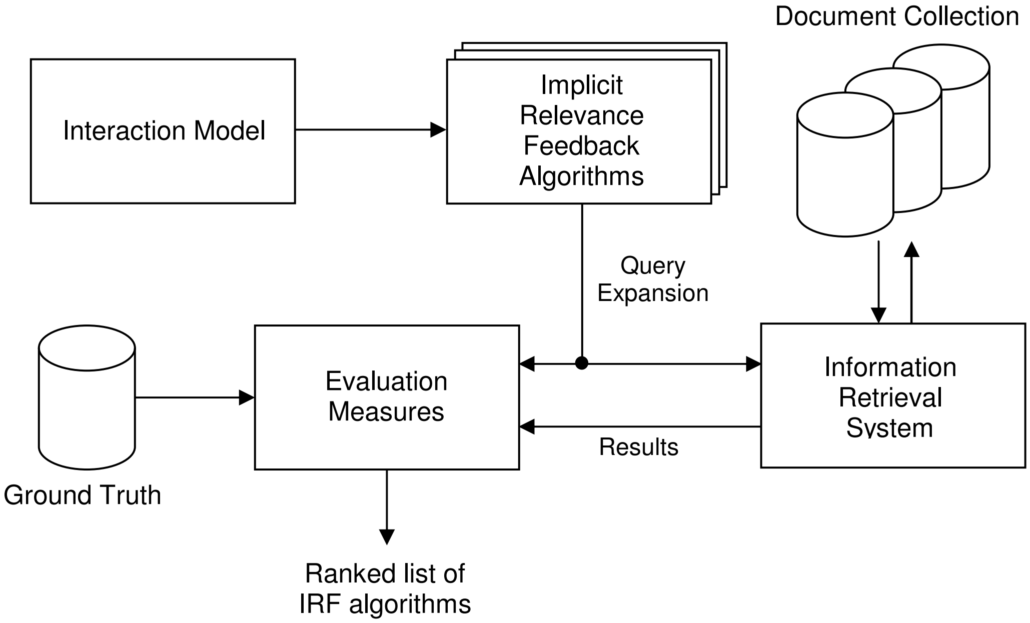
Document Collection
Interaction Model
Implicit Relevance Feedback Algorithms
Query Expansion
Evaluation Measures
Information Retrieval System
Results
Ground Truth
Ranked list of IRF algorithms

Figure 1. Evaluation framework deployed in study.

2. EVALUATION FRAMEWORK

Cranfield-style evaluation frameworks [5] have allowed researchers to study the effectiveness of IR systems and their components. Such evaluations generally contain three key parts: a document collection, tripartite topic descriptions derived from real information needs, and relevance assessments made by those who generated the topic descriptions. The aim of the evaluation is to determine the effectiveness of a given IR system (i.e., the independent variable) in retrieving documents relevant to a given set of topics. System performance (i.e., the dependent variable) is quantified using metrics such as precision and recall, allowing systems to be compared. In that framework, the interaction model is generally limited to query submission, and the explicit provision of feedback about document relevance (if explicit Relevance Feedback (RF) [3] is used). However, IRF algorithms require more information about interaction behavior than queries and explicit RF. For this reason, richer interaction models are essential in any evaluation framework designed to test algorithms that use IRF.

To perform our study we developed an evaluation framework that allowed us to incorporate a potentially more detailed interaction model directly into the experiment. The framework automated experimental processes including the provision of stimuli to the algorithms, the creation of expanded query statements, and the retrieval and scoring of experimental runs. The structure of the framework is shown in Figure 1. It is divided into six key components:

- The **interaction model** is a characterization of what is important about the user interaction data that serves as feedback to the IRF algorithms. This can be composed of logs gathered during longitudinal studies (e.g., [13]), or Web search engine interaction logs (e.g., [1]). The particular types of interactions logged prior to evaluation determine the sources that can be monitored in the framework. For example, a decision not to log document retention events, such as bookmarking and printing, means researchers cannot study the role of retention as IRF. It is conceivable that these logs could be replaced by simulations of searcher interaction behavior (e.g., [24, 27]) if reasonably accurate simulations could be constructed. An additional component to filter the IRF measures to be used plugs directly into the interaction model. For example, the logs used in this study allowed us to choose from many possible measures (e.g., document scrolling, document retention, document display time). However, to reduce the likelihood of confounding effects caused by interactions between measures (and for other reasons we will touch on later) we chose to concentrate on a single measure: document display time, excluding all other measures from presentation to the algorithms[1].
- **IRF algorithms** are evaluated by the framework. They take user interaction as input, and use the textual content of documents conforming to the relevance criteria defined as part of the algorithm to generate a set of candidate query expansion terms to add to the initial query.
- **Ground truth** information contains judgments on the usefulness of documents viewed during a search, generally in relation to a pre-determined search topic. It is assumed that this information is captured during an initial judgment phase separate from the framework, where assessments are generally provided at some point after viewing a document, either immediately [4, 8], or after a short time delay [13]. The decision about *when* to capture explicit assessments on documents viewed involves a trade-off between: the timeliness of the feedback, the extent of the feedback, and the degree of intrusion on the searcher's interaction.

[1] Ignoring other sources of IRF means we lose information about other aspects of the search context that may directly or indirectly affect document display times.

- The **document collection** comprises all documents for which any interaction was logged.
- The **information retrieval** system retrieves sets of search results from the document collection in response to the expanded queries the algorithms generate.
- **Evaluation measures** compute a score for each algorithm based on the ground truth information and the results retrieved by the information retrieval system.

