# Cohort Modeling for Enhanced Personalized Search

Jinyun Yan
Rutgers University
Piscataway, NJ 08854 USA
jinyuny@cs.rutgers.edu

Wei Chu
Microsoft Bing
Bellevue WA 98004 USA
wechu@microsoft.com

Ryen W. White
Microsoft Research
Redmond, WA 98052 USA
ryenw@microsoft.com

## ABSTRACT

Web search engines utilize behavioral signals to develop search experiences tailored to individual users. To be effective, such personalization relies on access to sufficient information about each user's interests and intentions. For new users or new queries, profile information may be sparse or non-existent. To handle these cases, and perhaps also improve personalization for those with profiles, search engines can employ signals from users who are similar along one or more dimensions, i.e., those in the same *cohort*. In this paper we describe a characterization and evaluation of the use of such cohort modeling to enhance search personalization. We experiment with three pre-defined cohorts—topic, location, and top-level domain preference—independently and in combination, and also evaluate methods to learn cohorts dynamically. We show via extensive experimentation with large-scale logs from a commercial search engine that leveraging cohort behavior can yield significant relevance gains when combined with a production search engine ranking algorithm that uses similar classes of personalization signal but at the individual searcher level. Additional experiments show that our gains can be extended when we dynamically learn cohorts and target easily-identifiable classes of ambiguous or unseen queries.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *search process*, *selection process*, *clustering*.

## Keywords

Cohort modeling; Personalization; Web search.

## 1. INTRODUCTION

Personalization of search results has been investigated in detail in domains such as Web search and beyond [24][27][31]. The ability to tailor search results to a particular individual enables a wealth of opportunity to better satisfy their particular information needs. Personalization models are typically learned from observed short- and long-term search behavior (such as queries and result clicks), which is either used directly [32] or is converted into a different representation (e.g., a set of topical categories) to build more general models and improve personalization coverage [5][24]. Despite the value of personalization, one drawback is that it requires sufficient user information to perform effectively; users must be willing to share their search history and the search engine must attain sufficient information on user interests to build accurate profiles. Even short-term personalization depends on the long-term behavior for the first query in the session when no other activity has been observed [5].

It is known that users frequently submit the same query to find the information they have searched previously. Teevan et al. [29] found that approximately 33% of query instances in their case study were an exact repeat of a query submitted by the same user at a previous time. For such refinding queries, the results clicked by individual users in their search history provide a strong signal to identify the correct result for each user when that query is repeated [32]. The remaining 67% queries are new, and it can be challenging to improve their search quality via personalization given limited history.

One way in which these issues can be addressed is by finding *cohorts* of searchers who share one of more attributes with the current searcher. Attributes that could be used to form cohorts include location, topical interest, and domain preferences, all easily accessible to search engines via users' long-term search histories. Given a user, we can leverage the search behavior of other members of their cohort(s) to enhance personalization by providing signals if sufficient information is unavailable or as an additional signal to build richer personalization models if they already exist. Cohorts have been used effectively in applications such as collaborative filtering (CF) [12], where groups of similar users (based on factors such as liking the same item [19]) can yield relevant recommendations. Cohorts have also shown some limited utility in retrieval settings. Groupization has shown promise in laboratory settings [30], task models to find those engaged in similar tasks have yielded strong results [37], and there have even been attempts to use CF more directly in search result ranking [25]. However, there has been no detailed study of applying cohort models to enhance Web-scale search personalization. We address that shortcoming in this paper.

We propose the construction and application of user cohorts to enhance Web search personalization. Our initial method creates pre-defined cohorts on three types: topic, location, and top-level domain preference (e.g., .gov, .edu). Rather than limiting ourselves to these pre-defined sets, we also propose clustering methods capable of learning the cohorts and dynamically assigning users to one or more clusters. We demonstrate through extensive experimentation with search engine log data that our cohort modeling methods can yield significant relevance improvements over a production ranker that already included personalization targeting the current searcher. We show that these gains are even larger when we target particular queries (e.g., those with high ambiguity) and particular users (e.g., those with no query-relevant history).

We make the following contributions with this paper:

- Describe a method to generate pre-defined cohorts using information readily available to search engines, specifically topic, location, and top-level domain preference.
- Demonstrate that modeling user interests within these cohorts can enhance state-of-the-art search personalization methods, leading to significant gains in search relevance.
- Demonstrate that there are particular sets of easily-identifiable queries (e.g., ambiguous or new queries for a user), for which cohort modeling can be particularly effective.
- Propose methods to dynamically learn cohorts rather than using pre-defined sets, and demonstrate strong relevance gains.

The remainder of the paper is structured as follows. In Section 2 we present related work in areas such as personalization, collaborative

*SIGIR'14*, July 6–11, 2014, Gold Coast, Queensland, Australia.
Copyright © 2014 ACM 978-1-4503-2257-7/14/07…$15.00.
http://dx.doi.org/10.1145/2600428.2609617

filtering, and cohort identification and use. Section 3 describes the pre-defined cohorts and the modeling process, as well as various definitions of important attributes such as click-through rate and smoothing, as well as how we can apply clustering methods to learn cohorts dynamically rather than using a pre-defined set. Section 4 describes the datasets. Section 5 describes our methods and experimental results, and Section 6 reports the results of learned cohorts. Section 7 discusses our findings, their implications, and concludes.

