A Large-scale Evaluation and Analysis of Personalized Search Strategies

Zhicheng Dou *
Nankai University
Tianjin 300071, China
douzc@yahoo.com.cn

Ruihua Song
Microsoft Research Asia
Beijing 100080, China
rsong@microsoft.com

Ji-Rong Wen
Microsoft Research Asia
Beijing 100080, China
jrwen@microsoft.com

ABSTRACT

Although personalized search has been proposed for many years and many personalization strategies have been investigated, it is still unclear whether personalization is consistently effective on different queries for different users, and under different search contexts. In this paper, we study this problem and provide some preliminary conclusions. We present a large-scale evaluation framework for personalized search based on query logs, and then evaluate five personalized search strategies (including two click-based and three profile-based ones) using 12-day MSN query logs. By analyzing the results, we reveal that personalized search has significant improvement over common web search on some queries but it has little effect on other queries (e.g., queries with small click entropy). It even harms search accuracy under some situations. Furthermore, we show that straightforward click-based personalization strategies perform consistently and considerably well, while profile-based ones are unstable in our experiments. We also reveal that both long-term and short-term contexts are very important in improving search performance for profile-based personalized search strategies.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering, Search process*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*

General Terms

Algorithms, Experimentation, Human Factors, Theory

Keywords

Click-through, Personalization, Personalized Search, Query Log, Re-ranking

1. INTRODUCTION

One criticism of search engines is that when queries are issued, most return the same results to users. In fact, the vast majority of queries to search engines are short [27, 12] and ambiguous [16, 7], and different users may have completely different information needs and goals under the same query [12, 26, 23, 34]. For example, a biologist may use query "mouse" to get information about rodents, while programmers may use the same query to find information about computer peripherals. When such a query is submitted to a search engine, it takes a moment for a user to choose which information he/she wishes to get. On the query "free mp3 download", the users' selections can also vary though almost all of them are finding some websites to download free mp3: one may select the website "www.yourmp3.net", while another may prefer the website "www.seekasong.com".

Personalized search is considered a solution to this problem since different search results based on preferences of users are provided. Various personalization strategies including [21, 22, 26, 31, 14, 9, 35, 30, 19] have been proposed, and personalized web search systems have been developed, but they are far from optimal. One problem of current personalized search is that most proposed methods are uniformly applied to all users and queries. In fact, we think that queries should not be handled in the same manner because we find:

(1) Personalization may lack effectiveness on some queries, and there is no need for personalization on such queries. This has also been found by [34]. For example on the query "mouse" mentioned above, using personalization based on user interest profile, we could achieve greater relevance for individual users than common web search. Beyond all doubt, the personalization brings significant benefit to users in this case. Contrarily, for the query "Google", which is a typical navigational query as defined in [3, 17], almost all of the users are consistently selecting results to redirect to Google's homepage, and therefore none of the personalized strategies could provide significant benefits to users.

(2) Different strategies may have variant effects on different queries. For the query "free mp3 download", using the typical user interest profile-based personalization such as the method proposed in [6], which led to better results for the query "mouse", we may achieve poor results because the results for query "free mp3 download" are mostly classified into one topic category and the profile-based personalization is too coarse to filter out the desired results. In such a case, simply leveraging pages visited by this user in the past may achieve better performance. Furthermore, simply applying one personalization strategy on some queries without any consideration may harm user experience. For example, when a sports fan submits the query "office", he/she may

*Work was done when the author was visiting Microsoft Research Asia.

WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
ACM 978-1-59593-654-7/07/0005.

not be seeking information on sports, but may be seeking help on Microsoft Office Software or any other number of office-related inquiries. In this situation, if interest-based personalization is done, many irrelevant results could erroneously be moved to the front and the user may become confused.

(3) Personalization strategies may provide different effectiveness based on different search histories and under variant contexts. For example, it could be difficult to learn interests of users who have done few searches. Furthermore, as Shen et al. [25] noted, users often search for documents to satisfy short-term information needs, which may be inconsistent with general user interests. In such cases, long-term user profiles may be useless and short-term query context may be more useful.

In short, the effectiveness of a specific personalized search strategy may show great improvement over that of non-personalized search on some queries for some users, and under some search contexts, but it can also be unnecessary and even harmful to search under some situations. Until now, little investigation has been done on how personalization strategies perform under different situations. In this paper, we get some conclusions on this problem and make the following contributions:

(1) We develop a large-scale personalized search evaluation framework based on query logs. In this framework, different personalized re-ranking strategies are simulated and the search accuracy is approximately evaluated by real user clicks recorded in query logs automatically. The framework enables us to evaluate personalization on a large scale.

(2) We propose two click-based personalized search strategies and three profile-based personalized search strategies. We evaluate all five approaches in the evaluation framework using 12-day query logs from MSN search engine[1] and provide an in-depth analysis on the results.

(3) We reveal that personalization has different effectiveness on different queries, users, and search contexts. Personalization brings significant search accuracy improvements on the queries with large click entropy, and has little effect on the queries with small click entropy. Personalization strategies can even harm the search accuracy on some queries. Therefore, we conclude that not all queries should be personalized equally.

(4) We show that click-based personalization strategies perform consistently and considerably well though they can only work on the repeated queries. We find that our profile-based strategies are unstable because of the straightforward implementation. We also find the profile-based methods become more unstable when users search history grows. We reveal that both long-term and short-term contexts are very important in improving search performance for profile-based personalization.

The remaining sections are organized as follows. In Section 2, we discuss related works. We present a re-ranking framework and introduce how to use this framework to evaluate the personalization strategies in Section 3. In Section 4, we give several personalization strategies in both person and group levels. In Section 5 we introduce the dataset used in our experiments and detailed data statistics. We compare and analyze the results of these strategies in Section 6. We conclude our work in Section 7.