The next section describes the study that uses this framework.

3. STUDY

The study we describe in this section uses interaction logs from a longitudinal study of seven subjects' interaction behaviors over a period of fourteen weeks. For each document the logs contain data on the task, the user, its usefulness for the task, and how subjects interacted with it. The richness of these logs presents a unique opportunity to study IRF algorithm performance. The current study has a 2 × 2 factorial design, with the independent variables (or factors) of *task information* (i.e., the display time thresholds of relevant documents for that task) and *user information* (i.e., the display time thresholds of relevant documents for that subject). Each factor has two levels: present or absent. One algorithm was developed by the researchers for all combinations of these two factors, resulting in four algorithms in total. The dependent variable in the study was result *precision*, measured through the proportion of relevant documents in the top ten retrieved, and across all documents retrieved. Each component of the study is described in more detail below. We begin by describing the research questions that drive our investigation.

3.1 Research Questions

Given a rich set of interaction logs, and some candidate IRF algorithms, the study aims to determine whether: (i) IRF algorithms personalized to users can outperform IRF algorithms that ignore personalization, (ii) IRF algorithms developed using task information can outperform algorithms that ignore such information, and (iii) IRF algorithms developed using a combination of personalization and task information can outperform algorithms using either source. All algorithms are compared against a baseline algorithm with a single display time threshold across all subjects. Performance is measured based on retrieval effectiveness following query expansion. That is, the relevance of documents retrieved once additional terms have been added to the original query by the IRF algorithms. The original query is derived from task labels assigned by subjects in the longitudinal study. More details are provided in Section 3.5.

3.2 Interaction Model

The interaction model used in this study is based on interaction logs gathered during a study of seven users' online information-seeking behaviors [13][2]. Throughout the study subjects' online activities were monitored with various pieces of logging and evaluation software. The study was naturalistic, and subjects were each given a laptop computer equipped with the WinWhatWhere Investigator. The software unobtrusively monitored and recorded subjects' interactions with all applications including the operating system, web browsers, and word processors. Information such as applications used, URLs visited, start, finish and elapsed times for interactions and all keystrokes, including queries, were recorded and stored on a protected file on the laptop. Subjects' web browsers were further directed through a proxy logger; this direction did not disrupt subjects' activities or cause any noticeable lag times. The proxy logger was a custom-built logging application that resided on a local proxy server, and saved a copy of each page request made by subjects. For a more detailed description of the logging procedures see [13].

For this study, all identifying information (e.g., credit card numbers, passwords) was removed from the interaction logs. The logs were also recoded into XML files with a suitable document type definition by a trained graduate student over a period of three weeks. During this process, each individual document viewed by subjects was marked-up with information about the interaction, such as the length of time the document was displayed in the subject's web browser, how often the document was displayed during a one-week time period, and if the subject printed, saved or bookmarked the document. Information about the subjects' context when viewing documents was also gathered and added to each document representation. Subjects associated a self-identified task and topic with each document, and also indicated things such as how long they expected to be working on tasks, how often they worked on particular tasks, and how familiar they were with particular topics. Aspects of this task information are used during the development of the IRF algorithms. The log files that resulted from this process are used to create a document collection, the contents of which serve as stimuli for the IRF algorithms studied. Although all interaction was logged, we use only interaction with Web documents in this study. This allowed us to create a relatively homogeneous document collection that would not be biased by different interaction behaviors for different document types (e.g., a subject may interact differently when viewing a Web page than when viewing a word processing file on their local machine).

3.3 IRF Measures

The interaction logs allowed us to investigate the use of the *display time* (i.e., the amount of time that a document is active on the display) as an IRF measure to be interpreted by the algorithms. Display time was a measure for which we had much data, and has been one of the most studied IRF measures in the research literature [16]. The circumstances under which display time is a useful IRF measure are still uncertain, and for this reason it is important to study its application in more detail.

3.4 Document Collection

The collection used in this study contains 2741 Web documents. Fifteen percent of the collection (412 documents) was held out for use as a development collection for the IRF algorithms.[3]

[2] For clarity, in the remainder of this paper, we refer to the study during which the interaction logs were generated as the *longitudinal study*.

[3] We refer to this as a *development collection* rather than a *training collection* since the algorithms did not learn in any way. The development collection was used to inform design decisions made by the researchers about relevance criteria in IRF algorithms.

During the development process we used the document display times in these 412 documents and the relevance scores assigned to them by subjects to select threshold display times for the four algorithms. Since we were closely involved with these data it would not have been fair to test algorithm performance using these documents. For this reason we used the development collection only to derive these thresholds and debug the evaluation framework. The remaining 2329 unseen documents were used to test algorithm performance. This mimicked a situation where the algorithms were deployed in a real-world setting, and presented with unseen evidence. Since the study required documents for which display time and usefulness information was available, the collection was pruned from an initial size of 4868 to remove documents with no display time and no usefulness score perhaps signaling erroneous log entries. The textual content of the documents was indexed by the Terrier IR system[4], and made accessible to the IRF algorithms.