## 2. RELATED WORK

There are three relevant areas of related work: (1) personalization of search engines based on short- and long-term searcher interests, (2) collaborative filtering, and (3) mining the search behavior of other users to complement and enhance search personalization.

Large-scale behavioral data from search engines has been mined extensively to improve result relevance in the aggregate across all users [1][15]. Search preferences are personal and research on personalizing retrieval [22][31] has shown that implicitly collected information such as browser history, query history, and desktop information, can be used to improve the relevance of search results for a particular individual. Short-term behavior from within the current search session has been used for tasks such as result ranking [39] or predicting future search interests [34][35]. Teevan et al. [31] showed that their personalization algorithm improved as more data was available about the current user. Long-term behavior has been used for personalizing search by constructing longitudinal models of user interests [24], including using the previous queries associated with the pursuit of similar information needs [27]. Models can use different sources, ranging from specific query-URL pairs which have high precision but low coverage [32] to more general methods that use topical representations of user search interests [24].

When there is insufficient data about the current user, the search behavior of other related users may be beneficial in modeling user interests and intentions. Teevan et al. [30] explored the similarity of query selection, desktop information, and explicit relevance judgments across a small group of work colleagues grouped along two dimensions: (1) the longevity of their personal relationship, and (2) how explicitly the group was formed. They found that some groupings provide insight into what members considered relevant to queries related to the group focus, but that it can be challenging to identify valuable groups implicitly. White et al. [37] address this issue by implicitly modeling the search *task* of the user, finding others who have attempted a similar task, and using their on-task behavior to enhance relevance. Although they used cohorts (location, topic expertise, and search engine entry point) as part of their ranking experiments, they observed limited gain in their experimental setting and how they chose to model and integrate cohorts.

Collaborative filtering (CF) [12] can also be used to find people with similar interests and leverage their activities and preferences to help the current user. The lack of sufficient personal information (sometimes referred to as the "cold start" problem) has been studied in research on CF and on recommender systems [20]. This research has shown that a number of sources can be used to generate recommendations from others in a given community, including agreement in item ratings [19] and social network memberships [16].

There are three predefined cohorts that we focus on in our study: topical interests, domain preferences, and geographic location. We now describe relevant related work in each area, beginning with topical interest, some of which leverages CF to find similar users.

Topic information can be used directly to improve search engine ranking [4]. Sugiyama et al. [25] addressed sparseness in user term-weight profiles by applying CF techniques to attain term weights based on those of users with similar profiles. Similar approaches have used click-through data to personalize result rankings and backed-off to the clicks of others [2][26]. Almeida and Almeida [2] used Bayesian algorithms to cluster users of an online bookstore into communities based on links clicked within the site and found that the popularity of links within different communities could be used to customize result rankings. Lee [17] proposed a system that uses data mining to uncover patterns in users' queries and browsing to generate recommendations for users with similar queries. These techniques perform matching with other users based on individual queries or URLs, severely limiting coverage. Freyne and Smyth [10] addressed this concern by connecting different communities based on the degree to which their queries and result clicks overlap.

Alternative methods have been proposed that are query independent. Smyth [23] suggested that click-through data from users in the same "search community" (e.g., a group of people who use a special-interest Web portal or work together) could enhance search. He provided evidence for the existence of search communities by showing that a group of co-workers had a higher query similarity threshold than general Web users. Ieong et al. [14] showed that searchers exhibited domain preferences, where they favored particular sources when selecting results. White et al. [38] found that domain experts preferred different top-level domains than novices, with more focus on educational (.edu) and governmental (.gov) sites, whereas novices preferred commercial (.com) sites.

Turning to location, Mei and Church [18] found that geographic location might serve as a reasonable proxy for community, since they observed that grouping users based on the IP address similarity could improve relevance. Cheng and Cantú-Paz [10] developed models for personalized click prediction in online advertising that leveraged demographic and location features to improve prediction accuracy. Bennett et al. [3] showed the effectiveness of location-based personalization, whereby models of searcher interests for particular locations can be learned and used in concert with the searcher's current location to improve relevance. White and Buscher [36] automatically identified users with local expertise (knowledge of a specific city or town) from search log data, and showed that the interests of these local users was both different and that there were differences in the quality of the entities they visited (restaurants reserved in this case), with locals selecting higher-rated venues. Weber and Castillo [33] estimated searcher demographics by joining search location with census data and demonstrated variations in search behavior for different demographic groups.

Our research extends previous work in the following ways. First we devise pre-defined searcher cohorts focused on attributes readily available at scale to Web search engines: specifically topic, location, and top-level domain preference. Second, we experiment with applying cohort models both in isolation and in combination to enhance personalization across all queries. Third, we analyze the performance of our methods in a number of additional search scenarios (e.g., ambiguous or unseen queries), and demonstrate strong relevance gains. Finally, we propose methods to learn cohorts via clustering, removing the need to use pre-defined sets. We show that employing this method allows us to further enhance personalization effectiveness over using our pre-defined cohort modeling methods.

## 3. COHORT MODELING

We now describe the construction of our cohort models, beginning with the nature of the data, but also including features computed.

