Probabilistic Models for Personalizing Web Search

David Sontag
New York University
New York, USA
dsontag@cs.nyu.edu

Kevyn Collins-Thompson
Microsoft Research
Redmond, USA
kevynct@microsoft.com

Paul N. Bennett
Microsoft Research
Redmond, USA
paul.n.bennett@microsoft.com

Ryen W. White
Microsoft Research
Redmond, USA
ryenw@microsoft.com

Susan Dumais
Microsoft Research
Redmond, USA
sdumais@microsoft.com

Bodo Billerbeck
Bing, Microsoft
Melbourne, Aus
bodob@microsoft.com

ABSTRACT

We present a new approach for personalizing Web search results to a specific user. Ranking functions for Web search engines are typically trained by machine learning algorithms using either direct human relevance judgments or indirect judgments obtained from click-through data from millions of users. The rankings are thus optimized to this generic population of users, not to any specific user. We propose a generative model of relevance which can be used to infer the relevance of a document to a specific user for a search query. The user-specific parameters of this generative model constitute a compact user profile. We show how to learn these profiles from a user's long-term search history. Our algorithm for computing the personalized ranking is simple and has little computational overhead. We evaluate our personalization approach using historical search data from thousands of users of a major Web search engine. Our findings demonstrate gains in retrieval performance for queries with high ambiguity, with particularly large improvements for acronym queries.

Categories and Subject Descriptors

H.3.3 [**Information Retrieval**]: Retrieval Models

General Terms

Algorithms

Keywords

Personalization, re-ranking, probabilistic models, machine learning

1. INTRODUCTION

Search personalization typically involves tailoring the ranking of results for individual users based on models of their interests. Personalization has been shown to be useful for improving retrieval effectiveness [15, 17, 20], but there has been little work on developing robust probabilistic formalisms or in evaluating these algorithms at Web scale.

The three key problems that must be solved in any personalization approach are: representation, learning, and ranking. Specifically, for representation we need some way to compactly summarize the interests or preferences for each user into a user profile. For learning, we need an algorithm to discover these user profiles from data. Finally, we need an algorithm to combine these user profiles with other relevance signals to rank documents with respect to a query.

Our paper proposes an end-to-end system for personalization addressing each of the above challenges. We formalize the problem using a probabilistic model for predicting the relevance of a document to a specific user with respect to a query. The user representation corresponds to user-specific parameters for part of the model. Our formalization is general and assumes only that there are document-specific latent variables (i.e., document features), user-specific latent variables (i.e., information need for this query), and some way of combining them to determine whether a document's features satisfy the user's information need.

Our approach begins with the assumption that the Web search engine provides a generic estimate of the probability that a document is relevant to the query. Since relevance is subjective, different people will find different documents relevant for the same query and no single ranking can satisfy all users [19, 20]. We explicitly consider the distribution of users for which the global ranking function was trained, and identify how a specific user is different from the population as a whole. Using this, we deconvolve the relevance probability into the probability that a page is relevant given any specific query intent. Then, we recompute the probability of relevance taking into consideration the user's profile.

Although our formalization is general, in this paper we specifically consider its application to the task of personalization using topic-based profiles. We have one discrete variable for each document whose states specify the topic of the document. The state space that we use corresponds to the top two levels of the human-generated ontology provided by the Open Directory Project (ODP, dmoz.org). Some example categories are 'Sports', 'Arts/Movies', and 'Shopping'. In a pre-processing step, we use a text-based classifier, trained with logistic regression, to obtain the distribution over topics for each document in the index. This allows the personalized ranking to be computed extremely quickly at query time.

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

In addition to having one variable per document, we have one variable for the user whose states specify the topic of the documents being searched for using the query. Even before seeing the query, the user's history provides a prior distribution for this variable. For example, if there are previous queries and clicks in this search session, these could be used to perform short-term personalization. In this paper we focus on personalization using *long-term* search histories, which have been less thoroughly investigated, especially with large numbers of users (see Section 4 for more details).

We evaluate our approach using the Bing search logs. We use the queries and search-result clicks in these logs to obtain the queries and search results, to build long-term profiles of user search interests, and to obtain personalized relevance judgments for each user-query pair based on search result clickthrough. As we will show, the methods lead to significant gains in retrieval effectiveness over competitive baselines.

The remainder of this paper is structured as follows. Section 2 describes related work in areas such as personalization and result re-ranking. Section 3 describes our personalization framework based on probabilistic models, and Section 4 describes how we learn user profiles from users' search history. Experiments to determine the performance of our models are described in Section 5, and we conclude and highlight possible areas of future work in Section 6.

2. RELATED WORK

There is a growing interest in the information retrieval and machine learning communities in moving beyond context-free search experiences, and toward examining how knowledge of a searcher's interests and search context can be used to improve various aspects of search (e.g., ranking, query suggestion, query classification). For example, there has been work on using session context, such as the previous few searches or result clicks, to personalize search results and improve retrieval performance [6, 15]. Short-term session profiles have also been used for other tasks such as predicting future interests [23], query categorization [3], query suggestion, and URL recommendation [2]. We focus on personalizing using user profiles constructed from logs comprising long-term interaction behaviors, potentially providing a richer view of searcher interests over time.

Another line of prior research uses long-term histories to directly improve retrieval effectiveness. Teevan *et al.* [18] constructed user profiles from indexed desktop documents and showed that this information could be used to re-rank search results and improve relevance for individuals. Matthijs and Radlinski [13] constructed user profiles using users' browsing history, and evaluated their approach using an interleaving methodology. Rather than using all of the previous search history, Tan *et al.* [17] focused only on the most relevant prior queries and constructed language models for this task. Personalization is not equally effective on all queries. Teevan *et al.* [19, 20] introduced a framework to identify the potential-for-personalization for different queries. In particular, the implicit measure click entropy (the number of different results that different people clicked) was highly correlated with explicit judgments of relevance by individuals. All of these approaches to personalization use word-based profiles, and ranking is done by re-weighting terms using an existing scoring method such as BM25 or TF-IDF. In contrast, our approach uses a higher-level representation. One of the key advantages of such a representation is that it allows us to naturally build on top of the probability of relevance computed by a more complex ranking function, such as that of a commercial search engine.