[1]MSN Search, http://search.msn.com

2. RELATED WORK

There are several prior attempts on personalizing web search. One approach is to ask users to specify general interests. The user interests are then used to filter search results by checking content similarity between returned web pages and user interests [22, 6]. For example, [6] used ODP[2] entries to implement personalized search based on user profiles corresponding to topic vectors from the ODP hierarchy. Unfortunately, studies have also shown that the vast majority of users are reluctant to provide any explicit feedback on search results and their interests [4]. Many later works on personalized web search focused on how to automatically learn user preferences without any user efforts [22, 19, 29, 26]. User profiles are built in the forms of user interest categories or term lists/vectors. In [19], user profiles were represented by a hierarchical category tree based on ODP and corresponding keywords associated with each category. User profiles were automatically learned from search history. In [29], user preferences were built as vectors of distinct terms and constructed by accumulating past preferences, including both long-term and short-term preferences. Tan et al. [31] used the methods of statistical language modeling to mine contextual information from long-term search history. In this paper, user profiles are represented as weighted topic categories, similar with those given in [28, 6, 22], and these profiles are also automatically learned from users' past clicked web pages.

Many personalized web search strategies based on hyperlink structure of web have also been investigated. Personalized PageRank, which is a modification of the global PageRank algorithm, was first proposed for personalized web search in [20]. In [10], multiple Personalized PageRank scores, one for each main topic of ODP, were used to enable "topic sensitive" web search. Jeh and Widom [14] gave an approach that could scale well with the size of hub vectors to realize personalized search based on Topic-Sensitive PageRank. The authors of [32] extended the well-known HITS algorithm by artificially increasing the authority and hub scores of the pages marked relevant by the user in previous searches. Most recently, [17] developed a method to automatically estimate user hidden interests based on Topic-Sensitive PageRank scores of the user's past clicked pages.

In most of above personalized search strategies, only the information provided by user himself/herself is used to create user profiles. These are also some strategies which incorporate the preferences of a group of users to accomplish personalized search. In these approaches, the search histories of users who have similar interest with test user are used to refine the search. Collaborative filtering is a typical group-based personalization method and has been used in personalized search in [29] and [30]. In [29], users' profiles can be constructed based on the modified collaborative filtering algorithm [15]. In [30], the authors proposed a novel method CubeSVD to apply personalized web search by analyzing the correlation among users, queries, and web pages contained in click-through data. In this paper, we also introduce a method which incorporates click histories of a group of users to personalize web search.

Some people have also found that personalization has variant effectiveness on different queries. For instance, Teevan et al. [34] suggested that not all queries should be handled

[2]Open Directory Project,http://dmoz.org/

in the same manner. For less ambiguous queries, current web search ranking might be sufficient and thus personalization is unnecessary. In [6] and [5], test queries were divided into three types: clear queries, semi-ambiguous queries, and ambiguous queries. The authors also concluded that personalization significantly increased output quality for ambiguous and semi-ambiguous queries, but for clear queries, one should prefer common web search. In [31], queries were divided into fresh queries and recurring queries. The authors found that recent history tended to be much more useful than remote history especially for fresh queries while the entire history was helpful for improving the search accuracy of recurring queries. This also gave us a sense that not all queries should be personalized in the same way. These conclusions inspired our detailed analysis.

3. EXPERIMENT METHODOLOGY

The typical evaluation method used in existing personalized search research is to conduct user studies [23, 26, 6, 34, 28, 29, 19, 5, 31]. Usually, a certain number of people participate in the evaluated personalized search system over several days. The user profiles are manually specified by participants themselves [6] or automatically learned from search histories. To evaluate the performance of personalized search, each participant is required to issue a certain number of test queries and determine whether each result is relevant. The advantage of this approach is that the relevance can be explicitly specified by participants. Unfortunately, there are still some drawbacks in this method. The constraint of the number of participants and test queries may bias the accuracy and reliability of the evaluation.

We propose a framework that enables large-scale evaluation of personalized search. In this framework, we use click-through data recorded in query logs to simulate user experience in web search. In general, when a user issues a query, he/she usually checks the documents in the result list from top to bottom. He/she clicks one or more documents which look more relevant to him/her, and skip the documents which he/she is not interested in. If a specific personalization method can re-rank the "relevant" documents fronter in the result list, the user will be more satisfied with the search. Therefore, we utilize clicking decisions as relevance judgments to evaluate the search accuracy. Since click-through data can be collected at low cost, it is possible to do large-scale evaluation using this framework. Furthermore, since click-through data reflect real world distributions of query, user, and user selections, they could be more accurate to evaluate personalized search using click-through data than user surveys.

One potential concern about the evaluation framework is that the original user selections may be influenced by initial result rankings[2], and thus it could be unfair to evaluate a reordering of the original search results using the original click data. Our framework may fail to evaluate the ranking alternation of documents that are relevant but were not clicked by users, and this may bias the evaluation. However, our framework is still effective to evaluate approximate search accuracy. It is the best method we could adopt to enable large-scale evaluation of personalized search. We will investigate more stable methodology in future work.

In the evaluation framework, we use MSN query logs to simulate and evaluate the personalized re-ranking. In MSN query logs, each *user* is identified by "Cookie GUID", which remains the same in a machine as long as a cookie is not cleared. For each *query*, MSN search engine logs the query terms and records all click-through information including clicked web pages and their ranks. A "Browser GUID", which remains the same before the browser is re-open, is also recorded for each query. It is used as the simple identifier of a *session*, which includes a series of queries made by a single user within a small range of time and is usually meant to capture a single user's attempt to fulfill a single information need [27, 12].

3.1 Re-ranking Evaluation Framework

In this evaluation framework, we first download search results from MSN search engine, then use one personalization strategy to re-rank the results. The click-through data recorded in test set is then used to evaluate the re-ranking performance. In more detail, query re-ranking and evaluation are completed in the following steps:

(1) Download the top 50 search results from MSN search engine for the test query. We denote the downloaded web pages with U and denote the rank list that contains the rankings of the web pages with τ_1.

(2) Compute a personalized score for each web page $x_i \in U$ using personalization algorithm and then generate a new rank list τ_2 with respect to U sorted by descending personalized scores. The personalized strategies are introduced in Section 4.

(3) Combine the rankings in τ_1 and τ_2 using Borda' ranking fusion method [13, 8] and sort the web pages with the combined rankings. The final rank list is denoted with τ. τ is the final personalized search result list for the query. Notice that we use the rank-based ranking fusion method because we are unable to get the relevance scores from MSN search engine.

(4) Use the measurements introduced in Section 3.2 to evaluate the personalization performance on τ.