Upon submission of a query to a search engine, a list of search results is retrieved and ranked for the user. The user examines the list and decides the next action on the results: click or not click. Search engine logs capture much of this interaction. In our study, we use logs sourced from the popular Microsoft Bing search engine. An entry in the search log comprises a tuple $< u, q, d, c, t >$, where $u$ is a user selected from the universe of users $U$, $q$ is a query in the universe set of queries $Q$, and $d$ is from the universe of documents (search results) $D$, $c$ is a binary value that 1 is a click and 0 otherwise, and $t$ is the timestamp. Such tuples can be created for each of the top-ranked search results returned by the search engine.

The click-through rate (CTR) of a query-document pair $< q, d >$ is the ratio of the number of clicks on the document to the number of impressions in which that result is shown for the search query. CTR is commonly used to measure the probability of a click given a query-document pair, i.e., $P(c = 1|d, q)$. Given this, plus the simplicity and general applicability of CTR, it seems appropriate to focus on applying it for this first study of cohort modeling.

In our approach we focus on a subset of result clicks suggesting that searchers are satisfied with the particular search results that they selected. We refer to these in this paper as satisfied (SAT) clicks.

**Definition 1 (SAT Click):** As defined in [12], SAT clicks have an associated dwell time of 30 or more seconds between search engine actions, or it is the last action in a search session (presumed SAT).

Using SAT clicks rather than all clicks can provide a more accurate CTR signal since accidental or misinformed clicks are excluded. Therefore, rather than simply counting the number of clicks divided by the number of impressions, we can compute CTR as:

$$ctr(d, q) = \frac{SATClicks(d, q)}{Impressions(d, q)} \tag{1}$$

Users may search for different information under the same query. This can also be reflected in a CTR tailored to each searcher. We use an individual's click-through rate $CTR(d, q, u)$ to estimate the degree of satisfaction of the user with a document given a query:

$$ctr(d, q, u) = \frac{SATClicks(d, q, u)}{Impressions(d, q, u)} \tag{2}$$

Armed with this important definition, we can now proceed to define the features that we use in our cohort models.

## 3.1 Contextual Features

As mentioned earlier, we represent users by contextual features corresponding to domain preference, location, and topical interests.

**Top-Level Domain:** The domain name of a URL represents its networking context, the administrative autonomy, and authority. A recent study showed that searchers exhibit a preference for particular domains irrespective of relevance [14]. The number of unique domain names across the broad range of information needs in our dataset is intractable. We therefore used the top-level domain (TLD). TLD includes generic domain extensions such as .com, .net, .org; sponsored extensions such as .mil, .asia, .edu; and country codes such as .us, .uk, .fr. Related work on domain expertise in search revealed that there were differences in the TLDs selected depending on user domain expertise level (experts preferred .edu and .gov, whereas novices preferred .com) [38]. The TLD may therefore offer some insight into the subject matter expertise of the searcher, which can be useful in performing richer personalization.

We limit our study on search logs collected in the United States geographic locale, but we observe many clicks on URLs with other country code domain extensions. This information could be used to estimate the native language of users or simply countries with which they have an interest. There are many TLDs, and because many are fairly new or visited infrequently (at least from search results), we do not observe many clicks related to them. We include all general and sponsored TLDs (23 in total), and also select 11 country code TLDs which are registered in the first and second year of availability, since we observe the number of Web pages and clicks of a TLD is related to its time in existence. We then randomly sample 3% search logs during a two-month period and examine the number of SAT clicks on selected TLDs. TLDs with < 1000 SAT clicks, e.g., .arpa, .post, .tel, are excluded. We retained the remaining 31 popular TLDs and use "other" for all other cases. This set of TLDs is used to construct our cohorts.

**Location:** The location of a user may also reveal their search interests and intentions [3]. We estimated the location of the user at query time using reverse IP geocoding. Since a user may not be confined to a particular city, but will generally remain within a state, we compute location preference for each user at the state level. There are 51 U.S. state features. When we failed to identify the location of a user, we categorize their location as "other".

**Topic:** We utilize the Open Directory Project (ODP, dmoz.org), a human-generated hierarchical taxonomy of Websites, as our topical ontology. This has been used extensively in previous work on personalization to model search interests at a level beyond queries and documents [5][24]. Topics are assigned to URLs using the content-based classifier described and evaluated in [4]. The user's degree of interest in a topic is then inferred from the number of clicks of URL results under that topic in their click history. ODP contains 15 top-level categories such as "Arts", "Sports", etc. To manage the size of the feature space, we focus on top-level categories only.

Given features explained above, we define a cohort as a group of users sharing a contextual feature. The total number of selected features is 99. In other words, we define 99 cohorts of users based on shared contextual features. A user can be a member of multiple cohorts. We model cohort membership to indicate how likely a user is to be a member. It is also used to measure how strongly to weight the user contribution to the cohort when aggregating cohort clicks.

We denote $C_j$ as the $j$-th cohort of a particular type, $C^T$ as cohorts of top-level domains, $C^L$ as cohorts of locations, and $C^O$ of ODP categories (topic). Since the following calculations for each of the three cohort types are the same, we ignore the superscript in the cohort notation for simplicity.

**Definition 2 (Cohort Membership):** The cohort membership vector for user $u$ is defined as a $m$-tuple $W(u) = [w(u, 1), w(u, 2), \ldots, w(u, m)]$, in which $m$ is the number of cohorts, and $w(u, i)$ represents the degree of membership for the user in $i$-th cohort (say, "California"). $W(u)$ is normalized such that $\sum_i w(u, C_i) = 1$.