Various authors have considered topic-based representations for personalization, typically learning a user's profile from either browsing or search history [4, 7, 9, 12, 14, 16]. These papers suggest a variety of heuristic methods for ranking using a user's profile. Our approach differs significantly in that we propose a probabilistic framework for personalization, resulting in principled procedures for: (1) estimating the topic that a user is searching for, given the query and user profile, and (2) computing the personalized ranking by combining what we know about the user with other relevance signals. A notable exception is Zhai *et al.* [24], who studied the incorporation of novelty into search results. They propose a probabilistic approach that is similar in spirit to the first model that we present in Section 3. Our approach also differs from these earlier works in that we propose using a *background model*, explicitly taking into consideration the relative likelihood that this user, compared to a generic user, has a particular search intent. We show in the large-scale evaluation of Section 5 that the background model results in large improvements in ranking accuracy over a personalization approach that does not use the background model.

3. PROBABILISTIC MODELS FOR PERSONALIZATION

In this section we present two probabilistic models and inference algorithms for computing the probability that a document d is relevant to user u for the query q. These probabilistic models are called *generative models* because they describe the process by which a user decides whether a document is relevant to a particular query. We have a single variable for the document, T_d, and a single variable for the user, T_u. These discrete-valued variables refer to the document's topic and the topic that the user is searching for, respectively. A document about topic T_d is assumed relevant to a user looking for topic T_u if both:

1. Topic T_d satisfies a user with information need T_u, and
2. Given that the document's topic matches that of the search intent, the document is relevant to the query.

The first criterion is measured by the variable $\text{cov}_u(d, q) \in \{0, 1\}$, which represents the extent to which T_d "covers" the information need T_u. The second criterion is measured by the variable $\psi(d, q) \in [0, 1]$, which we call the *non-topical* relevance score, corresponding to the user-independent probability that the document is relevant to this query. This score is assumed to be comprised of a large number of user-independent signals such as the match of the query to document text or anchor text, aggregate user behavior for this query, etc.

In Model 1 we assume that $\psi(d, q)$ is observed, given to us by the search engine. This turns out to be a very strong assumption, and we relax it in Section 3.2 when we describe Model 2. We intentionally do not model how this score arises, instead choosing to take a black-box view of it. The reason for this choice is that modern Web search engines use a large number of relevance signals that are typically combined in a complex fashion, and we want our algorithms to be broadly applicable.

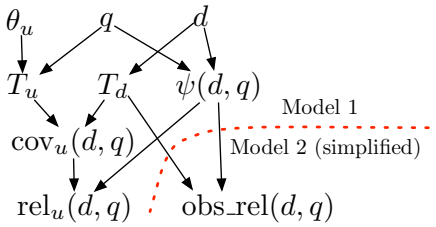

Figure 1: Graphical illustration of two probabilistic models used for personalization. Arrows denote dependencies between the variables [11]. The first approach, Model 1, is described in Section 3.1. The second, Model 2 (simplified), additionally includes the variable obs_rel(d,q), which is the expected probability of relevance with respect to the distribution of users that typically search for query q. It is equivalent to Model 2 from Fig. 2, except that the background model, i.e. the variables in the box, are integrated out.

The variable $\text{rel}_u(d,q) \in \{0,1\}$ combines the two criteria, taking the value 1 when the user finds the document relevant to the query, and 0 otherwise. Specifically, we have $\Pr(\text{rel}_u(d,q)=1 \mid \text{cov}_u(d,q), \psi(d,q)) = 0$ if $\text{cov}_u(d,q)$ is 0, and $\psi(d,q)$ if $\text{cov}_u(d,q)$ is 1. The user's personalized ranking is then obtained by sorting all of the documents by this probability.[1] In practice, this approach would be used to *re-rank* a number of top-scoring documents with respect to the user-independent ranking function.

3.1 Model 1 (no background model)

Fig. 1 shows the probabilistic model for the simplest personalization method, denoted as Model 1.

The notation θ_u refers to user-specific parameters, also called the user profile, that are learned from the user's historical data in an offline step. The user profile together with the current query q are used to come up with a *distribution* over the user's search intent (i.e. a distribution over topics), $\Pr(T_u \mid \theta_u, q)$. Our framework is modular, with many different data sources able to feed into this distribution. For example, one could do short-term personalization by conditioning on the previous queries of the session. In Section 4 we describe our approach to estimating this distribution by learning the user profiles from long-term user interaction data.

The conditional distribution $\Pr(T_d \mid d)$ specifies the topic of each document. This distribution could be estimated using a variety of techniques. In our evaluation, we use a text-based classifier, described in [1], that was trained using logistic regression to predict the ODP category for each Web page present in the Bing index.

The distribution $\Pr(\text{cov}_u(d,q) \mid T_u, T_d)$ could simply be given by $1[T_u = T_d]$, the indicator function for whether T_u is the same as T_d. This choice would imply that a document is irrelevant for queries outside of its topic area. More generally, $\Pr(\text{cov}_u(d,q) \mid T_u, T_d)$ can be a function of some distance between topics T_u and T_d, or it could be learned from data.

[1]There are also more sophisticated methods for personalizing the results using the probabilistic model, such as blending the personalized results with the original search results, and considering the confidence in our estimates when deciding whether to personalize.

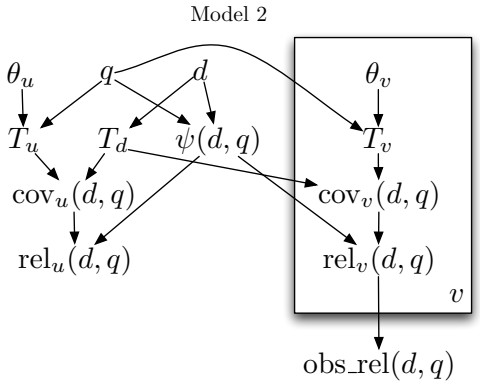

Figure 2: Graphical illustration of the probabilistic model of personalization for user u where we explicitly take into consideration the background distribution of users (denoted as v) who typically search the query q and for whom the ranking function was trained. The box notation used in this figure signifies that each of the variables inside the box should be replicated, once for each user $v \in V(q)$ (see Section 3.2). We assume that the probability of relevance reported by the search engine, obs_rel(d,q), is equal to the expectation of $\text{rel}_v(d,q)$ with respect to the distribution of users that typically search on q, described by $\Pr(\theta_v \mid q)$. The only observed variables are the user-specific parameters θ_u, the query q, the document d, and obs_rel(d,q).