In this paper, we assume the results downloaded from MSN search engine are consistent with those returned to the user when the query was submitted. We use the most recent MSN query logs on August 2006 and download search results in the early days on September 2006 so that the changes of search results can be ignored (We also tried the approach of rebuilding the search results from query logs but it failed because of the sparseness of queries and user clicks). We downloaded only the top 50 search results because most users never look beyond the top 50 entries in the test set.

3.2 Evaluation Measurements

We use two measurements to evaluate the personalized search accuracy of different strategies: rank scoring metric introduced in [30, 15] and average rank metric introduced in [23].

3.2.1 Rank Scoring

Rank scoring metric proposed by Breese [15] is used to evaluate the effectiveness of the collaborative filtering systems which return an ordered list of recommended items. Sun et al.[30] used it to evaluate the personalized web search accuracy and we also use it in this paper.

The expected utility of a ranked list of web pages is defined as

$$R_s = \sum_j \frac{\delta(s,j)}{2^{(j-1)/(\alpha-1)}}$$

where j is the rank of a page in the list, $\delta(s,j)$ is 1 if page j is clicked in the test query s and 0 otherwise, and α is set to 5 as the authors did. The final rank scoring reflects the utilities of all test queries:

$$R = 100\frac{\sum_s R_s}{\sum_s R_s^{Max}} \quad (1)$$

Here, R_s^{Max} is the obtained maximum possible utility when all pages which have been clicked appear at the top of the ranked list. Larger rank scoring value indicates better performance of personalized search.

3.2.2 Average Rank

Average rank metric is used to measure the quality of personalized search in [23, 28]. The average rank of a query s is defined as below.

$$AvgRank_s = \frac{1}{|\mathcal{P}_s|}\sum_{p\in\mathcal{P}_s} R(p)$$

Here $\mathcal{P}_s$ denotes the set of **clicked** web pages on test query s, $R(p)$ denotes the rank of page p. The final average rank on test query set $\mathcal{S}$ is computed as:

$$AvgRank = \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} AvgRank_s \quad (2)$$

Smaller average rank value indicates better placements of relevant result, or better result quality.

In fact, rank scoring metric and average rank metric has similar effectiveness on evaluating personalization performance, and our experimental results show that they are consistent.

4. PERSONALIZATION STRATEGIES

As we described in Section 2, personalized search methods can be categorized into person level and group level. In this paper, we propose several re-ranking methods in both levels to accomplish personalized search. These strategies are used to re-rank search results by computing a personalized score $S(q,p,u)$ for each page p in the results returned to user u on query q, as Section 3 introduced. In the following sections, we will introduce the strategies.

4.1 Person-level Re-ranking

4.1.1 Person-level Re-ranking Based on Historical Clicks

We suppose that for a query q submitted by a user u, the web pages frequently clicked by u in the past are more relevant to u than those seldom clicked by u, and thus, the personalized score on page p can be computed by:

$$S^{P\text{-}Click}(q,p,u) = \frac{|Clicks(q,p,u)|}{|Clicks(q,\bullet,u)|+\beta} \quad (3)$$

Here, $|Clicks(q,p,u)|$ is the click number on web page p by user u on query q in the past, $|Clicks(q,\bullet,u)|$ is the total click number on query q by u, and β is a smoothing factor ($\beta = 0.5$ in this paper). Notice that $|Clicks(q,p,u)|$ actually decides the ranking of the page, while $|Clicks(q,\bullet,u)|$ and β are only used for normalization.

A disadvantage to this approach is that the re-ranking will fail when the user has never asked this query. We find that in our dataset, about one-third of the test queries are repeated by the same user and this approach will only bring benefits to these queries.

This approach is denoted with **P-Click**.

4.1.2 Person-level Re-ranking Based on User Interests

As introduced in Section 2, many current researches use interest profiles to personalize search results [22, 19, 6]. In this paper, we also proposed a personalization method based on user interest profile (we denote this method with **L-Profile**). User's profile $\boldsymbol{c}_l(u)$ is presented as a weighting vector of 67 pre-defined topic categories provided by KDD Cup-2005 [18]. When a user submits a query, each of the returned web pages is also mapped to a weighting category vector. The similarity between the user profile vector and page category vector is then used to re-rank search results:

$$S^{L\text{-}Profile}(q,p,u) = \frac{\boldsymbol{c}_l(u)\cdot\boldsymbol{c}(p)}{\|\boldsymbol{c}_l(u)\|\,\|\boldsymbol{c}(p)\|} \quad (4)$$

Here $\boldsymbol{c}(p)$ is category vector of web page p. $\boldsymbol{c}(p)$ is generated by a tool using the query and web page classification method introduced in [24]. Given a web page p, the tool returns top 6 categories which p belongs to with corresponding confidences. Each component $c(p)_i$ of $\boldsymbol{c}(p)$ is the classification confidence returned by the tool, which means the probability that page p should be classified into category i. If category i is not in the 6 categories returned by the tool, then we set $c(p)_i = 0$. User's profile $\boldsymbol{c}_l(u)$ is automatically learned from his/her past clicked web pages as the following equation:

$$\boldsymbol{c}_l(u) = \sum_{p\in\mathcal{P}(u)} P(p|u)w(p)\boldsymbol{c}(p)$$

Here $\mathcal{P}(u)$ is the collection of web pages visited by user u in the past. $P(p|u)$ can be thought of as the probability that user u clicks web page p, i.e.,

$$P(p|u) = \frac{|Clicks(\bullet,p,u)|}{|Clicks(\bullet,\bullet,u)|}$$

Here, $|Clicks(\bullet,\bullet,u)|$ is the total click times made by u and$|Clicks(\bullet,p,u)|$ is the click times on web page p made by u. $w(p)$ is the impact weight for page p when generating user profiles. We assume that the web pages submitted by many users are less important when building user profiles, thus,

$$w(p) = \log\frac{|\mathcal{U}|}{|\mathcal{U}(p)|}$$

$|\mathcal{U}|$ is the number of total users; $|\mathcal{U}(p)|$ is the number of users who have ever visited web page p.