The cohort membership is drawn from a multinomial distribution of SAT clicks, and calculated as follows:

$$w(u, C_j) = \frac{SATClicks(u, C_j) + 1}{\sum_i SATClicks(u, C_i) + K} \tag{3}$$

**Example 1:** Suppose that there are only three cohorts: California, Washington, and Oregon. If we observe three SAT clicks when the user is in California and one SAT click in Washington, the cohort membership across the three states would be [0.57, 0.29, 0.14].

## 3.2 Cohort CTR

**Definition 3 (Cohort CTR):** Given a cohort type (e.g., Topic), the cohort CTR for a query and URL document $< q, d >$ is a $m$-tuple *c-ctr(d,q)*=$[ctr(d, q, C_1), ctr(d, q, C_2), \ldots, ctr(d, q, C_m)]$, in which $m$ is the number of cohorts, and $ctr(d, q, C_i)$ is the probability that users in $i$-th cohort will click document $d$ for the query $q$. It is a weighted aggregation of individual CTR as follows:

$$ctr\big(d,q,C_j\big) = \frac{\sum_u SATClicks(d,q,u)\cdot w(u,C_j)}{\sum_u Impressions(d,q,u)\cdot w(u,C_j)} \tag{4}$$

Cohort CTR is used to measure the cohort preference on the document $d$ given the query $q$. It weights a user's clicks by their cohort membership. Users who exhibit strong preference to the cohort will contribute more to the cohort CTR, e.g., for a California state cohort, a user residing in that state for a long duration will have a higher influence factor than a user who only visits occasionally.

**Example 2:** Suppose that there are only three cohorts: California, Washington and Oregon, and two users *a* and *b*. The cohort membership vector for $a$ is $W(a)$ = [0.57, 0.29, 0.14], for $b$ is $W(b)$ = [0.1, 0.1, 0.8]. Given the query [osu], considering two search results $d_1$= "osu.ppy.sh", and $d_2$ = "oregonstate.edu", the number of SAT clicks by user $a$ is $S(a)$ = [5, 1] for $d_1$ and $d_2$ respectively, and the number of SAT clicks by user $b$ is $S(b)$ = [1, 5]. For simplicity, we assume number of impressions on each document for each user is 100. By Equation (4), we can compute the cohort CTR for the result $d_1$ as *c-ctr*($d_1$, $q$) = [0.044, 0.039, 0.016], and for $d_2$ as *c-ctr*($d_2$, $q$) = [0.0159, 0.02, 0.044]. This demonstrates that the California cohort prefers the result $d_1$, and that the Oregon cohort prefers $d_2$, given the query [osu]. Note that the global CTRs for both results are the same, i.e., *ctr*($d_1$, $q$) = *ctr*($d_2$, $q$) = (5+1)/(100+100).

There are two intuitions behind our model. First, users in a cohort with shared contextual features are likely to be coherent in search intentions and click preferences (an assertion supported by our preliminary investigations – not reported here for space reasons). Second, a common approach for handling the problem of insufficient individual historical data is to leverage global CTR. However, global CTR treats clicks from all users equally, and therefore has limited potential to help in personalization. Our approach identifies and separates cohort clicks from global clicks. When estimating an individual's click preference, we can learn more from clicks by cohorts of similar users, who have higher impact on the estimation, and are better aligned with the target user. We show in our later experiments that cohort modeling can outperform global CTR.

## 3.3 Smoothing Cohort CTR

CTR is one of the most informative metrics to measure search result quality. However, CTR estimates are sometimes noisy when observations are scarce. For example, if we only observe one impression for a pair $< d, q >$, and a single SAT click on the document $d$, we will obtain $ctr(d,q) = 1$. This is an inaccurate estimate of the true click probability, and is caused by data sparseness. These instances are common in logs, especially for tail queries that occur rarely.

To handle this situation, we apply smoothing methods to estimate CTR. We add a pseudo count that counts SAT clicks $\alpha \cdot N$ times during $N$ impressions. The smoothed CTR is computed as follows.

$$\widehat{ctr}(d,q) = \frac{SATClicks(d,q) + \alpha\cdot N}{Impressions(d,q) + N} \tag{5}$$

After smoothing, extreme cases should have lower CTR than URLs with sufficient SAT clicks and impressions, but higher than those with no SAT clicks. Based on this expectation, we sample hundreds of instances and manually validate the output to tune $\alpha$ and $N$. Following several experiments we set $\alpha$ = 0.001, and $N$ = 1000.

When calculating cohort CTR for a given cohort $C_j$, we then smooth cohort CTR with smoothed global CTR as follows:

$$\widehat{ctr}\big(d,q,C_j\big) = \frac{N\cdot\widehat{ctr}(d,q) + \sum_u SATClicks(d,q,u)\cdot w(u,C_j)}{N + \sum_u Impressions(d,q,u)\cdot w\big(u,C_j\big)} \tag{6}$$

We set $N$=10 through similar manual validation. For unobserved or scarcely observed $< d, q, C_j >$, the cohort CTR is aligned with a smoothed global CTR of $< d, q\ \ >$.

## 3.4 Cohort Features

For a user, the click probability on a URL document can be estimated from the click history of similar people. Given the cohort model, we now derive cohort features, which infer individual click probabilities that are associated with the user's cohort membership.