The variables θ_u, q, d, and $\psi(d,q)$ are observed. Integrating over all of the latent variables, we obtain the following formula to use during ranking:

$$\begin{aligned}\Pr(\text{rel}_u(d,q)=1 \mid \theta_u, q, d, \psi(q,d)) \\ = \psi(d,q) \sum_{T_d} \Pr(T_d \mid d)\alpha(T_d) \ , \quad (1)\\ \alpha(T_d) = \sum_{T_u} \Pr(T_u \mid \theta_u, q)\Pr(\text{cov}_u(d,q)=1 \mid T_u, T_d)\end{aligned}$$

$\alpha(T_d)$ can be computed just once for each query, regardless of the number of documents to be ranked. Furthermore, for a specific document d, $\Pr(T_d \mid d)$ typically has support on only a few categories (having close to zero probability elsewhere), and so the sum in Eq. 1 has very few terms. As a result, we can compute the personalized probabilities of relevance in linear time with respect to the number of documents to be ranked. Over time, the profile θ_u can be updated efficiently as we see additional behavior from users.

3.2 Model 2 (background model)

The approach described in the previous section assumes that the search engine provides the non-topical relevance score, $\psi(d,q)$, which is the user-independent probability that the document is relevant. However, this quantity is difficult to obtain. The ranking functions of modern search engines are trained using a combination of hand-labeled relevance judgments and implicit relevance judgments from millions of users' click-through information. Thus, the probability of relevance obtained from the ranking function is biased toward the population of users that typically search on the query using this search engine.

For example, on the query "Kevin Murphy", the top search results using Bing are about a hair stylist and an actor. This

is not to say that the other results, such as that of Kevin Patrick Murphy (a researcher who frequently publishes in Computer Science) are irrelevant, just that for the generic user they are less likely to be relevant.

Suppose that we knew the set of users $V(q) = \{v\}$ that have previously searched for query q and whose relevance feedback we used to train the ranking function. The probabilistic model described in this section, shown in Fig. 2, explicitly takes these users' intended topics into consideration when interpreting the probability of relevance computed by the ranking function. In particular, we assume that, rather than $\psi(d, q)$, the search engine only provides the following quantity:

$$\text{obs\_rel}(d, q) = \frac{1}{|V(q)|} \sum_{v \in V(q)} \text{rel}_v(d, q) \ .$$

This is the *expected* relevance with respect to the distribution of users who typically search for query q (across all possible query intents). We no longer assume that $\psi(d, q)$ is observed.

Since it is unrealistic to assume that we know $V(q)$, we instead propose to use an aggregate distribution. In particular, consider a simplified version of Model 2, shown in Fig. 1, where we integrated over the variables T_v, $\text{cov}_v(d, q)$, and $\text{rel}_v(d, q)$ for all $v \in V(q)$. Now $\text{obs\_rel}(d, q)$ depends on $\psi(d, q)$, q, and T_d, and can be shown to be equal to:

$$\psi(d, q) \sum_T \Pr(\text{cov}(d, q) = 1 \mid T, T_d) \Pr_r(T \mid q), \qquad (2)$$

where we define:

$$\Pr_r(T \mid q) = \frac{1}{|V(q)|} \sum_{v \in V(q)} \Pr(T \mid \theta_v, q).$$

Instead of assuming that we know $V(q)$, we assume that we know this *background distribution* $\Pr_r(T \mid q)$ (the r denotes that this is for a random, or generic user). We propose estimating $\Pr_r(T \mid q)$ using an approach similar to [23]. Assuming that the existing ranking function is targeted at the generic user, we compute this distribution by taking the weighted average of the topic distributions (as computed by our classifier) for each of the top-scoring search results. Specifically, we estimate:

$$\Pr_r(T = t \mid q) \propto \sum_{i=1}^{N} \text{obs\_rel}(d_i, q) \Pr(T_{d_i} = t \mid d_i)$$

for the N highest scoring documents according to $\text{obs\_rel}(d, q)$.

As with Model 1, we next compute the posterior marginals $\Pr(\text{rel}_u(d, q) = 1 \mid \theta_u, q, d, \text{obs\_rel}(d, q))$, and sort these to obtain the personalized rankings. From Eq. 2 we obtain

$$\psi(d, q) = \frac{\text{obs\_rel}(d, q)}{\sum_T \Pr(\text{cov}(d, q) = 1 \mid T, T_d) \Pr_r(T \mid q)} \ .$$

Note that this is a random variable because of its dependence on T_d. To simplify probabilistic inference, we assume:

ASSUMPTION 1. *$Pr(T_d \mid d, obs\_rel(d, q)) \approx Pr(T_d \mid d)$.*

That is, although the observed relevance of a document with respect to a query does provide some information about the document's topic, we assume that this is dwarfed by the information contained in the document's text. Using this, we integrate over T_u and T_d, and obtain the following for the posterior marginal:

$$\text{obs\_rel}(d, q) \overbrace{\sum_{T_d} \Pr(T_d \mid d) \frac{\sum_{T_u} \Pr(T_u \mid \theta_u, q) f(T_u, T_d)}{\sum_T \Pr_r(T \mid q) f(T, T_d)}}^{\text{Re-weighting factor}}, \qquad (3)$$

where $f(T, T_d) = \Pr(\text{cov}(d, q) = 1 \mid T, T_d)$. The numerator of the fraction is $\alpha(T_d)$ (see Eq. 1) and, as before, can be computed just once per query (same for the denominator).

The personalized relevance score given by Model 2 (specified in Eq. 3) satisfies an important invariance property:

> *When $Pr(T_u \mid \theta_u, q)$ is the same as $Pr_r(T \mid q)$, the ranking is unchanged.*

That is, given what we know about the user, if we cannot distinguish the user's query intent from that of the general population of users that search for this query, then the reweighting factor has value 1 and the personalized probability of relevance is simply given by $\text{obs\_rel}(d, q)$. We believe that this invariance property is essential to our approach's success. To our knowledge, our approach is the first personalization algorithm that explicitly uses a background distribution and satisfies this invariance property.

We conclude this section by noting that $\Pr_r(T \mid q)$ could come from a variety of different sources. For example, rather than estimating it using the approach described above, for popular queries we could simply collect statistics for the number of times that users click on results that are labeled with category T given that they have searched for query q. More broadly, our method could be used to adjust the rankings of any search engine that was trained on one set of users but is then applied to a different set of users.