In method L-Profile, user's profile $\boldsymbol{c}_l(u)$ is accumulated from user's visited web pages in the past. This profile is called long-term profile in previous works [29, 26, 31]. In fact, as investigated by [26], short-term user profile is more useful for improving search in current session. In this paper, we use the clicks on the previous queries in current session to build user's short-term profile. A user's short-term profile $\boldsymbol{c}_s(u)$ is computed as below.

$$\boldsymbol{c}_s(u) = \frac{1}{|\mathcal{P}_s(q)|}\sum_{p\in\mathcal{P}_s(q)}\boldsymbol{c}(p)$$

$\mathcal{P}_s(q)$ is the collection of visited pages on previous queries in current session. The personalized score of page p using

short-term profile is computed as the following equation:

$$S^{S\text{-}Profile}(q,p,u) = \frac{\boldsymbol{c}_s(u) \cdot \boldsymbol{c}(p)}{\|\boldsymbol{c}_s(u)\| \, \|\boldsymbol{c}(p)\|} \quad (5)$$

This approach is denoted with **S-Profile**.

We can also fuse the long-term personalized score and the short-term personalized score using a simple linear combination:

$$S^{LS\text{-}Profile}(q,p,u) = \theta S^{L\text{-}Profile}(q,p,u) + (1-\theta) S^{S\text{-}Profile}(q,p,u) \quad (6)$$

We denote this approach with **LS-Profile**. Methods L-Profile, S-Profile, and LS-Profile are generally called *profile-based* methods for short in this paper.

4.2 Group-level Re-ranking

We use the K-Nearest Neighbor Collaborative Filtering algorithm to test group-based personalization. Due to the data sparsity in our dataset, using traditional CF methods on web search is inadequate. Instead, we compute the user similarity based on long-term user profiles:

$$Sim(u_1,u_2) = \frac{\boldsymbol{c}_l(u_1) \cdot \boldsymbol{c}_l(u_2)}{\|\boldsymbol{c}_l(u_1)\| \, \|\boldsymbol{c}_l(u_2)\|}$$

The K-Nearest neighbors are obtained based on the user similarity computed as follows.

$$\mathcal{S}_u(u_a) = \{u_s | rank(Sim(u_a,u_s)) \leq K\}$$

Then we use the historical clicks made by similar users to re-rank the search results:

$$S^{G\text{-}Click}(q,p,u) = \frac{\sum_{u_s \in \mathcal{S}_u(u)} Sim(u_s,u) \, |Clicks(q,p,u_s)|}{\beta + \sum_{u_s \in \mathcal{S}_u(u)} |Clicks(q,\bullet,u_s)|} \quad (7)$$

We denote this approach with **G-Click**.

5. DATASET

In this section, we introduce the dataset used in our experiments.

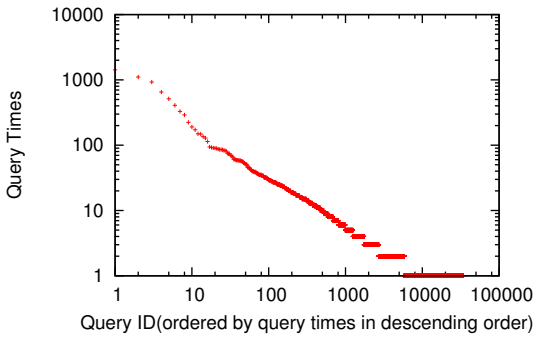
5.1 Statistics about Dataset

We collect a set of MSN query logs for 12 days in August 2006 for our experiments. Because the entire log set is too large, we randomly sample 10,000 distinct users (identified by "Cookie GUID") from the users in the United States on August 19, 2006. These users and their click-through logs are extracted as our dataset. In addition, the queries without any clicks (about 34.6% of all queries) are excluded from the dataset because they are useless in our experiments. The entire dataset is split into two parts: a training set and a test set. The training set contains the log data of the first 11 days and the log data of the last day is used for testing. Table 1 summarizes the statistics. Notice that all 10,000 users have search activities in the training set because users are sampled from the logs of training days.

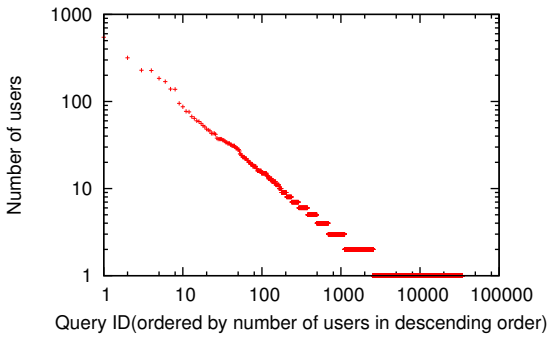
Because users are randomly sampled, this dataset could reflect the characteristics of the entire logs. It also has similar characteristics of those in existing reports [27, 37, 11, 36, 1]. We show detailed statistics of the dataset in the following sections.

Table 1: Basic statistics of dataset

Item	ALL	Training	Test
#days	12	11	1
#users	10,000	10,000	1,792
#queries	55,937	51,334	4,639
#distinct queries	34,203	31,777	3,465
#Clicks	93,566	85,642	7,924
#Clicks/#queries	1.6727	1.6683	1.7081
#sessions	49,839	45,981	3,865



(a) Distribution of query frequency (by log scale).



(b) Distribution of user number of queries (by log scale).

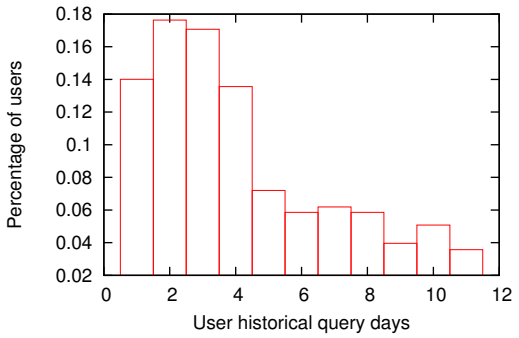
Figure 1: Query popularity distributions.