3.5 Tasks

During the longitudinal study questionnaires were used to elicit tasks that were of current interest, or that were expected to be of interest, to subjects during the study. Subjects were asked to think about their online information-seeking activities in terms of tasks, and to create personal labels for each task. They were provided with some example tasks such as "writing a research paper," "travel," and "shopping," but in no other way were they directed, influenced or biased in their choice of tasks. For the current study a generic classification was devised for all tasks identified by all subjects, producing the following nine *task groupings*:

1. Academic Research
2. News and Weather
3. Shopping and Selling
4. Hobbies and Personal Interests
5. Jobs/Career/Funding
6. Entertainment
7. Personal Communication
8. Teaching
9. Travel

For example, the task labels "viewing news," "read the news," and "check the weather" would be classified in Group 2: "News and Weather." All 2741 documents in the collection were hand-classified according to these nine groups prior to this study by one of the authors. This classification scheme is used to investigate the impact of task information on the effectiveness of the IRF algorithms.

For each of the seven subjects, and for each of the nine task groups (e.g., (Subject 1, Task 1), (Subject 1, Task 2), etc.), an initial "title" query was created from the top three most frequent terms in the union of the non-stopword terms in the task labels generated by that subject. This query length is representative of the length of queries submitted to Web search systems [11][5]. Using the union of terms lessened the variation caused by slight differences in the labels assigned by a subject within a particular task grouping. 46 *tasks* were generated using this method[6], and were stored in standard TREC-style format.

3.6 Ground Truth: Relevance Judgments

Subjects in the longitudinal study used an evaluation interface to identify tasks they were working on, classify documents that they viewed according to those tasks, and evaluate document usefulness. Subjects also indicated their confidence in the usefulness ratings that they assigned to documents. Usefulness values obtained from this interface were on seven-point scales. These ratings are used in our study as ground truth relevance judgments. These judgments were derived from data that does not simply reflect relevance assessments for standard IR search sessions but more free-form use of information. In general, the majority of documents were classified by subjects using the higher end of the relevance scale. Given that many researchers have found that selection or click-through is a good indicator of relevance [e.g., 12], the skewness is unsurprising considering that subjects had selected all documents which they evaluated.

We conducted an analysis of the level of kurtosis (i.e., how flat the distribution is) based on usefulness scores assigned by subjects to documents. The aim of doing this was to determine how best to collapse the usefulness data from a seven-point scale to a scale of less granularity, and hence more consistency between subjects. Kurtosis scores that are closer to 0 indicate flatter distributions (i.e., more equal numbers of cases in each group). The findings of this analysis revealed that using all seven points was not optimal, and that using binary schemes customized to subjects led to the flattest distributions. This resulted in three binary divisions. Table 1 shows how the scale was divided for each subject.

Table 1. Binary division of seven-point usefulness scale.

Subjects	User group	Ratings	
		Non-relevant	Relevant
1, 3, 5, 7	1	1, 2, 3, 4, 5	6, 7
2, 4	2	1, 2, 3, 4, 5, 6	7
6	3	1, 2, 3, 4	5, 6, 7

These user groupings were used by the algorithms tested in this study to determine whether a given document was relevant or non-relevant. For example, if an algorithm personalized to Subject 5 was presented with a document that had been assigned a usefulness rating of 4 by that user, then that document would be classed as non-relevant. This document would therefore be ignored for IRF, since only relevant documents are used.

Table 2 shows the number of non-relevant and relevant documents for each subject and overall. Although we refer to these two groups as non-relevant and relevant, it is important to note that the relevant group corresponds to strong relevance and the non-relevant group corresponds to weak and non-relevance.

[4] http://ir.dcs.gla.ac.uk/terrier

[5] As stated earlier, the collection we used comprised only Web documents.

[6] Although the maximum possible number of tasks was 63 (i.e., seven subjects over nine task groupings), documents for some subject-task pairings were not present in the collection used in this study.

Table 2. Non-relevant and relevant documents (per subject).

Subject	Non-relevant	Relevant	Total
1	191	110	301
2	100	313	413
3	93	192	285
4	166	59	225
5	25	47	72
6	344	615	959
7	192	294	486
Total	1111	1630	2741

Although the division does not result in an even distribution of relevant and non-relevant judgments for all subjects, this was the most consistent distribution that was obtainable from the data. As mentioned earlier, subjects generally rated more documents relevant than non-relevant, which is unsurprising since selection has been found to be a good relevance indicator [12].

3.7 Implicit Feedback Algorithms

The IRF algorithms selected query expansion terms from documents assumed relevant based on subject interaction. Relevance was determined based on whether viewing time equaled or exceeded a temporal threshold. Four algorithms were developed that used the following criteria to determine this threshold:

TaskAndUser: Separate threshold document display times for each subject-task pair.

TaskOnly: Separate threshold document display times for each task, across all subjects.

UserOnly: Separate threshold document display times for each subject, across all tasks.

All: A single threshold document display time across all subjects and all tasks. This algorithm is a baseline.

The presence of task and user information was varied in the development of these algorithms as shown in Table 3

Table 3. Task and user information in the four algorithms.

		Task Information	
		Present	Not present
User Information	Present	TaskAndUser	UserOnly
	Not present	TaskOnly	All

The performance of the four IRF algorithms was monitored for each document presented following the initial query submission. Documents were presented to the algorithms in the order in which they appeared in the interaction logs. This improved the realism of the experiments since feedback was provided to the algorithms in the order in which it would generally be given. All algorithms used the popular *wpq* method [19] to rank terms for query expansion. This method has been shown to be effective and produce good results. The equation for *wpq* is shown below, where the typical values r_t = the number of marked relevant documents containing the term t, n_t = the number of documents containing t, R = the number of marked relevant documents for query q, N = the total number of documents.

$$wpq_t = \log \frac{r_t/(R-r_t)}{(n_t-r_t)/(N-n_t-R+r_t)} \cdot \left(\frac{r_t}{R} - \frac{n_t-r_t}{N-R}\right) \tag{1}$$

The *wpq* method is based on the probabilistic distribution of terms in relevant and non-relevant documents. It was used to select the six expansion terms to be added to the original query. This was done without any prior knowledge of the effectiveness of adding this number of terms to queries for this collection. However, adding this number of terms has been shown to be effective in previous related work [10]. This method requires the presence of prior relevance judgments created by the binary classification of the usefulness scores described in Section 3.6.