4. MODELING USERS

In this section we describe how we use a user's long-term search history to compute a compact user profile. In particular, to apply the personalization approaches introduced in the previous section, we need the distribution $\Pr(T_u \mid \theta_u, q)$, the probability that when issuing a query q, a user u is seeking information on topic T_u. θ_u denotes the user-specific parameters. We propose two different approaches for obtaining this conditional distribution:

1. Learn a user-independent language model $\Pr(q \mid T)$ and a user-specific prior $\Pr(T \mid \theta_u)$, and then apply Bayes' rule:

$$\Pr(T_u \mid \theta_u, q) = \frac{\Pr(T_u \mid \theta_u) \Pr(q \mid T_u)}{\sum_{T'} \Pr(T' \mid \theta_u) \Pr(q \mid T')}. \qquad (4)$$

2. Learn a user-specific way of reweighting $\Pr_r(T \mid q)$ to obtain $\Pr(T_u \mid \theta_u, q)$.

The first approach (Generative method) is based on the generative models described in the previous section. It assumes that the user first chooses a topic to search for, and then chooses a particular query from this topic. The second approach (Discriminative method) we formulate as a discriminative learning task, directly attempting to maximize the probability of the user's actual intent conditioned on the query, over their previous search history.

Both approaches have their advantages and disadvantages. The first approach gives an accurate picture of a user's typical intent distribution, but relies heavily on being able to estimate a good language model. For some infrequently

searched categories, such as Computer Science, we do not observe sufficiently many queries to estimate a good language model. As a result, we may end up missing these intents for all but their most popular queries. The second approach usually does a good job of predicting a user's most likely intent, but can give very peaked distributions, underestimating the model's uncertainty in the user's intent.

We also evaluated a convex combination of the two distributions (50% each) which we call the Interpolation method. Averaging the output of the two different classifiers is a type of ensemble method, frequently used in machine learning, and results in a more stable prediction. We show in the evaluation that this results in significantly fewer queries where the personalized ranking is worse than the original ranking.

4.1 Training data

We assume that we have search history for each user consisting of the queries issued, the list of documents in the visible search results, and the list of documents clicked on by the user in response to each query. There has been a significant amount of work on interpreting click-through information as a weak relevance signal, and it is straightforward to use these approaches together with our probabilistic models. The simplest approach, for example, equates a user's click on a document with the observation $\text{rel}_u(d, q) = 1$ and, conversely, the lack of a click as $\text{rel}_u(d, q) = 0$. Then, we could estimate the user's parameters by maximizing the likelihood of the observed click-through data. Unfortunately, the resulting learning problem is computationally difficult to solve because some of the variables, e.g. the user's true intent T_u, are unknown and their values must be integrated over.

We could use expectation maximization (EM) or gradient ascent to reach a local maximum of the likelihood. However, in this paper we avoid this complexity by making a simple approximation. We assume that the user's intended topic T_u is equal to the topic of the document that they click on. Specifically, let $d_1, \ldots, d_c$ be the documents that the user clicks on for query q. Then, we let

$$\widehat{\Pr}(T)_t = \frac{1}{c} \sum_{i=1}^{c} \Pr(T \mid d_i),$$

where the documents' topic distributions are computed by our classifier and the subscript t refers to a specific query. Then, the training data for each user consists of a set of pairs, $(q_t, \widehat{\Pr}(T)_t)$.

This approximation corresponds to ignoring the negative data points (i.e., documents that a user *does not* click on), assuming that a click implies that a user thinks that the document is relevant, and assuming that $\Pr(\text{cov}_u(d, q) \mid T_u, T_d) = 0$ if $T_u \neq T_d$. The first two assumptions are often very reasonable, and become less important the more training data that we have for a user.

4.2 Language model

In this approach, we estimate the prior probability that a user u searches for topic T, independent of the query. Let N denote the number of training points, $(q_t, \widehat{\Pr}(T)_t)$. Then, our estimate is:

$$\Pr(T \mid \theta_u) = \frac{1}{N} \sum_{t=1}^{N} \widehat{\Pr}(T)_t.$$

where we ignore the queries.

To apply Eq. 4 we estimate a unigram language model, $\Pr(q \mid T) = \prod_{w \in q} \Pr(w \mid T)$ for each topic T, where w denotes a word. We calculate the required statistics using half a billion search queries sampled from a month of web search query logs, and the topic distributions of the clicked documents as computed by our classifier.

4.3 Reweighting the generic user's intent

In this section, we propose a discriminative approach to directly estimate the conditional distribution $\Pr(T_u \mid \theta_u, q)$ from training data. We begin by assuming that the conditional distribution lies in the exponential family, with the following parametric form:

$$\Pr(T \mid q; \theta) = \exp(\phi(T, q) \cdot \theta - A(\theta)), \tag{5}$$

where $A(\theta)$ denotes the log partition function. The feature vector we use is:

$$\phi(T, q) = < \log \Pr_r(T \mid q), 0, 0, \ldots, 0, 1, 0, \ldots 0 > \ ,$$

where the 1 is in the T'th location. That is, the parameters that we learn correspond to a user-specific reweighting of $\Pr_r(T \mid q)$, the topic distribution for a generic user who searches on query q. We learn one multiplier per topic.

We learn a different parameter vector θ for each user. Our goal during learning is to minimize

$$\sum_{t=1}^{N} \text{KL}\left(\widehat{\Pr}(T)_t, \Pr(T \mid q; \theta)\right) + C_1(\theta_0 - 1)^2 + C_2 ||\theta_{1:\text{end}}||^2 \tag{6}$$

where θ_0 refers to the weight which multiplies $\log \Pr_r(T \mid q)$. We also require that $\theta_0 \geq 0$.

Ignoring constant terms, $\text{KL}\left(\widehat{\Pr}(T)_t, \Pr(T \mid q; \theta)\right)$ equals

$$\begin{aligned} & -\sum_T \widehat{\Pr}(T)_t \log \Pr(T \mid q; \theta) \\ = \ & \log \sum_T \exp(\phi(T, q) \cdot \theta) - \sum_T \widehat{\Pr}(T)_t \phi(T, q) \cdot \theta. \end{aligned}$$

The objective in Eq. 6 is convex, and can be optimized using a number of standard methods.