5.2 Statistics about Queries

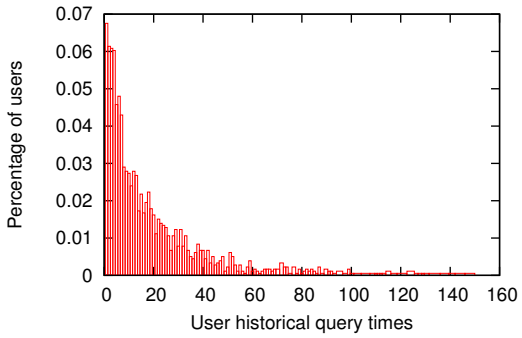
In our dataset, more than 80% distinct queries are only issued once in a 12-day period, and about 90% distinct queries string are issued only by one user. The 3% most popular distinct queries are issued by more than 47% users. The statistics is similar with that given in [27, 37, 11], and this indicates that information needs on the Web are quite diverse. Furthermore, we find that query frequency can also be characterized by Zipf distributions, consistent with that found by [37]. Figure 1(a) plots the distributions of query frequency. In this figure, queries are sorted by query times in descending order: the first query is the most popular one, and the last is the most unpopular one. Figure 1(b) plots the distribution of number of users on each query.

5.3 Statistics about Test Users

As Table 1 shows, 1,792 users have search activities on the test day. Figure 2 plots the distribution of historical (i.e. in



(a) Distribution of user historical query days



(b) Distribution of user historical query times

Figure 2: Distributions of user search frequency in training days for test users

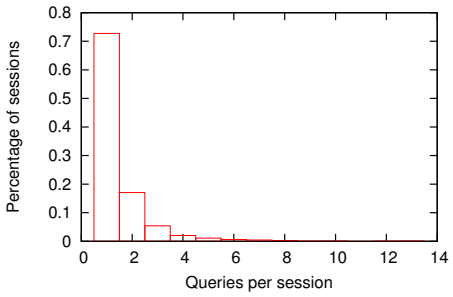


Figure 3: Distribution of query number per session.

training days) query day and query times for users in the test set. Because users are sampled on one of the training days, each user has at least a day-long query history. We find about 30% users in the test set have more than 5 days' query history and about 50 % of them submit more than 10 queries in training days.

5.4 Statistics about Query Repetition

In our dataset, 2,130 (about 46%) of all 4,639 queries in the test set are repeated ones that have been submitted in the training days, either by the same user or by different users. Furthermore, 1,535 queries (72% of repeated ones and 33% of test queries) are repeated by the same user. These results are consistent with those given in [33] and are helpful for personalized search.

5.5 Statistics about Sessions

In our dataset, we use "Browser GUID" as a simple identifier of session. A user opens a browser and asks one or more queries and then closes the browser: the whole process is considered as a session in this paper. Figure 3 shows the distribution of number of queries in a session. About 30% sessions contain at least two queries. This indicates that users sometimes submit several queries to fulfill an information need.

5.6 Distribution of Query Click Entropies

As found by [34], for queries which showed less variation among individuals, the personalization may be insufficient. In this paper, we define click entropy of query as Equation 8.

$$ClickEntroy(q) = \sum_{p \in \mathcal{P}(q)} -P(p|q) \log_2 P(p|q) \qquad (8)$$

Here $ClickEntroy(q)$ is the click entropy of query q. $\mathcal{P}(q)$ is the collection of web pages clicked on query q. $P(p|q)$ is the percentage of the clicks on web page p among all the clicks on q, i.e.,

$$P(p|q) = \frac{|Clicks(q, p, \bullet)|}{|Clicks(q, \bullet, \bullet)|}$$

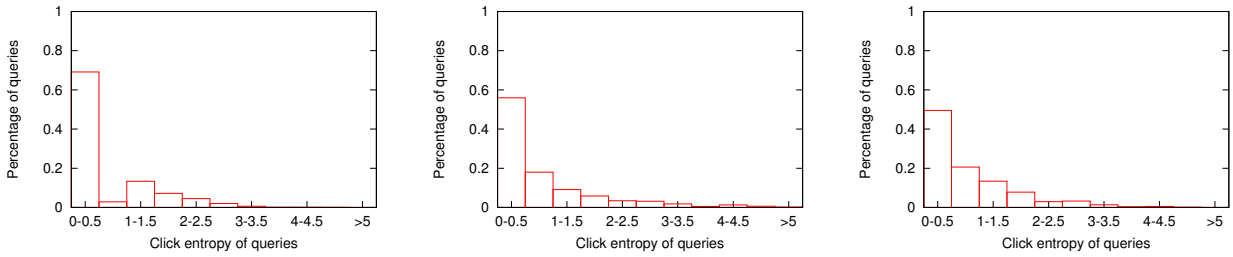
Click entropy is a direct indication of query click variation. If all users click only one same page on query q, then we have $ClickEntroy(q) = 0$. Smaller click entropy means that the majorities of users agree with each other on a small number of web pages. In such cases, there is no need to do personalization. Large click entropy indicates that many web pages were clicked for the query. This may mean: (1) a user has to select several pages to satisfy this query, which means the query is an informational query [3, 17]. Personalization can help to filter the pages that are more relevant to users by making use of historical selections. (2) Different users have different selections on this query, which means the query is an ambiguous query. In such cases, personalization can be used to provide different web pages to each individual.

Figure 4(a) shows the click entropy distribution. More than 65% queries have low click entropy between 0 and 0.5. We find many of these queries are submitted only by one user and the user only clicks one web page. Figure 4(b) shows the click entropy distribution for queries asked more than five times. Figure 4(c) plots the click entropy distribution for queries submitted by at least three users. From these figures, we also find that the majorities of the more popular queries have low click entropies.

6. RESULTS AND DISCUSSIONS

In this section, we will give detailed evaluation and analysis of the five personalized search strategies. Notice that the original web search method without any personalization, which is used for comparing with the personalized methods, is denoted with "WEB". We let $K = 50$ for method G-Click and $\theta = 0.3$ for method LS-Profile, and both of the two settings are empirical.

In our experiments, we find 676 queries in total 4,639 test



(a) All queries (b) Queries with query times>5 (c) Queries with user number>2

Figure 4: Distribution of query click entropy.

queries lost the clicked web pages in downloaded search results. This is because MSN search engine has changed for these queries. We excluded these queries when reporting the experimental results in the following sections. Furthermore, we find for 57% (2,256/3,963) of the left queries, users select only the top results. In other words, original search method WEB has done the best on these queries and personalization does not provide improvements. We call the other 1,707 queries, on which users select not only the top results, *not-optimal queries*.