Some algorithms were devised based on the presence of information about the search task, provided by subjects during the longitudinal study. In previous work with this set of interaction logs it was demonstrated that display time differed significantly according to task [14]. Algorithms were further categorized based on subjects' individual information-seeking behaviors (i.e., one individuated algorithm per subject), used a combination of task and user information, or remained general across all subjects (i.e., one generic algorithm for all subjects).

To identify potentially useful rules for classifying documents as relevant and non-relevant based on display time, we computed and evaluated a variety of statistics from the 412 documents in the development collection: mean, median, mode, quartiles, and standard deviation. Skewness measures showed that the data were not distributed normally making means unreliable indicators of relevant and non-relevant documents. Further analysis indicated that the median was the most consistent indicator of relevance. In all algorithms the median document display time was used as a relevance threshold value; documents viewed for that time or above were assumed to be relevant. Table 4 shows the threshold times.

Table 4. Threshold display times in seconds (per subject / task group).

Task group	Subject							All
	1	2	3	4	5	6	7	
1	23	20	25	34	**15**	3	34	8
2	41	43	**10**	7	**15**	9	74	23
3	10	8	**10**	16	**15**	1	35	6
4	9	23	9	7	**15**	3	39	12
5	**20**	23	**10**	4	5	3	61	4
6	21	**20**	27	9	**15**	2	32	23
7	**20**	**20**	**10**	7	**15**	**4**	30	8
8	23	**20**	**10**	**9**	**15**	**4**	36	29
9	**20**	**20**	**10**	**9**	29	4	25	9
All	20	20	10	9	15	4	49	11

We now describe features of the four IRF algorithms tested in this study. All algorithms used times shown in Table 4 in some way: *TaskAndUser* uses the *un-shaded cells*, *TaskOnly* uses the *shaded column*, *UserOnly* uses the *shaded row*, and *All* uses the shaded cell in the lower-right corner (i.e., 11 seconds).

The only difference between the four algorithms was the display time used in determining document relevance. Since the development collection contained only a small number of documents, there were instances where documents for some subject-task pairs were not represented in the development collection. In such cases we used the document display time for the subject (i.e., the values from the shaded row of Table 4). These instances are highlighted in bold in Table 4.

Although self-identified information about tasks may be difficult to obtain in operational environments, recent research has shown that users are willing to partition their computer activities according to task when provided with appropriate interface support [6]. Thus, in exploring task-based algorithms, we assume that task information is available during information interactions (i.e., we assume that users are self-identifying tasks as they search for information) and that this information would be available to the system.

3.8 Evaluation Measures

The evaluation measures adopted in this study are mean average precision (MAP) (i.e., the average of the precision value after each relevant document has been retrieved, across all tasks) and precision at the top-10 documents retrieved (P10) (i.e., the proportion of the top 10 documents that are relevant). These metrics are used commonly in IR evaluation and can provide good insight into the quality of the revised queries generated by the IRF algorithms. We would expect the MAP and P10 values to increase following the provision of more feedback.

3.9 Methodology

The MAP and P10 values for the four algorithms were computed across a series of feedback iterations using the framework described earlier. An iteration was defined as a document that met the relevance criteria (i.e., viewed for equal to or longer than the threshold display time specified by the algorithm). The following methodology was applied during this study:

1. Create initial set of queries from task labels.
2. For each algorithm, loop through the document set for each task and subject:
    a. If document display time equals or exceeds the pre-determined threshold for that algorithm, for the current task and subject:
        i. Pass the document to the algorithm and use it, and any previous seen relevant documents, to expand initial query.
        ii. Use expanded query to retrieve new set of documents using a best match *tf.idf* weighting scheme.
        iii. Use ground truth information to evaluate the documents retrieved, and score the current IRF algorithm.
3. IRF algorithms are ranked based on MAP and P10 averaged across all search tasks, users, and tasks to determine algorithm performance.

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

4. FINDINGS

Findings are presented for the 2329 documents present in the test collection, over iterations 1, 2, 5, 10, 15, and 20. Using these six milestones gives us insight into algorithm performance following different amounts of feedback[7]. It is possible that an algorithm may require numerous examples of relevant information before it can generate expanded queries that significantly improve retrieval performance. All new queries generated are expansions of the original set of queries where the MAP is .077 and mean P10 (MP10) is .111. Parametric statistical testing is used at a .05 level of significance where appropriate.

The document collection contained information spanning 46 topics. However, since only documents that met the relevance criteria specified by each of the algorithms were used for feedback, the number of topics for which queries were expanded is less than this value. As described earlier, these original queries were generated from the most frequently occurring task labels assigned by each subject for each of the nine task groupings. In Table 5 we show the number of tasks across which average values were computed (i.e., at each iteration the number of tasks for which there was a relevant document).

Table 5. Number of tasks used to compute average values.