The quadratic terms in Eq. 6 regularize the data terms to prevent the learning algorithm from overfitting. The regularization is motivated by the following generative model. Suppose that we estimate $\Pr_r(T)$, the prior distribution over topics for generic users of the search engine. We can apply Bayes' rule to "invert" $\Pr_r(T \mid q)$:

$$\Pr(q \mid T) = c \frac{\Pr_r(T \mid q)}{\Pr_r(T)}, \tag{7}$$

where c is a constant. Then, using Eq. 7 as our new language model, we do a second application of Bayes' rule as in Eq. 4 to obtain:

$$\Pr(T_u \mid \theta_u, q) \propto \Pr_r(T_u \mid q) \frac{\Pr(T_u \mid \theta_u)}{\Pr_r(T_u)}. \tag{8}$$

Setting the parameters $\theta_0 = 1$ and $\theta_T = \log \frac{\Pr(T \mid \theta_u)}{\Pr_r(T)}$ in Eq. 5 recovers Eq. 8. The regularization then says that, unless we see a lot of evidence to the contrary, we think θ_0 should be equal to 1 and θ_T should be equal to 0, where the latter corresponds to setting $\Pr(T \mid \theta_u) = \Pr_r(T)$.

5. EVALUATION

In Section 5.1 we present examples that illustrate the different components of our personalization framework and how applying personalization with topic-based user profiles affects re-ranking. Then in Section 5.2 we summarize key performance metrics for the models described above.

Our models make use of a probability of relevance that is supposed to be provided by the search engine, $\text{obs\_rel}(d, q)$. When the score provided by the ranking function cannot be easily interpreted as a probability of relevance (e.g., because it can be negative), a simple substitute is to use the inverse rank of the document. Rather than use the ranking score, we use this second alternative of inverse rank. This enables others without access to commercial search engine logs to reproduce our work as a client-side personalized re-ranking study, such as the one performed in [13].

We obtain the personalized ranking by ordering the results according to β*original score + $(1 - \beta)$*personalized score. In our experiments we set $\beta = 0.3$, where β serves as a serendipity parameter. This corresponds to a generative model where with probability β the user's intent matches the generic user, ignoring the user's typical interests.

We learn the coverage function $\Pr(\text{cov}_u(d, q) = 1 \mid T_u, T_d)$, which describes the relationship between the classes and the extent to which a user's intent T_u (e.g., "Computers/Artificial Intelligence") is satisfied by a document's topic T_d (e.g., "Computers"). We used a hold-out set of one month's worth of training logs. In particular, we assume:

$$\Pr(\text{cov}_u(d, q) = 1 \mid T_u, T_d = t) = \frac{\Pr(d \text{ satisfies } u \mid u \text{ has intent } T_u, T_d = t)}{\max_c \Pr(d \text{ satisfies } u \mid u \text{ has intent } T_u, T_d = c)} \quad (9)$$

The max term normalizes with respect to the most frequently satisfying class, to account for the fact that multiple document topics may align well with a user's intent. For example, "Shopping/Vehicles" and "Recreation/Autos" may both equally satisfy queries about automobiles. This is often because how information needs are distributed across an ontology may differ in practice from how the ontology was designed. We compute an empirical estimate of the satisfaction probability as follows:

$$\Pr(d \text{ satisfies } u \mid u \text{ has intent } T_u, T_d = t) = \frac{\frac{1}{|Q|}\sum_{q \in Q} \Pr_r(T_u \mid q) \sum_{d' \in \text{results}(q)} \text{sat}(d', q) \Pr(T_{d'} = t \mid d')}{\frac{1}{|Q|}\sum_{q \in Q} \Pr_r(T_u \mid q)}$$

Here the indicator function $\text{sat}(d', q)$ is 1 if d' received the *last* satisfied result click for query q, and 0 otherwise. We define a satisfied result click (SAT) as either a click followed by no further clicks for 30 seconds or more, or the last result click in the session [17, 21]. The set Q corresponds to all queries with at least one satisfied result click.

We implemented the discriminative learning algorithm for learning the user profiles using Matlab and `CVX`, a package for specifying and solving convex programs [10]. We set the regularization parameters to be $C_1 = 25$ and $C_2 = 0.5$.

5.1 Illustrative examples

In this section, we demonstrate various aspects of our personalization approach using a demo that we implemented to re-rank the top 200 Bing search results. The user profiles used in the demo were learned from two months of search logs from one of the authors, a computer scientist, and from another volunteer, a biologist.

Fig. 3 shows the results of our algorithms for the ambiguous query `[kevin murphy]` issued by the computer science researcher. Fig. 3(a) shows the top five ODP categories from the background model, $\Pr_r(T \mid q)$, and also the top five ODP categories that our algorithms predict as the query intent for the computer science researcher, given by $\Pr(T_u \mid \theta_u, q)$. There are marked differences. The distribution over query intents for the generic user for the `[kevin murphy]` query is centered around business, society, and health, whereas for the computer science researcher, the predicted query intent involved artificial intelligence, people, and science.

Fig. 3(b) presents the search results returned for this query, issued by the computer science researcher, to: (i) Bing, (ii) the same Web search engine with results re-ranked using Eq. 1, and (iii) the same engine with results re-ranked using Eq. 3. When a computer science researcher issues this query, it is likely that the intent is to reach the website of the University of British Columbia (UBC) professor (`http://www.cs.ubc.ca/~murphyk`), and not the actor or hair stylist. As can be seen from the example, both Eq. 1 and Eq. 3 promote the UBC professor's page from outside the top 10 results.

Fig. 4 shows the results of our algorithms for the ambiguous query `[rockefeller]` issued by the biologist. We see in Fig. 4(a) that, again, there are marked differences in the query intents, with the dominant categories for the generic user centering on business, society, and health, whereas for the biologist they center on biology, science, and health. Fig. 4(b) shows the results returned for the query `[rockefeller]` issued by the biologist to: (i) Bing, (ii) the same Web search engine with results re-ranked using Eq. 1, and (iii) the same engine with results re-ranked with Eq. 3.

We observed across a large number of ambiguous queries that Eq. 3 (Model 2) performs significantly better than Eq. 1 (Model 1), typically promoting the desired result directly to the top position. The algorithms appear to work particularly well for name queries (e.g., on the query `[Michael Jordan]`, promoting the website of the statistician to position 1 from position 198 when queried by the computer scientist) and acronyms (e.g., `[sigir]`).

5.2 Large-scale evaluation

The primary source of data for this study is a proprietary data set comprising the search logs (queries and result clicks) for the Bing Web search engine. The data set consists of tuples including a random unique user identifier (stored in a browser cookie), the date and time, and the query issued. For each of the queries, we also know the top-10 search results that were shown to users at query time, the rank order in which they were presented, and which results were clicked on. These data provide us with examples of real-world searching behavior that are useful for evaluating the performance of our personalized search algorithms. To isolate the impact of long-term personalization, we did not use any other form of personalization from the Bing search engine over the time period for which the data were collected. To remove variability caused by cultural and linguistic variation in search behavior, we only include log entries generated in the English-speaking United States locale.