**Definition 4 (Cohort Features):** Consider a user $u$ with cohort membership $W(u)$, and the cohort CTR for a query document pair *c-ctr*(*d*,*q*), we derive cohort features $Z(d,q,u)$ as an $m$-tuple: $[\ z(d,q,u,C_1), z(d,q,u,C_2), \ldots, z(d,q,u,C_m)\ ]$, where $m$ is the number of cohorts, and $z\big(d,q,u,C_j\big)$ is the click probability in the $j$-th cohort. The probability is computed as follows:

$$z\big(d,q,u,C_j\big) = w\big(u,C_j\big)\cdot\widehat{ctr}\big(d,q,C_j\big) \tag{7}$$

**Example 3:** Following the setting in Example 2, that *c-ctr*($d1$,$q$) = [0.044, 0.039, 0.016], given a new user with $W(c)$ = [0.56, 0.22, 0.22], the cohort features for document $d_1$, query $q$ which is [osu] and the user $c$, is $z(d_1,q,c)$ = [0.02464, 0.00858, 0.00352].

When a user submits a query $q$, we estimate their click preference on a document $d$ depending on their cohort membership $w\big(u,C_j\big)$ and the cohort click probability $\widehat{ctr}\big(d,q,C_j\big)$. The weight of cohort membership controls how much we can infer about this user's click behavior based on a cohort's click behavior. If a user belongs to the California cohort with a weight of 0.9 and the Washington cohort with a weight of 0.1, the estimation of their click probability therefore relies mainly on the cohort California, and only slightly on the cohort Washington. We create cohort features for each $< d, q, u >$ tuple, and let the ranking algorithm decide the ranking of URL candidates based on these cohort signals.

# 4. DATASETS

To evaluate the effectiveness of our cohort model for enhancing personalization, we apply it to extend a personalization model on the Microsoft Bing commercial search engine. The existing personalization approach is built upon the standard search engine that retrieves the most relevant documents via querying. This is a state-of-the-art personalization method that employs a number of short- and long-term topical, location, and domain preference features, some that are similar to prior work, e.g., [3][5][24][32][35]. These features are derived and used by the engine at the individual level. In addition to these personalized features, the model uses a global CTR feature for each query and document pair. This ranker in production serves as a strong baseline for our cohort experiments.

We evaluate our methods retrospectively using logs from Bing containing search behavior and the original (sometimes personalized) result ranking from the engine. We mined over two months of logs from the US English geographic locale, and extracted events comprising tuples of: query, an ordered list of the top-10 search results returned by the engine, and clicks on those results. The order of the URLs for a query was produced by the baseline ranker which employed personalization for some queries as described above. We re-ranked results using our enhanced model. This methodology allows us to estimate the effectiveness of our cohort modeling approach.

## 4.1 All Queries

Cohort features, which are based on click history, are good indicators of document relevance for given queries. However, the volume of URL documents is large, and many documents are not selected or displayed many users. Although we incorporated smoothing techniques to overcome such sparseness, we found in practice that

**Table 1. Data sets used in experiments. All dates from 2013.**

| Data | Cohort Profiling | Training and Validation | Testing |
|---|---|---|---|
| Date range | 03/31–05/28 | 05/29–06/02 | 06/03–06/04 |
| #impressions | 1,016,333,942 | 11,615,957 | 5,352,460 |
| #distinct queries | 248,419,356 | 4,096,337 | 2,192,327 |
| #distinct domains | 25,704,086 | 3,116,209 | 2,087,303 |
| #users | 23,378,476 | 1,144,715 | 739,281 |

constructing features at a higher level further addresses this challenge. For instance, we can replace URL documents in our cohort models by URL domains and re-rank using the same personalization approach. Specifically, the symbol $d$ is used to represent a URL domain rather than a URL document. A domain is part of a URL, e.g., URL=http://www.cnn.com/politics, domain=cnn.com.

As stated above, we use the production ranker from the search engine as the baseline for comparison. We then train a new model, with cohort features added. Bing search logs for a two-month period (March 31 2013 to May 28 2013) are used to construct cohort membership vectors and cohort CTRs. We refer to this time segment as the *profiling period*. Cohort features are then built for $< u, q, d >$ tuples in logs from the following week (May 29 2013 to June 4 2013), which is then divided into *training*, *validation* and *testing* periods. The first three days were used for training, the next two days were used for validation, and the last two days were used for testing. The performance is evaluated by re-ranking top results returned from the baseline ranker. This method has been used successfully in prior studies of search personalization at scale [3][24].

Table 1 presents the statistics on the datasets used, including the number of search queries (impressions), the number of distinct queries, the number of distinct URL domains, and the number of users in our dataset. Besides the comparison on all queries, we also classified queries into various segments to facilitate a more detailed analysis of the performance of our cohort modeling methods.

## 4.2 New Queries in User Search History

Previous studies have shown that although searchers frequently submit repeated queries for refinding purposes, there are also a large fraction of user queries that are new [21][29]. A new query from a particular user means by definition that user has not submitted it previously (at least not in an observable period, such as the two months used for profile building). Given their frequency, new queries are a particular subset where search engines could offer significant benefit, but since there is no user history it is not clear what support they can offer on an individual level. This means that they must resort to global models of all users' on-query behavior. These are queries where cohort modeling may offer particular assistance.

**Definition 5 (New Queries):** In our experiment, for each user, queries that are shown in the testing period but not in the profiling, training and validation periods are defined as *new* queries. In contrast, queries that appear in all periods are defined as *old* queries.