Algorithm	Iteration					
	1	2	5	10	15	20
TaskAndUser	27	25	24	17	12	12
TaskOnly	27	25	22	17	17	14
UserOnly	27	25	23	18	13	11
All	27	27	24	21	16	12

The number of tasks falls as the number of iterations increases since there are few tasks with a large number of relevant documents. To get an understanding for what was really changing between feedback iterations, we computed the MAP and MP10 values across only tasks for which there was a relevant document (i.e., the number of tasks shown in Table 5). In Figures 2 and 3 we present the MAP and MP10 values across all search tasks for each of the four algorithms.

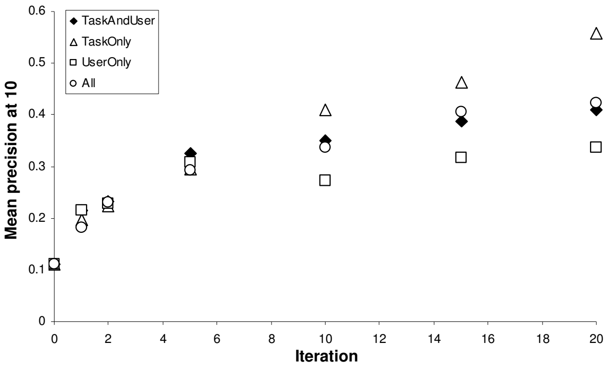

Figure 2. MP10 for all algorithms (per iteration).

[7] Extending the amount of feedback beyond the first 20 iterations affects the reliability of the statistical analysis, since few tasks have more than 20 relevant documents.

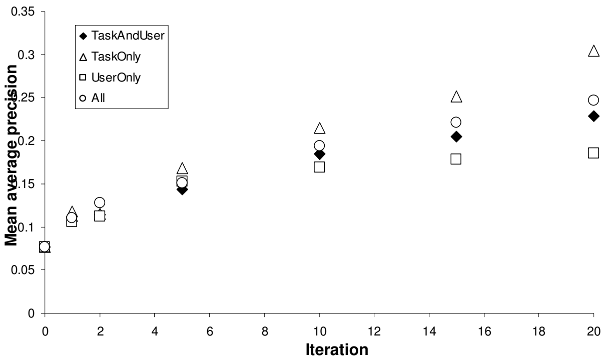

Figure 3. MAP for all algorithms (per iteration).

A modified residual collection method [21] was used to address feedback effects, where the presence of already-seen relevant documents in the scored result set positively skews measures of retrieval effectiveness such as precision. From the MAP and MP10 values obtained during this analysis it appears that *UserOnly* performs worse than any of the other algorithms, including the baseline algorithm (i.e., *All*), where task and user information are ignored. To reduce the emphasis on actual precision scores (which may be affected by experimental conditions and therefore difficult to generalize) we compute the percentage of MAP and MP10 values obtained from *TaskAndUser*, *TaskOnly*, and *UserOnly*, over the baseline (i.e., *All*) at each of the six iteration milestones. The percentage values obtained are shown in Table 6, rounded to the nearest whole point. The largest values for each measure are highlighted in bold.

Table 6. Percentage difference in MAP / MP10 over baseline.

Iter.	Algorithm					
	TaskAndUser		TaskOnly		UserOnly	
	MAP	MP10	MAP	MP10	MAP	MP10
1	−3	+6	**+4**	+6	−9	**+17**
2	**−10**	−1	−12	−5	−11	**0**
5	−7	**+10**	**+13**	−1	+3	+5
10	−8	+3	**+13**	**+35**	−13	−15
15	−7	**+5**	**+14**	**+15**	−18	−22
20	−8	−2	**+20**	**+44**	−26	−20

The form of personalization used (i.e., tailoring display time thresholds to individual subjects) seems to degrade retrieval performance. In contrast, using information about the search task (in *TaskOnly*) appears to enhance retrieval performance, especially in later iterations. The inclusion of user information in decisions about threshold display times in *TaskAndUser* may have harmed the performance of that algorithm.

Two-way factorial ANOVAs were applied to the data gathered at each of the feedback iterations depicted in Figures 2 and 3. The results of this analysis are summarized in Table 7 (the values representing significant differences are shown in bold). In this table the user variable, the task variable, and the interaction between these variables are represented by *U*, *T*, and *U*×*T* respectively. Also shown are the F-value (F), the within-groups and between-groups degrees of freedom (df), the probability (p) of the independent variables having an effect on the dependent variable (i.e., MAP or P10).

As Table 7 shows, the results of this analysis reveal significant differences in the precision of the results retrieved at later iterations (i.e., iterations 10, 15, and 20) between the four algorithms[8]. This can be attributable to the presence of task information and the lack of user information. That is, using information about the search task to tailor threshold display times during IRF algorithm development appears to enhance performance in later iterations, and tailoring display time thresholds based on users appears to worsen performance.

The MAP and P10 values were strongly correlated at all six iteration milestones (all Pearson's $r \geq .907$, all $p \leq .0001$)[9]. This implies that perhaps only one of these measures need be used in analysis such as this in the future.

Table 7. ANOVA values for MAP and P10 (per iteration).