The evaluation results described in this paper are based on queries and result clicks in this data during September

Pr(topic | query) for generic user
Business: 0.213
Society: 0.107
Shopping/Health: 0.096
Business/Consumer Goods+Services: 0.077
Arts: 0.062

Pr(topic | query) for CS researcher
Computers/Artificial Intelligence: 0.663
Arts/People: 0.098
Science: 0.044
Computers: 0.042
Arts/Performing Arts: 0.036

(a)

Web search engine results	Categories
1. http://www.kevinmurphy.com.au	Business, Shopping
2. http://en.wikipedia.org/wiki/Kevin_Murphy_(actor)	Arts
3. http://www.kevinmurphystore.com	Health, Shopping

Personalized re-ranking results (using Model 1)		Categories
1. http://en.wikipedia.org/wiki/Kevin_Murphy_(actor)	(2)	Arts
2. http://www.kevinmurphy.com.au	(1)	Business, Shopping
3. http://www.cs.ubc.ca/~murphyk	(13)	Reference, Computers

Personalized re-ranking results (using Model 2)		Categories
1. http://www.cs.ubc.ca/~murphyk	(13)	Reference, Computers
2. http://en.wikipedia.org/wiki/Kevin_Murphy_(actor)	(2)	Arts
3. http://www.kevinmurphystore.com	(3)	Health, Shopping

(b)

Figure 3: (a) Top categories based on Pr(topic | query) for a generic user and a computer science researcher for the query [kevin murphy]. (b) The original top three results from a Web search engine for query [kevin murphy], and re-ranked results using Models 1 and 2. Also shown to the right of each result is the original rank in parentheses and the top-level ODP categories, as predicted by the text classifier used throughout this paper.

2010. 20 days of search logs from Sept. 1-20 were used to construct users' long-term profiles. The queries in five days of search logs from Sept. 21-25 were used to evaluate the performance of our personalization algorithms. We selected users from the 5-day test period who had at least 100 satisfied result clicks in the 20-day profile building period (see Table 1). For this subset of users, we also identified search sessions using a session extraction methodology similar to [22]. Search sessions begin with a query and contain result clicks and any subsequent queries and clicks that occurred. Sessions terminated following 30 minutes of inactivity. We used these sessions to obtain personalized relevance judgments for each query (see below for more details).

Table 1: Descriptive statistics about our users, after filtering for those who had at least 100 SAT clicks, computed on the 20 days of search history.

	average	stdev	median
num days	16.21	3.72	17
num queries	229.60	112.28	204
num SAT clicks	143.82	52.80	128

To explore parameter choices, we use a set of five weeks of hold-out log-data from the same search engine and of a similar type to our evaluation data described above but non-overlapping with it. In particular, this hold-out data was used to explore the parameter choices mentioned in this section (e.g., β), learn the coverage function as described earlier, and set a threshold for the entropy criteria used to identify ambiguous queries, described later.

To focus on underspecified queries which [20] have found especially amenable to personalization, we filtered the test queries to only include one word queries. We also filtered out the one word queries that we have not seen sufficiently many times in the historical query logs to reliably estimate the language model.[2]

This resulted in 571598 queries from 195108 users. In our primary experiments, to further emphasize ambiguity, we retained only non-navigational queries (using a classifier) and queries where the entropy of the ODP topics of the top 10 URLs (*i.e.*, the entropy of $\Pr_r(T_d \mid q)$) is above a threshold. We refer to the queries that have passed the entropy filter as "ambiguous" in our results below. After these filters, our test set consisted of 54581 users with at least one query, and 102417 queries in total.

Evaluation of our personalized ranking algorithms required a personalized relevance judgment for each result. Obtaining relevance judgments from a large number of real users is impractical, and there is no known approach to train expert judges to provide reliable personalized judgments that reflect real user preferences. Instead, we obtained these judgments using a log-based methodology inspired by [8]. Specifically, we assign a positive judgment to one of the top 10 URLs if it is the last satisfied result click in the session (Last SAT). The remaining top-ranked URLs receive a negative judgment. This gives us one positive judgment and nine negative judgments for each of the top-10 URLs for each session.

One consequence of evaluating on retrospective data is that we can only evaluate based on the search results which were shown. Since items below the last clicked item may have been unexamined by the user and actually be relevant, treating them as irrelevant serves as a lower bound on the performance of our algorithms.

The rank position of the single positive judgment is used to evaluate retrieval performance before and after re-ranking. Specifically, we measure our performance using the inverse

[2]In particular, we considered words w that had at least one category c such that w was part of at least 50 queries leading to a click on a document with category c.

Pr(topic | query) for generic user
Business: 0.213
Society: 0.107
Shopping/Health: 0.096
Business/Consumer Goods+Services: 0.077
Arts: 0.062

Pr(topic | query) for biologist
Science/Biology: 0.402
Science: 0.228
Society: 0.052
Reference: 0.040
Health: 0.031

(a)

Web search engine results	Categories
1. http://en.wikipedia.org/wiki/John_D._Rockefeller	Society
2. http://en.wikipedia.org/wiki/Rockefeller_family	Science, Society
3. http://www.rockefeller.edu	Reference, Science

Personalized re-ranking results (using Model 1)		Categories
1. http://en.wikipedia.org/wiki/Rockefeller_family	(2)	Science, Society
2. http://www.rockefeller.edu	(3)	Reference, Science
3. http://en.wikipedia.org/wiki/John_D._Rockefeller	(1)	Society

Personalized re-ranking results (using Model 2)		Categories
1. http://www.rockefeller.edu	(3)	Reference, Science
2. http://en.wikipedia.org/wiki/Rockefeller_family	(2)	Science, Society
3. http://bridges.rockefeller.edu/?page=news	(12)	Science, Health

(b)

Figure 4: (a) Top categories based on Pr(topic | query) for a generic user and a biologist for the query [rockefeller]. (b) Top three results from a Web search engine for query [rockefeller], and re-ranked search results using Models 1 and 2. Also shown to the right of each result is the original rank in parentheses and the top-level ODP categories assigned to that result.

of the rank of the relevant document, otherwise known as the mean reciprocal rank (MRR). Queries for which we cannot assign a positive judgment to any top-10 URL are excluded from the evaluation dataset.