In our analysis, we identify new queries for each user and separate them from old queries to evaluate the re-ranking performance of cohort models. To simplify the determination of new queries, we focus on exact match of queries on training and testing periods. The derivation and application of more sophisticated matching methods (e.g., semantically-equivalent queries), is a separate research problem and is reserved for future work. Some preprocessing steps are applied, including converting queries to lowercase, removing surplus whitespace, and deleting punctuation while preserving the n-grams for terms joined by punctuation (e.g., asp.net).

## 4.3 Popularity of Queries

Our cohort model leverages group click preferences to estimate individual click preferences. For a user who submitted a query, we identified a cohort of other users who are similar. However, if only a small number of users in the group submitted the same query, the prediction of cohort preference for the query will be biased and not representative of the full cohort. As part of our re-ranking experiments, we wanted to better understand the impact of query popularity on re-ranking performance when cohorts were utilized.

**Definition 6 (Popular Queries):** The popularity of a query is determined by the number of distinct users who submitted the query during the profiling period, which is denoted by $N_u$. Search has a long tail effect that many tail queries are submitted only one or two times, by a small number of users. Cumulatively, there are a large number of such queries. We divided queries into two datasets: (1) *popular*: $N_u \geq 10$, and (2) *unpopular*: $N_u < 10$. Approximately 30% of distinct queries are popular per our definition.

## 4.4 Query Entropy

As mentioned earlier, some queries have almost uniform click preference among all users, for example, [facebook] or [amazon] have high CTR on their associated sites. For these cases, individual, cohort, and global preferences are consistent. Thus the cohort model has limited potential to improve retrieval performance for such queries. We measure the diversity of clicks among users for each query by computing the *query entropy* as follows:

$$H(q) = -\sum_{d} \frac{\widehat{ctr}(d,q)}{\sum_d \widehat{ctr}(d,q)} \log\left(\frac{\widehat{ctr}(d,q)}{\sum_d \widehat{ctr}(d,q)}\right) \tag{8}$$

where $\widehat{ctr}(d,q)$is from Equation (5).

We focus on top five URL domains returned as search results ordered by global CTR. As a result, the maximum entropy value is $log(5) \sim 1.6$, and the minimum is zero. If clicks of all users led to the same destination, the value of the entropy will be zero, indicating the query has the smallest variation in click behavior. A high value of entropy indicates the query has large variations. Click entropy has been used in many studies to evaluate the complexity of queries, e.g., [16]. However, by assuming that the same search results are shown to all users who submitted the same query, its implementation in those studies only considers the number of clicks and ignores the impression counts. Our data comprises logs of a search engine equipped with personalization. Consequently, URL documents have unequal chance of being shown. Therefore we take advantage of CTR and consider both clicks and impressions.

To examine how the performance of our cohort model relates to the level of query entropy, we separate queries into three subsets: *low entropy*, *medium entropy* and *high entropy*. The corresponding entropy ranges are [0, 0.2), [0.2, 1.2), and [1.2, 1.6). The motivation is that for queries with small entropy on global CTR, it is less likely that cohort click preference differs from global click preference. For queries with large entropy, global clicks are diverse, thus we expect that cohorts can differentiate clicks, and therefore offer better personalized search results.

## 4.5 Acronym Queries

Many acronyms are ambiguous and associate with more than one meanings. For example the intent behind [msg] may differ depending on the user location, e.g., users in New York City may be more likely to mean Madison Square Garden, whereas the likely intent elsewhere in the United States could be monosodium glutamate. As such, search engine performance can be improved on acronym queries via personalization that considers the location of the searcher as part of the ranking process [24]. To understand the effect of acronym queries on the performance of our models, we used a set of acronyms defined in previous work [28]. From these data, we selected 432,564 acronyms which had a length of 2, 3 and 4 characters. The average number of meanings per acronym was 2.91. We then intersected these with the two days of logs used for testing in our study, resulting in around 11,000 distinct query matches.

# 5. METHODS AND FINDINGS

We now describe our experimental results. As mentioned earlier, our baseline is the current production ranker in the commercial search engine .Our cohort model extends it by integrating cohort features. Comparing the models let us estimate changes in personalization effectiveness attributable to the cohort modeling.

## 5.1 Ranking Models

Using the dataset described in the previous section, we train a LambdaMART-based ranking model [39] to re-ranking the top ten search results. LambdaMART is an extension of LambdaRank [8] which is based on boosted decision trees. It has been shown to be one of the best algorithms for learning to rank. Indeed, an ensemble model in which LambdaMART rankers were the key component won Track 1 of the 2010 Yahoo! *Learning to Rank Challenge* [9]. Our cohort features are insensitive to ranking algorithm, thus any reasonable learning-to-rank algorithm should also observe relevance gains as we do. We trained four ranking models using three types of predefined cohort features introduced earlier, specifically:

1. A model with ODP cohorts only (ODP);
2. A model with top-level domain cohorts only (TLD);
3. A model with location cohorts only (Location), and;
4. A model with all three cohorts, concatenated together (ALL).

We also construct cohort models dynamically by clustering users based on contextual features. In Section 6, we describe the cohort clustering methods and experiment with varying the number of clusters (cohorts), denoted as $k$ in the remainder of this paper.