6.1 Overall Performance of Strategies

Table 2 shows the overall effectiveness of the personalization strategies on the test queries. We find:

(1) Click-based personalization methods G-Click and P-Click outperform method WEB on the whole. For instance, on the not-optimal queries, method P-Click has a significant ($p < 0.01$) 3.68% improvement over method WEB and method G-Click have a significant ($p < 0.01$) 3.62% improvement over WEB (using rank scoring metric). P-Click and G-Click methods also have significant improvements (1.39% and 1.37%) over WEB on all test queries including both not-optimal and optimal queries. These results show that click-based personalization methods can generally improve web search performance.

(2) Methods P-Click and G-Click have no significant different performances on the whole. In our experiments, we sample 10,000 users and select the 50 most similar users for each test user in G-Click approach (we also try the methods to select 20 and 100 users, but they show no significant difference). By reason of high user query sparsity, selected similar users may have few search histories on the queries submitted by test user. This makes group-level personalization perform no significant improvement over person-level personalization. If more days' logs are given and more users are selected, method G-Click may perform better.

(3) Profile-based methods L-Profile, S-Profile, and LS-Profile perform less well on average. We compute rank scorings of all the methods for each single test query and then plot the distributions of rank scoring increment over WEB method for each personalization strategy in Figure 5. We find that though L-Profile, S-Profile, and LS-Profile methods improve the search accuracy on many queries, they also harm the performance on more queries, which makes them perform worse on average. This indicates that the straightforward implementation of profile-based strategies we employ in this paper do not work well, at least not as stable as the click-based ones. We will give some analysis on why our profile-based methods are unstable in Section 6.5.

Table 2: Overall performance of personalization strategies. R.S. denotes the rank scoring metric and A.R. denotes the average rank metric.

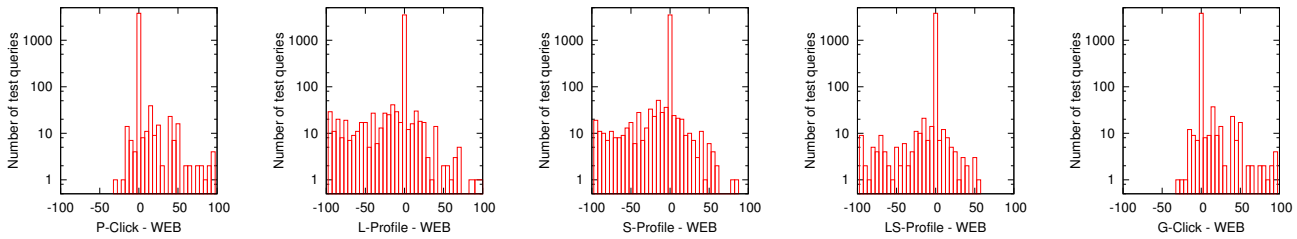
method	all		not-optimal	
	R.S.	A.R.	R.S.	A.R.
WEB	69.4669	3.9240	47.2623	7.7879
P-Click	**70.4350**	**3.7338**	**49.0051**	**7.3380**
L-Profile	66.7378	4.5466	45.8485	8.3861
S-Profile	66.7822	4.4244	45.1679	8.3222
LS-Profile	68.5958	4.1322	46.6518	8.0445
G-Click	70.4168	3.7361	48.9728	7.3433

6.2 Performance on Different Click Entropies

We give the average search accuracy improvements of different personalization strategies on the test queries with different click entropy in Figure 6. We use only the queries asked by at least three users to make the click entropy more reliable.

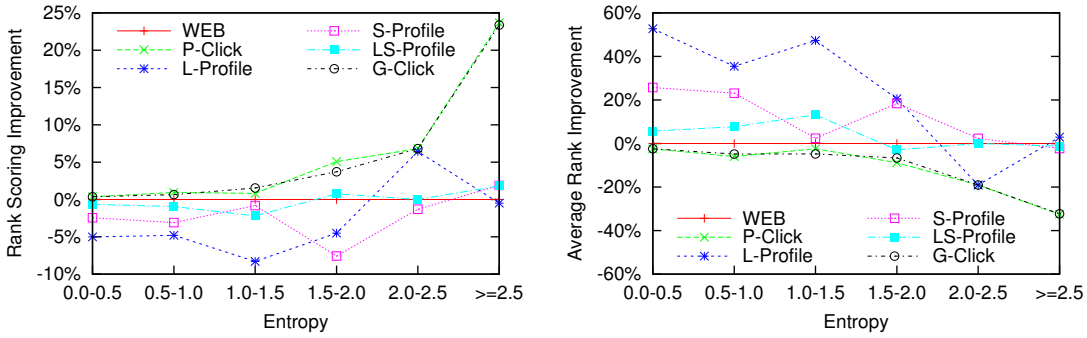
We see that the improvement of the personalized search performance increases when the click entropy of query becomes larger, especially when click entropy ≥ 1.5. For the click-based methods P-Click and G-Click, the improvement of personalization is very limited on the queries with click entropies between 0 and 0.5. The G-Click method, which gets the best performance for these queries, has only a non-significant 0.37% improvement over WEB methods in rank scoring metric. This means users have small variance on these queries, and the search engine has done well for these queries, while on the queries with click entropy$\geq$2.5, the result is disparate: both P-Click and G-Click methods make exciting performance. In the rank scoring metric, method G-Click has a significant ($p < 0.01$) 23.37% improvement over method WEB and P-Click method have a significant ($p < 0.01$) 23.68% improvement over method WEB. Profile-based methods L-Profile, S-Profile and LS-Profile worsen search performance when click entropy < 1.5, while L-Profile and LS-Profile also achieve better performances on queries with click entropy ≥ 1.5 (we wonder why method L-Profile also worsens search accuracy when click entropy$\geq$2.5 and will provide additional analysis on this in future work).

All these results indicate that on the queries with small click entropy (which means that these queries are less ambiguous or more navigational), the personalization is insufficient and thus personalization is unnecessary.