We labeled each of top-10 results with ODP categories using a text-based classifier, described in [1] and mentioned earlier in the paper. The text-based classifier has a micro-averaged F1 value of 0.60. The coverage of the classifier across all result URLs was 86.2%; classifier coverage was not 100% due to index churn over time. When producing the personalized search results, we do not change the position of any URL for which we do not have ODP classifications.

5.2.1 Retrieval performance

Table 2 shows the change in MRR of the queries in the test set for each of the three methods of predicting the user's query intent, $\Pr(T_u \mid \theta_u, q)$, and for both Models 1 and 2. The generative method refers to the language modeling approach described in Section 4.2, the discriminative method refers to the user-specific re-weighting of $\Pr_r(T \mid q)$, described in Section 4.3, and the interpolation method refers to using the convex combination of the distributions predicted by the generative and discriminative methods.

The baseline for these experiments is the original ranking provided by the Bing Web search engine. The results are shown relative to the MRR of the baseline.[3] The first column summarizes the proportion of queries in which the method moves the last SAT click, which reflects the coverage of the method. The second column shows the MRR for this subset of queries. The final column shows the overall effect of this method, obtained by multiplying the first and second columns. All of the methods improve on the baseline, with the best results achieved by using the background model (Model 2) together with the interpolation method.

[3]To help interpret MRR Δ, if the last satisfied clicked document was always returned in the fourth position by the baseline and the personalized ranking always returned it in the third position, then this would be an MRR Δ of 0.0833. Moving from third to second would yield a Δ of 0.1667.

Table 2: Performance on ambiguous, one word non-navigational queries. MRR Δ is improvement over the baseline commercial search engine's ranking. Bold face indicates significant ($p = 0.05$, Bonferroni correction) improvement over the baseline according to a sign test.

Model	Last SAT Moved	Moved MRR Δ	MRR Δ
Generative, Model 1	8.93%	**0.0753**	**0.0067**
Generative, Model 2	18.41%	0.0187	0.0034
Discriminative, Model 1	4.22%	**0.0732**	**0.0031**
Discriminative, Model 2	7.96%	**0.1808**	**0.0144**
Interpolated, Model 1	5.23%	**0.0957**	**0.0050**
Interpolated, Model 2	11.18%	**0.1686**	**0.0189**

In general, Model 2 appears to be more aggressive than Model 1, re-ranking more often (as can be seen in the first column). This is because while Model 1 may change scores, it often does not change scores enough to change the ranking. Because Model 2 normalizes by the background model, a document whose topic is substantially more likely to be the intent of the user than of the generic user has its score dramatically amplified, even if the absolute probability of this topic being the user intent is small. It is thus essential that we correctly predict the user's query intent. For the generative method, this aggressiveness results in lower performance (as seen in the third column), while the discriminative method is much more reliable and actually gains in performance. The interpolation method provides the best overall estimate of the user's query intent: when applied together with Model 2 it achieves good performance with high coverage, yielding the highest total gain of 0.0189, much higher than the next best method.

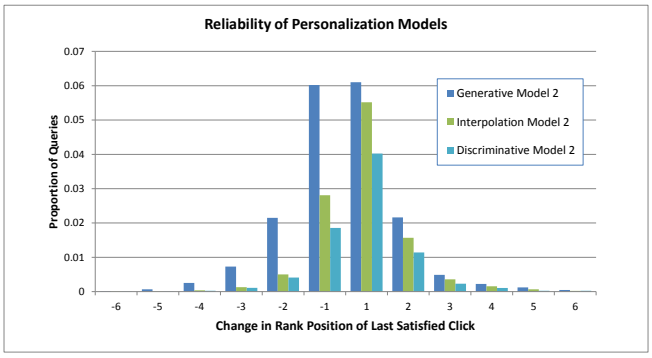

Figure 5: Histogram showing the variance of rank position gains and losses for personalization. The loss or gain in rank position of the last satisfied click is given on the x-axis. The y-axis denotes the fraction of queries in that bucket.

5.2.2 The risk of personalization

While achieving good average retrieval gains across queries is important, in the case of re-ranking an existing set of results, the *variance* of relative gains and losses compared to the initial ranking is also critical to measure. For example, two algorithms may appear identical with respect to their MRR performance, but one algorithm may have significantly higher variance in the magnitude of gains and losses it achieves compared to the initial ranking. Thus, we include a brief analysis of the variability in gains and losses.

Fig. 5 compares the distribution of gains and losses across queries using Model 2 for each of the three methods of predicting the user's query intent, $\Pr(T_u \mid \theta_u, q)$, as measured by the gain/loss in rank position of the last satisfied click. The x-axis shows the change in rank position of the last satisfied click, and the y-axis shows the fraction of queries with this change. Larger positive changes in rank positions are better. As shown in Table 2, the ensemble interpolation method reranks more frequently than the discriminative model though at a lower precision yielding a higher total gain. Fig. 5 shows that this gain results from having significantly fewer hurt queries (31%) than the generative model (50%) while being as reliable as the discriminative (30%) model. Thus, the ensemble effectively combines the greatly reduced downside risk of the discriminative model with the majority of upside gains achieved by the generative model.

Out of the total 102417 ambiguous, non-navigational queries in the test set, for the interpolation method, 11448 queries (11%) had a change in position of the relevant item, with 7881 (69%) of these queries helped – significantly higher than the null hypothesis of 50%.[4]

5.2.3 Filter Analysis

As research has indicated, when to personalize is as important as how to personalize [19]. Therefore, we evaluated the impact of the filtering conditions we used (non-navigational and ambiguous queries). Additionally, having noticed over the hold-out data that acronyms were a large class of ambiguous queries, we also explored acronyms as an alternative filter for identifying ambiguous queries.

[4]For a competitive baseline this is a conservative estimate; assuming a random method is equally likely to move the result to any position, there are far fewer positions on average above a result than below.

Table 3: Performance of filtering conditions on one word queries where the last satisfied clicked document moves position. MRR Δ is improvement over the baseline commercial search engine's ranking. Bold face indicates statistically significant ($p = 0.05$, Bonferroni correction) improvement over the baseline according to a sign test.