## 5.2 Metrics

As described earlier, we collected two months of search logs to construct user profiles, and the next one week of logs for training, validation, and test. Evaluating personalization at scale is challenging; since users can have different intentions for the same query, employing third-party relevance labels may be insufficient. To address this concern, we exploit user clicks to obtain personalized relevance judgments for each query-document pair retrieved for a query. Clicks can be classified into various types by their associated dwell times on the landing page. If the dwell time is too short, the searcher may be dissatisfied with the search result. In this study, we label URLs with a SAT click (defined earlier) positively, and other URLs negatively. This method for generating click-based relevance judgments has been used in prior personalization studies [5][24][37].

We measure the quality of re-ranking using mean reciprocal rank (MRR) and mean average precision (MAP). In both cases, the mean is the average across all impressions, including those where the ranking does not change as a result of the treatment. MAP considers cases where there are multiple SAT clicks (better for informational queries); MRR is focusing only on the rank of the first SAT click.

MAP is the mean of the average precision scores for each query,

$$MAP = \frac{1}{N}\frac{\sum_{i=1}^{n} Precision(i)Rel(i)}{\sum_{k=1}^{n} Rel(i)} \qquad (9)$$

where $n$ is the number of URLs in the impression, ranging from 4 to 10. $Rel(i)$ is an indicator function returning 1 if the URL at rank $i$ is relevant, otherwise 0. $Precision(i)$ is the precision at cut-off $i$ in the ranked list.

**Table 2. Gains in MAP and MRR over baseline (±SEM).**

| Cohort | Rerank@1 | ΔMAP±SEM | ΔMRR±SEM |
|---|---|---|---|
| ODP | 0.91% | 0.0181±0.00130 | 0.0187±0.00142 |
| TLD | 0.96% | 0.0224±0.00140 | 0.0229±0.00144 |
| Location | 0.90% | 0.0111±0.00138 | 0.0113±0.00141 |
| ALL | 0.98% | 0.0193±0.00140 | 0.0211±0.00145 |

MRR targets the rank of the first relevant document in the result list. It is the average of the reciprocal ranks over all queries,

$$MRR \ = \frac{1}{N}\sum_{q}\frac{1}{rank(q)} \qquad (10)$$

where $rank(q)$ is the rank position of the first URL document that received satisfied click for the query $q$.

Due to proprietary concerns, we do not report absolute metric values. Instead, we report relative changes from the cohort model versus the baseline: $\Delta MRR \ = 100 \cdot (MRR(cohort) - MRR(base))$ and $\Delta MAP = 100 \cdot (MAP(cohort) - MAP(base)$.

## 5.3 Research Questions

To understand the effect of cohorts in personalized search, we answer the following questions in the remainder of the paper:

1. Can our method enhance the baseline generally, for all queries?
2. Can we identify particular classes of queries that benefit from our cohort modeling, and to what extent?
3. Can we improve relevance further by learning cohorts (rather than the pre-defined cohorts defined earlier)? (See Section 6).

We now present the results of our analysis on all queries and on each of the query subsets described in the previous section.

## 5.4 Findings

We now present the findings of our study, grouped by dataset.

### *5.4.1 All Queries*

We begin with the first question: in general, can cohort models improve the retrieval performance when used in addition to existing personalization method(s)? This helps us understand the overall impact of the cohort modeling on search engine performance. Table 2 reports the MAP/MRR gains of our model versus the baseline (the production ranker) along with the standard error of the mean (SEM). The findings presented in the table show our cohort model significantly outperforms the baseline (with paired t-tests). Results that received SAT clicks by users are promoted by the cohort by the ranking (as can be seen with the low reranked@1 percentage in Table 2). All types of cohorts are informative. In particular, TLD yields the largest gain, perhaps because it captures differences in expertise of interests (e.g., people selecting en.wikipedia.org rather than a commercial domain). Location cohorts may have achieved the lowest gain because firstly, the baseline already covered the individual location preference; secondly we use state to represent location and it could mask important intra-state movements. A finer grained representation of location may be required, but we also need to consider how best to do that in a scalable manner while ensuring that there are sufficient numbers of users in each cohort. Note that although the changes may appear small, they are averaged over all queries, including many whose performance is unchanged.

A trend that we observe in Table 2 that is mirrored in all of our findings is that ALL performs as well or less well than the other models. This model re-ranks slightly more results (Rerank@1 in Table 2), meaning that its application is less focused. Also, the presence of multiple cohorts may make the ALL cohort signal noisier.