Iter.	Var.	df		Measure			
		Num.	Denom.	MAP		P10	
				F	p	F	p
1	*U*	1	104	.21	.65	.23	.63
	T	1	104	.32	.57	.36	.55
	U×*T*	1	104	.02	.89	.01	.92
2	*U*	1	98	.33	.57	.45	.50
	T	1	98	.19	.66	.52	.47
	U×*T*	1	98	.06	.81	.01	.89
5	*U*	1	86	1.84	.18	2.81	.10
	T	1	86	1.88	.17	2.75	.10
	U×*T*	1	86	.01	.92	.03	.86
10	*U*	1	69	**4.19**	**.04**	**4.68**	**.03**
	T	1	69	**4.27**	**.04**	**4.47**	**.04**
	U×*T*	1	69	.02	.89	.08	.78
15	*U*	1	54	**5.01**	**.03**	**5.30**	**.03**
	T	1	54	**4.82**	**.03**	**5.42**	**.02**
	U×*T*	1	54	.01	.92	.01	.92
20	*U*	1	45	**6.55**	**.01**	**7.33**	**> .001**
	T	1	45	**6.87**	**.01**	**7.40**	**> .001**
	U×*T*	1	45	.02	.89	.01	.92

As an additional form of analysis we grouped MAP and P10 values by subject and task grouping. To test this, we combined the MAP and P10 values for all four IRF algorithms, and grouped the values by subject and by task. This allowed us to

[8] The poor initial performance of all algorithms can be attributed to lack of IRF they had received at that stage.

[9] The probability is derived from a test of the null hypothesis that the observed value comes from a population in which there is no correlation. More details on significance tests for Pearson's r can be found in [9].

determine whether there were any subjects or tasks for which the algorithms performed particularly well or particularly poorly. For each task grouping, and for each subject, the initial MAP and P10 values (i.e., the precision before any user interaction) were computed across all four algorithms. In Table 8 we show the percentage change in MAP and P10 from these initial values for each of the seven subjects, at each iteration milestone. For example, after the first feedback iteration, for Subject 1, the MAP jumped on average 191%. In Table 9 we show the same information, by task. Cells containing values that were significantly different from others in the same row are shown in bold. There were fewer than 20 relevant documents for Task Grouping 6 (with a total of six relevant documents) and Task Grouping 7 (with no relevant documents). This meant that we were unable to provide MAP and P10 scores across all iteration milestones for these two groupings. A hyphen is placed in the cells in Table 9 for which we have insufficient information to compute the average values.

Table 8. Percentage change in MAP and P10 (per subject).

Iter.	Meas.	Subject						
		1	2	3	4	5	6	7
1	MAP	191	142	**99**	**31**	283	**124**	326
	P10	186	151	**94**	**142**	388	**117**	269
2	MAP	149	157	**138**	**40**	430	**144**	332
	P10	105	147	**159**	**100**	200	**263**	338
5	MAP	248	170	**385**	**47**	470	243	302
	P10	198	171	**281**	**116**	264	447	413
10	MAP	271	174	**629**	**41**	375	260	368
	P10	249	138	**459**	**233**	270	263	272
15	MAP	273	198	**409**	**102**	427	280	445
	P10	270	167	**638**	**518**	270	331	375
20	MAP	309	247	512	299	427	285	513
	P10	310	200	488	500	450	394	366

Table 9. Percentage change in MAP and P10 (per task group).

Iter.	Meas.	Task group								
		1	2	3	4	5	6	7	8	9
1	MAP	98	220	219	**79**	213	203	–	251	157
	P10	96	271	200	**76**	229	267	–	300	175
2	MAP	135	131	219	**61**	183	142	–	302	141
	P10	121	208	328	**53**	296	167	–	300	200
5	MAP	176	202	462	**106**	277	210	–	356	192
	P10	152	333	445	**54**	358	417	–	300	275
10	MAP	235	241	487	223	316	–	–	362	168
	P10	145	306	453	372	500	–	–	397	400
15	MAP	260	238	592	267	299	–	–	258	153
	P10	181	167	588	275	583	–	–	312	400
20	MAP	271	362	574	378	278	–	–	530	153
	P10	169	500	475	206	417	–	–	319	400

The values shown in Tables 8 and 9 suggest that the performance of IRF algorithms exhibits a degree of variation between subjects, and perhaps less variation between tasks.

One-way independent measures ANOVAs were applied to the values separately across the seven subjects at each of the six iteration milestones. The results of this analysis indicated the presence of significant differences between the subjects at iterations 1, 2, 5, and 10 (all F(6, 54) ≥ 2.80, all p ≤ .019). Tukey post-hoc tests were applied to determine which pairs of subjects were significantly different. The results of this analysis suggested that the MAP and P10 values obtained by Subjects 3, 4, and 6 were significantly lower than the MAP and P10 values obtained by all other subjects at iterations 1 and 2 (all Z ≥ 2.33, all p ≤ .026), and that the MAP and P10 values obtained for Subjects 3 and 4 was also significantly lower than all other subjects at iterations 5, 10, and 15 (all Z ≥ 2.29, all p ≤ .029). All other differences between subjects were insignificant. These findings suggest that IRF algorithm performance can be affected by differences between subjects.

In a similar way, for the nine task groupings, we applied one-way independent measures ANOVA across all MAP and P10 values across all six feedback iteration milestones. The results of this analysis reveal significant differences in retrieval effectiveness between some of the task groupings in iterations 1, 2, and 5 (F(7,72) = 2.57, p = .02). Tukey post-hoc tests were applied to determine which pairs of subjects were significantly different. The findings of these tests revealed that the MAP and P10 values were significantly lower in iterations 1, 2, and 5, for Task Grouping 4 "Hobbies and Personal Interests" (all Z ≥ 2.45, all p ≤ .019). This task was by nature personal to subjects – perhaps more so than other tasks – and it may have been difficult to obtain the same degree of consistency in display times between this task grouping and other groupings. Nonetheless, the performance of the IRF algorithms across search tasks does appear to be more consistent than across users.