Filter Conditions	Set Size	Filter Set MRR Δ	MRR Δ
One Word	100.00%	**0.1213**	**0.1213**
One Word, Ambig.	68.21%	**0.1361**	**0.0928**
One Word, non-Nav	73.82%	**0.1442**	**0.1064**
One Word, Ambig., non-Nav	54.81%	**0.1686**	**0.0924**
Acronym	31.73%	**0.1745**	**0.0554**
Acronym, Ambig.	25.05%	**0.1988**	**0.0498**
Acronym, non-Nav	25.55%	**0.2052**	**0.0524**
Acronym, Ambig., non-Nav	21.08%	**0.2269**	**0.0478**

We therefore extracted acronyms from a bipartite graph of co-clicked queries [5] and clicked page titles. The queries and clicked pages were taken from Bing search logs, pruning edges with less than two clicks. We only used query pairs connected directly to co-clicked pages (i.e., a two-step walk); we also related queries with clicked page titles and vice versa. An acronym was added for each case-folded pair of queries and/or titles where one of these expanded to the other following these rules: expansions may contain additional stopwords (we used 114 common stopwords), acronyms only contain alphanumerical characters (0-9, a-z).

Table 3 presents the results of our filter analysis. For all results in this table, we consider only the set of queries where the last satisfied clicked document moves position, using Interpolation and Model 2. The first column shows the proportion of single word queries covered by the filter. The second column shows MRR on this subset of queries, and the final column shows the overall effect on this subset of one word queries (obtained by multiplying the first and second column). As the MRR gains in the table illustrate, filtering to both ambiguous and non-navigational queries identifies segments of queries that have larger potential for personalization (gain on this segment in "Filter Set MRR Δ" column) than either filter alone. Also, as can be seen in the table, filtering to acronyms alone yields a segment which has high potential for personalization and therefore is an interesting class for study. Adding the ambiguity or non-navigational filter increases the gains that personalization can yield by a large amount (up to 0.2269). However, since acronyms make up a small proportion of queries, the total gain seen (far right column) is lower than for other segments. The highest total gain was achieved over all one-word queries. Nonetheless, the use of filtering allows higher precision groups of queries to be identified, with acronyms providing one such high-potential segment for personalization.

6. CONCLUSION

We presented a general framework for personalization based on probabilistic models. We showed that we could achieve gains over a competitive baseline for one-word queries, and

especially for queries comprising acronyms, which represent a query segment that seems particularly amenable to our long-term personalization approach. Importantly, acronyms can easily be identified by practitioners who may lack access to search engine log data needed to identify ambiguous and non-navigational queries.

Although in this paper we specifically applied the framework to the problem of single topic-based personalization, the same ranking formula can be directly applied to a number of different types of personalization criteria such as multiple topics, geographic location, and reading proficiency. The ranking approach can also be used for personalizing based on short-term user profiles, by simply plugging in a different distribution for $\Pr(T_u \mid q, \theta_u)$.

Our approach can be extended in a number of directions. In this paper we describe personalization based on a single criterion using an ODP topic distribution. In future work, it would be interesting to explore personalization based on a variety of different criteria. Although the same probabilistic model can naively be used by simply using a larger state space for the user and document latent variables, it would be better to consider the structure underlying the various personalization criteria. We would likely need to resort to approximate inference methods to maintain low ranking latency. Learning user profiles also becomes more difficult in this setting, as we mentioned in Section 4.

The objective functions that we optimize to learn the user profiles are convex, making it straightforward to design online learning algorithms for the user profiles. This would give a simple update to use for θ_u after observing each new search query, and would guarantee low regret relative to the best possible θ_u chosen in hindsight. We can also consider smoothing across similar users, in contrast to our current approach which only uses a single user's data when learning the parameter vector θ_u.

Acknowledgements

Thanks to Dan Liebling and Josh Feng for assistance with data collection and classification.

References

[1] P. Bennett, K. Svore, and S. Dumais. Classification-enhanced ranking. In *WWW '10*, pages 111–120, 2010.

[2] H. Cao, D. Hu, D. Shen, D. Jiang, J.-T. Sun, E. Chen, and Q. Yang. Context-aware query classification. In *SIGIR '09*, pages 3–10, 2009.

[3] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware query suggestion by mining click-through and session data. In *KDD '08*, pages 875–883, 2008.

[4] P. Chirita, W. Nejdl, R. Paiu, and C. Kohlschutter. Using ODP metadata to personalize search. In *SIGIR '05*, pages 178–185, 2005.

[5] N. Craswell and M. Szummer. Random walks on the click graph. In *SIGIR '07*, pages 239–246, 2007.

[6] M. Daoud, L. Tamine-Lechani, M. Boughanem, and B. Chebaro. A session based personalized search using an ontological user profile. In *SOC '09*, 2009.

[7] Z. Dou, R. Song, and J. Wen. A large-scale evaluation and analysis of personalized search strategies. In *WWW '07*, pages 581–590, New York, NY, USA, 2007. ACM.

[8] J. Gao, W. Yuan, X. Li, K. Deng, and J.-Y. Nie. Smoothing clickthrough data for web search ranking. In *SIGIR '09*, pages 355–362, 2009.

[9] S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based user profiles for search and browsing. In *WIAS '03*, pages 219–234, 2003.

[10] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. http://cvxr.com/cvx, Apr. 2011.

[11] M. Jordan. Graphical models. *Statistical Science (Special Issue on Bayesian Statistics)*, 19:140–155, 2004.

[12] F. Liu, C. Yu, and W. Meng. Personalized web search for improving retrieval effectiveness. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):28–40, 2004.

[13] N. Matthijs and F. Radlinski. Personalizing web search using long term browsing history. In *Proceedings of the fourth ACM international conference on Web Search and Data Mining*, WSDM '11, pages 25–34, New York, NY, USA, 2011. ACM.

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

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

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

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

[18] J. Teevan, S. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *SIGIR '05*, pages 449–456, 2005.

[19] J. Teevan, S. Dumais, and E. Horvitz. Potential for personalization. *ACM TOCHI*, 17(1), 2010.

[20] J. Teevan, S. Dumais, and D. Liebling. To personalize or not to personalize: modeling queries with variation in user intent. In *SIGIR '08*, pages 163–170, 2008.

[21] K. Wang, T. Walker, and Z. Zheng. Pskip: Estimating relevant ranking quality from web search clickthrough data. In *SIGKDD '09*, pages 1355–1364, 2009.

[22] R. White and S. Drucker. Investigating behavioral variability in web search. In *WWW '07*, pages 21–30, 2007.

[23] R. W. White, P. N. Bennett, and S. T. Dumais. Predicting short-term interests using activity-based search context. In *CIKM '10*, pages 1009–1018, 2010.

[24] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *SIGIR '03*, pages 10–17, 2003.