(a) P-Click (b) L-Profile (c) S-Profile (d) LS-Profile (e) G-Click

Figure 5: Distributions of rank scoring increment over WEB method. The count of the test queries with the same rank scoring increment range is plot in y-axis with log scale.



(a) Ranking scoring (b) Average rank

Figure 6: Search accuracy improvements over WEB method on the queries with variant click entropies. Only the queries asked by at least three users are included. Notice that in figure (b), smaller average rank means higher search accuracy.

6.3 Performance on Repeated queries

In Subsection 5.4, we find about 46% test queries are repeated by the same user or different users and 33% queries are repeated by the same user. It means that users often review the queries and results they ever referred. Teevan et al. [33] have also observed that re-finding behavior is common, and have shown that repeat clicks can often be predicted based on a user's previous queries and clicks. In this paper, methods P-Click and G-Click are based on historical clicks. The high repetition ratio in real world makes these click-based personalization strategies work well.

Table 3(a) shows the personalization performance on the repeated non-optimal queries repeated by either the same user or different users and Table 3(b) gives the results on the queries repeated by the same user. We find the personalization methods P-Click and G-Click have significant improvements over WEB method on queries repeated by same user. This means that when a user re-submit a query, his/her selections are also highly consistent with the past and the personalization based on his/her past clicks performs well. These results tell us that we should record user query and click histories and use them to improve future search if no privacy problems exist. We also should provide convenient ways for users to review their search histories, just like those provided by some current search engines.

6.4 Performance on Variant Search Histories

Do users who frequently use search engine benefit more from personalized search? Do profile-based personalized search strategies perform better when the search history grows? To answer these questions, we plot the improvements of rank scorings on queries given by users with different search frequencies in Figure 7. We find:

(1) Using click-based methods P-Click and G-Click, users who have greater search activities in training days do not consistently benefit more than users who do less searching. This is because users who frequently use the search engine may have more varied information needs. They may repeat old queries, but they may also submit lots of fresh queries, which makes our click-based methods P-Click and G-Click perform similar averages for users with different search frequencies (notice that the two series of methods P-Click and G-Click are very close to each other).

(2) Method L-Profile when using a user's long-term interest profile can perform better when a user has more queries, especially when the number of queries grows from 0 to 70. This is because we can catch users' long-term interests more accurately when their search histories are long enough. At the same time, we find that the performance of L-Profile becomes more unstable when the user has more and more queries, especially when they have more than 80 queries. This is because there is more noise in queries and furthermore the users have varied information needs. This tell us that when the user's search histories increase, we should take more analysis on user's real information need and select only appropriate search histories to build up user profiles. Tan et al. [31] find that the best performance of profile-based personalized search methods they proposed is achieved when

Table 3: Performance on repeated queries. In Table(a), Y means that the query is repeated by either the same user or different users and N means not. In Table(b), Y means that the query is repeated by the same user and N means that the query is first submitted by a user. All the queries are not-optimal queries.

(a) Performance on repeated queries

method	Y		N	
	R.S.	A.R.	R.S.	A.R.
WEB	46.6285	8.0620	47.4013	7.7002
P-Click	**55.9090**	**6.1663**	47.4013	7.7002
L-Profile	47.7405	8.2953	45.4091	8.4141
S-Profile	46.7600	8.0695	44.7980	8.4003
LS-Profile	46.8138	8.1340	46.6142	8.0169
G-Click	55.7377	6.1886	47.4013	7.7002

(b) Performance on user-repeated queries

method	Y		N	
	R.S.	A.R.	R.S.	A.R.
WEB	45.7215	8.0522	47.4858	7.7387
P-Click	**59.4750**	**5.2090**	47.4858	7.7387
L-Profile	48.0128	8.2575	45.5346	8.4100
S-Profile	45.5959	8.1306	45.1058	8.3579
LS-Profile	45.8936	8.1679	46.7618	8.0215
G-Click	59.1086	5.2500	47.5025	7.7332

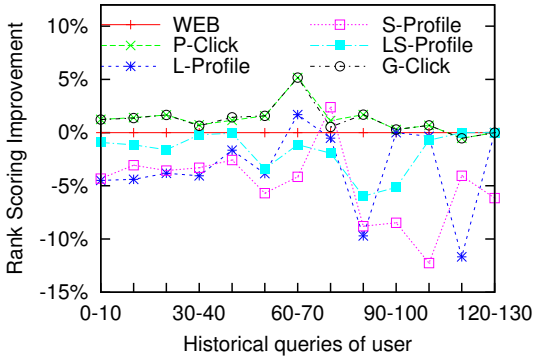


Figure 7: Rank scoring increments over WEB method on all test queries submitted by users with different query frequencies.

using click-through data of past searches that are related to the current query. We think this is because of the same reasons.

(3) Methods S-Profile and LS-Profile are less sensitive to historical query number. Method LS-Profile is more stable than method L-Profile.

6.5 Analysis on Profile-based Strategies

From Table 2, we surprisingly find that the profile-based personalization strategies perform less optimally, which is inconsistent with existing investigations [28]. We think this is due to the rough implementation of our strategies. The experimental results indicate that the straightforward implementation we employ does not work well. From Subsection 6.4 we see it is difficult to build an appropriate user profile even when the user history is rich. The search history inevitably involves a lot of noisy information that is irrelevant to the current search and such noise can harm the performance of personalization, as indicated by [31]. In our experiments we simply use all the historical user searches to learn user profiles without distinguishing between relevant and irrelevant parts, which may make the personalization unstable. We also do none normalization and smoothing when generating user profiles. Since these strategies are far from optimal, we will do more investigation and try to improve their performance in future work.

Although profile-based approaches perform badly in our experiments, we can still find an interesting thing. Method LS-Profile is more stable than methods L-Profile and S-Profile, as shown in Table 2, Figure 5, Figure 6 and Figure 7. That means the incorporation of long-term interest and short-term context can gain better performance than solely using either of them. In other words, both long-term and short-term search contexts are very important to personalize search results. The combination of the two type of search context can make the prediction of real user information need more reliable.