The findings presented in this section suggest that using task information to tailor display time thresholds in IRF algorithms leads to improved performance over a baseline algorithm that does not use such information, and alternative algorithms that personalize threshold display times to the user. We have also shown that there appears to be more variability in algorithm effectiveness between users than there is between task groupings. This suggests that using task groupings as the basis for tuning IRF algorithms may be one way to improve their reliability. These are important findings with ramifications for how IRF algorithms should be developed. In the next section we discuss the implications of these findings.

5. DISCUSSION AND IMPLICATIONS

In this study we sought to deepen our understanding of the role of mediating factors in IRF algorithm performance. Results suggest that tailoring display time thresholds to search tasks leads to improved performance over algorithms that do not use such information. The threshold display times for each of the nine task groupings were carefully selected by the researchers based on the display times of relevant documents in the development collection. Therefore, it is interesting, but not all that surprising that the algorithms that utilized task information were able to perform effectively.

The more surprising finding of this study was that tailoring display time thresholds to the individual user appeared to worsen retrieval performance, significantly so in feedback iterations of 10 documents and beyond (even compared to a baseline with no tailoring of display time thresholds). This may have been due to the measure we selected (perhaps display time is not a reliable indicator of relevance for each individual), or the way in which we derived threshold display times (perhaps using the median time is not the best approach for individual users, or perhaps an approach whereby the algorithms "learned" threshold display times may be more effective). As the findings showed, there was a lot of variability between subjects, with the personalized algorithms performing well for some and poorly for others. It may be that users interact in a more consistent way between users within a given task grouping than within each user across multiple task groupings[10], and that tailoring IRF support to the task being attempted is more prudent than trying to do so for each user. For example, in the longitudinal study, all subjects engaged in "Academic Research" tended to do view documents for a similar length of time (M=29.8s, SD=8.4s), but across all tasks the display time varied greatly[11]. Furthermore, as this data set showed, there are large variations in how much evidence is available to tailor algorithms to individuals; some subjects viewed many pages, while others viewed only a few.

Further analysis of the data was performed using the three subject groupings devised during the binary classification of the relevance scores. We felt this may yield more consistency than using findings for all seven subjects independently. Findings revealed a slight decrease in variance in display times, and only significant differences in retrieval effectiveness between the combination of all four algorithms on User Group 3 (comprising only Subject 6), and the combination of all four algorithms for the other two user groups ($F(2,104) = 4.44$, $p = .01$). This demonstrates that grouping users (in this case by their assessments of usefulness), and developing algorithms based on groups rather than individual users may be one way to improve the consistency of IRF algorithm performance.

As Dragunov, et al. [6] have shown, eliciting task information from users is possible given adequate interface support. Given that such task information also appears to lead to improved algorithm performance, developing IRF algorithms tailored to a set of pre-defined task groupings similar to those defined in this study, and getting the user to indicate their active group, may lead to significant improvements in retrieval effectiveness. Tasks groupings could be created by designers based on interaction logs, and offered to users at the interface as an additional (optional) form of information need specification. The challenge of course is how to offer this support in a lightweight way that will be easy to use by the broader user populace, who has become accustomed to minimal interaction with search systems.

In some respects building IRF algorithms for task groupings is more attractive than personalization; there are generally fewer task groupings than there are users, and inter-task variability in document display time (and perhaps other user behaviors) is not as severe as inter-user variability. Rather than relying on self-identified tasks (at the client-side), further research is also needed into automatically identifying tasks. Data that can be used to comprise these groupings is readily available in the interaction logs of popular search engines. Although work in this area has begun already with server-side data [20], and inference following user-provided training information [6], there may be scope to apply it to the development of task-dependent IRF algorithms. Although it is possible to personalize RF algorithms using the information stored on users' personal computers [23], our results suggest that when designing IRF algorithms, system designers should pay particular attention to the tasks users are likely to attempt.

The lack of consistency in document display times between users may have been related to the small number of users involved in this study. To further investigate the effect of personal differences in future experiments it may be necessary to recruit a larger subject pool, or use data gathered from other information sources such as search engine interaction logs.

Gathering data that is as rich in nature as that used in this study is a challenging and costly undertaking. Logs from Web search engines yield some insight into the interaction behaviors of many searchers, but provide only limited access to information about the task they are attempting or the users themselves. The use of these data to conduct this study presented a unique opportunity to study users' information-seeking behaviors over a period of time. Although the collection was limited in size, and we did not explore the use of data on task stage or topic familiarity, the research we conducted represents a step forward in understanding how to develop effective IRF algorithms, and in particular, what is the best way to handle implicit evidence as input to these algorithms.

A useful byproduct of this study was the framework developed to automate aspects of the experimental process. This can be used to encourage the rapid exploration of different algorithms that leverage IRF. The framework can potentially empower designers interested in exploring issues in algorithm development. In this study the algorithm was varied and one measure was used. A feature of the framework is that it allows one to vary the IRF measures used, to determine the most effective measure for a given algorithm. Other avenues such as measuring IRF algorithm effectiveness based on rate of learning [27] or query quality (rather than result quality) [25] are also attractive research directions. A limitation of this study was only one IRF measure was used to address our research questions. In future work we will use the framework to test for the presence of similar effects in other measures, and combinations of measures.

[10] The exception to this is the "Hobbies and Personal Interests" task, where there was less consistency in document display time between subjects performing that task. Although all tasks were in some way personal to the subject, this task appeared to be particularly so, leading to larger variations in document display times than the other tasks.

[11] Subject 1: M=13.6s SD=18.6s; Subject 2: M=16.9s SD=22.2s; Subject 3: M=15.3s SD=14.6s; Subject 4: M=15.3s SD=20.3s; Subject 5: M=14.5s SD=12.5s; Subject 6: M=6.5s, SD=7.2s; Subject 7: M=42.5s SD=27.7s.

6. CONCLUSIONS

In this paper we have presented a study of two factors that may influence the performance of IRF algorithms. We developed four algorithms and varied relevance threshold values for one IRF measure – document display time – based on the presence of information about users and their search tasks during algorithm development. To conduct this study we created an automated evaluation framework that incorporated a rich interaction model, that allowed us to compare IRF algorithm performance and control parts of the experimental process. Findings of the study show that tailoring display time thresholds to the search task leads to an increase in the performance of IRF algorithms. However, variations in subjects' interaction styles may have prevented us from doing so for personalized IRF algorithms; generally multiple subjects interacted more consistently within a single task grouping than one subject did within multiple groupings. This is an important finding, and suggests that IRF algorithms should be created for each task grouping rather than personalized for each user, at least when using display time in isolation from other behavioral data. Although further study is required, our findings suggest that the future of IRF may well lie in task-dependent algorithms.

7. REFERENCES

[1] Agichtein, E., Brill, E., and Dumais, S. (2006). Improving web search ranking by incorporating user behavior. In *Proceedings of 29th ACM SIGIR Conference*, pp. 3-10.

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

[3] Buckley, C., Salton, G., and Allan, J. (1994). The effect of adding relevance information in a relevance feedback environment. In *Proceedings of the 16th ACM SIGIR Conference*, pp. 292-300.

[4] Claypool, M., Le, P., Waseda, M., and Brown, D. (2001). Implicit interest indicators. In *Proceedings of the ACM IUI Conference*, pp. 33-40.

[5] Cleverdon, C.W., Mills, J., and Keen, M. (1966). *Factors determining the performance of indexing systems*. ASLIB Cranfield project, Cranfield.

[6] Dragunov, A. N., Dietterich, T. G., Johnsrude, K., McLaughlin, M., Li, L., and Herlocker, J. L. (2005). TaskTracker: A desktop environment to support multi-tasking knowledge workers. In *Proceedings of the ACM IUI Conference*, pp. 75-82.

[7] Efthimiadis, E. N. (2000). Interactive query expansion: A user-based evaluation in a relevance feedback environment. *Journal of the American Society for Information Science & Technology,* 51(11): 989-1003.

[8] Fox, S., Karnawat, K., Myland, M., Dumais, S.T., and White, T. (2005). Evaluating implicit measures to improve the search experience. *ACM TOIS*, 23(2): 147-168.

[9] Gravetter, F.J. and Wallnau, L.B. (2004). *Essentials of statistics for the behavioral sciences*. Wadsworth Publishing: New York.

[10] Harman, D. (1988). Towards interactive query expansion. In *Proceedings of 11th ACM SIGIR Conference*, pp. 321-331.

[11] Jansen, B.J., Spink, A., and Saracevic, T. (2000). Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing and Management*, 36(2): 207-227.

[12] Joachims, T., Granka, L., Pang, B., Hembrooke, H., and Gay, G. (2005). Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th ACM SIGIR Conference*, pp. 154-161.

[13] Kelly, D. (2004). *Understanding implicit feedback and document preference: A naturalistic user study.* Unpublished doctoral dissertation, Rutgers University.

[14] Kelly, D. and Belkin, N.J. (2004). Display time as implicit feedback: understanding task effects. In *Proceedings of the 27th ACM SIGIR Conference*, pp. 377-384.

[15] Kelly, D., Dollu, V. J., and Fu, X. (2005). The loquacious user: A document-independent source of terms for query expansion. In *Proceedings of the 28th ACM SIGIR Conference*, pp. 457-464.

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

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

[18] Pirolli, P., Card, S.K., and Van Der Wege, M.M. (2001). Visual information foraging in a focus + context visualization. In *Proceedings of the ACM SIGCHI Conference*, pp. 506-513.

[19] Robertson, S. (1990). On term selection for query expansion. *Journal of Documentation*, 46(4): 359-364.

[20] Rose, D. E. and Levinson, D. (2004). Understanding user goals in Web search. In *Proceedings of the 13th International World Wide Web Conference*, pp. 13-19.

[21] Salton, G. and Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4): 288-197.

[22] Taylor, R.S. (1968). Question negotiation and information seeking in libraries. *College and Research Libraries*, 29: 178-194.

[23] Teevan, J, Dumais, S.T., and Horvitz, E. (2005). Personalizing search via automated analysis of interests and activities. In *Proceedings of 28th ACM SIGIR Conference*, pp.449-456.

[24] White R.W. (2006). Using searcher simulations to redesign a polyrepresentative implicit feedback interface. *Information Processing and Management*, 48(5): 1185-1202.

[25] White, R.W. and Marchionini, G. (2006). Examining the effectiveness of real-time query expansion. *Information Processing and Management*, in press.

[26] White, R.W., Ruthven, I., and Jose, J.M. (2005). A study of factors affecting the utility of implicit relevance feedback. In *Proceedings of 28th ACM SIGIR Conference*, pp. 35-42.

[27] White, R.W., Ruthven, I., Jose, J.M., and Van Rijsbergen, C.J. (2005). Evaluating implicit feedback models using searcher simulations. *ACM TOIS*, 23(3): 325-361.