7. CONCLUSIONS

In this paper, we try to investigate whether personalization is consistently effective under different situations. We develop a evaluation framework based on query logs to enable large-scale evaluation of personalized search. We use 12 days of MSN query logs to evaluate five personalized search strategies. We find all proposed methods have significant improvements over common web search on queries with large click entropy. On the queries with small click entropy, they have similar or even worse performance than common web search. These results tell us that personalized search has different effectiveness on different queries and thus not all queries should be handled in the same manner. Click entropy can be used as a simple measurement on whether the query should be personalized and we strongly encourage the investigation of more reliable ones.

Experimental results also show that click-based personalization strategies work well. They are straightforward and stable though they can work only on repeated queries. We suggest that search engine keeps the search histories and provides convenient and secure review ways to users.

The profile-based personalized search strategies proposed in this paper are not as stable as the click-based ones. They could improve the search accuracy on some queries, but they also harm many queries. Since these strategies are far from optimal, we will continue our work to improve them in future. We also find for profile-based methods, both long-term and short-term contexts are important in improving search performance. The appropriate combination of them can be more reliable than solely using either of them.

8. ACKNOWLEDGMENTS

We are grateful to Dwight Daniels for edits and comments on writing the paper. Comments from the four anonymous referees are invaluable for us to prepare the final version.

9. REFERENCES

[1] S. M. Beitzel, E. C. Jensen, A. Chowdhury, D. Grossman, and O. Frieder. Hourly analysis of a very large topically categorized web query log. In *Proceedings of SIGIR '04*, pages 321–328, 2004.

[2] J. Boyan, D. Freitag, and T. Joachims. Evaluating retrieval performance using clickthrough data. In *Proceedings of AAAI Workshop on Internet Based Information Systems*, 1996.

[3] A. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

[4] J. M. Carroll and M. B. Rosson. Paradox of the active user. *Interfacing thought: cognitive aspects of human-computer interaction*, pages 80–111, 1987.

[5] P. A. Chirita, C. Firan, and W. Nejdl. Summarizing local context to personalize global web search. In *Proceedings of CIKM '06*, 2006.

[6] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschütter. Using odp metadata to personalize search. In *Proceedings of SIGIR '05*, pages 178–185, 2005.

[7] S. Cronen-Townsend and W. B. Croft. Quantifying query ambiguity. In *Proceedings of HLT '02*, pages 94–98, 2002.

[8] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of WWW '01*, pages 613–622, 2001.

[9] P. Ferragina and A. Gulli. A personalized search engine based on web-snippet hierarchical clustering. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 801–810, 2005.

[10] T. H. Haveliwala. Topic-sensitive pagerank. In *Proceedings of WWW '02*, 2002.

[11] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic. Real life information retrieval: a study of user queries on the web. *SIGIR Forum*, 32(1):5–17, 1998.

[12] B. J. Jansen, A. Spink, and T. Saracevic. 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, 2000.

[13] J.C.Borda. Mémoire sur les élections au scrution. *Histoire de l'Académie Royal des Sciences*, 1781.

[14] G. Jeh and J. Widom. Scaling personalized web search. In *Proceedings of WWW '03*, pages 271–279, 2003.

[15] D. H. John S. Breese and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of UAI '98*, pages 43–52, 1998.

[16] R. Krovetz and W. B. Croft. Lexical ambiguity and information retrieval. *Information Systems*, 10(2):115–141, 1992.

[17] U. Lee, Z. Liu, and J. Cho. Automatic identification of user goals in web search. In *Proceedings of WWW '05*, pages 391–400, 2005.

[18] Y. Li, Z. Zheng, and H. K. Dai. Kdd cup-2005 report: facing a great challenge. *SIGKDD Explor. Newsl.*, 7(2):91–99, 2005.

[19] F. Liu, C. Yu, and W. Meng. Personalized web search by mapping user queries to categories. In *Proceedings of CIKM '02*, pages 558–565, 2002.

[20] L. Page, S. Brin, R. Motwani, , and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Computer Science Department, Stanford University, 1998.

[21] J. Pitkow, H. Schutze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. *Commun. ACM*, 45(9):50–55, 2002.

[22] A. Pretschner and S. Gauch. Ontology based personalized search. In *Proceedings of ICTAI '99*, pages 391–398, 1999.

[23] F. Qiu and J. Cho. Automatic identification of user interest for personalized search. In *Proceedings of WWW '06*, pages 727–736, 2006.

[24] D. Shen, R. Pan, J.-T. Sun, J. J. Pan, K. Wu, J. Yin, and Q. Yang. Q2c@ust: our winning solution to query classification in kddcup 2005. *SIGKDD Explor. Newsl.*, 7(2):100–110, 2005.

[25] X. Shen, B. Tan, and C. Zhai. Context-sensitive information retrieval using implicit feedback. In *Proceedings of SIGIR '05*, pages 43–50, 2005.

[26] X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In *Proceedings of CIKM '05*, pages 824–831, 2005.

[27] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[28] M. Speretta and S. Gauch. Personalized search based on user search histories. In *Proceedings of WI '05*, pages 622–628, 2005.

[29] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of WWW '04*, pages 675–684, 2004.

[30] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. Cubesvd: a novel approach to personalized web search. In *Proceedings of WWW '05*, pages 382–390, 2005.

[31] B. Tan, X. Shen, and C. Zhai. Mining long-term search history to improve search accuracy. In *Proceedings of KDD '06*, pages 718–723, 2006.

[32] F. Tanudjaja and L. Mui. Persona: A contextualized and personalized web search. In *Proceedings of HICSS '02*, pages volume3, pp.53, 2002.

[33] J. Teevan, E. Adar, R. Jones, and M. Potts. History repeats itself: repeat queries in yahoo's logs. In *Proceedings of SIGIR '06*, pages 703–704, 2006.

[34] J. Teevan, S. T. Dumais, and E. Horvitz. Beyond the commons: Investigating the value of personalizing web search. In *Proceedings of PIA '05*, 2005.

[35] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *Proceedings of SIGIR '05*, pages 449–456, 2005.

[36] S. Wedig and O. Madani. A large-scale analysis of query logs for assessing personalization opportunities. In *Proceedings of KDD '06*, pages 742–747, 2006.

[37] Y. Xie and D. R. O'Hallaron. Locality in search engine queries and its implications for caching. In *INFOCOM '02*, 2002.