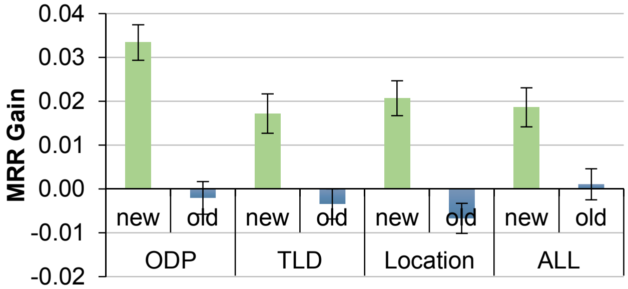

**Figure 1. Gains in MRR over baseline for each cohort type for new and old queries from each user (±SEM).**

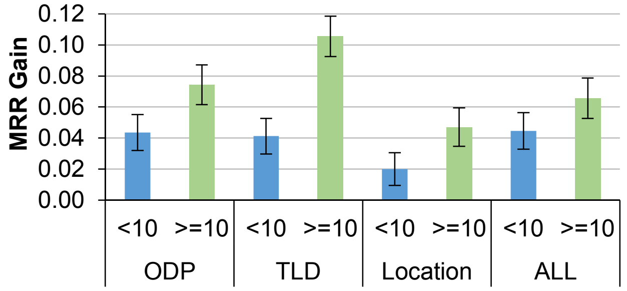

**Figure 2. Gains in MRR over baseline for each cohort type for differences in the popularity of the query (±SEM).**

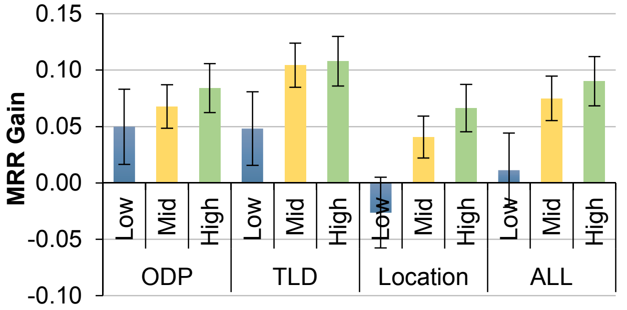

**Figure 3. Gains in MRR over baseline for each cohort type for different query entropy bins (±SEM).**

Given the promising gains observed across all queries, we now turn our attention to the various query subsets that were introduced earlier in the paper. In the remainder of this section we present results on each of those subsets defined in Section 4. Since the performance of both of the MRR and MAP metrics is similar, we focus on a single metric (MRR) for the remaining analysis.

### *5.4.2 New Queries*

In our dataset, the average ratio of distinct new queries of all queries is about 70% per user, consistent with previous work [25]. It indicates that users submit a large portion of new queries that are not recorded by the search engine previously (at least not in the past two months, which may be all the engine has access to for a user at query time given profile size limitations at scale). Thus personalization based solely on an individual history is insufficient.

Our cohort model utilizes search and click history of similar users to alleviate the challenge of insufficient data. We split the testing data into two subsets composed of old queries and new queries for each user respectively. Figure 1 shows the performance difference over the baseline for each type of cohorts. In this figure and others in this section, the value of zero denotes the original performance of the baseline. The figure shows that indeed our model works well on new queries that have not been observed previously (at least in the profiling period) from a given user. We observe statistically significant gains for new queries across all predefined cohorts (all $p <$ 0.001). When queries are repeated, the baseline with individual search history may work well, and adding cohort features had little or even slightly negative effect by introducing noise from other searchers' activity (as is evidenced by the blue bars and negative MRR changes). It is also interesting to observe that the ODP (topic) cohort performed best for new queries. One possible explanation for this finding is that queries without an exact match that appear in the users' history are most likely to be informational, and therefore benefit most from users with similar topical interests.

**Table 3. Gains in MAP and MRR over baseline for acronym queries (±SEM).**

| Cohort | ΔMAP±SEM | ΔMRR±SEM |
|---|---|---|
| ODP | 0.1566±0.0562 | 0.1622±0.0568 |
| TLD | 0.1585±0.0568 | 0.1519±0.0578 |
| Location | 0.1450±0.0535 | 0.1552±0.0544 |
| ALL | 0.1212±0.0544 | 0.1265±0.0553 |

### *5.4.3 Query Popularity*

We are also interested in the effect of query popularity on the performance of the cohort modeling, in order to understand how sensitive our model to the size of cohorts. Figure 2 shows MRR gains on the popular and unpopular query sets, as described earlier. The performance gain on popular set is much larger than that in unpopular set. Again all differences are significant given the extent of the gains and large sample sizes ($p < 0.001$). The results match our expectation. When a query is searched by many users, we can distinguish cohort preference accurately. However, if a query is searched by only few people, the estimation is less accurate.

### *5.4.4 Query Entropy*

We conjectured that since a large entropy implies diverse clicks on URLs, separating and assigning weights on clicks by cohorts can help identify an individual's preference more accurately. Therefore we expect queries with large entropy will obtain large benefit from the cohort model. Figure 3 presents the MRR gain over the baseline for the three query entropy bins: low, medium, and high.

The results shown in the figure confirm our intuition regarding where personalization enhancements might help. On all types of cohorts, the query set with low entropy received smallest gain over the baseline. Queries with medium and high entropy obtained larger performance increases (all statistically significant, $p < 0.001$). This suggests that one strategy to realize strong gains from the cohorts may be to bypass low entropy queries and only apply cohort models on queries with medium or higher entropy. As observed in other analyses in this section, we also observe that the Location cohorts achieved the smallest gain, and even resulted in a loss for low entropy queries. As mentioned earlier, one explanation is the use of state-level cohort features, which may be too coarse to capture individual click preferences. More work is required to determine how best to represent and apply location for cohort modeling.

### *5.4.5 Acronym Queries*

As mentioned earlier, acronym queries such as [acl], [atm], etc. are a specific set of ambiguous queries where personalization may help [24]. We examine the effectiveness of our cohort modeling methods on the subset of acronym queries described earlier in Section 4. Table 3 shows the MAP and MRR gains over the baseline for this query set (all significant at $p < 0.001$). The results clearly demonstrate extremely strong gains in performance for the subset of acronym queries for each of the cohort types studied. Although this may only be a relatively small query set (around 11k distinct queries), it is encouraging to see the significant gains in acronym queries.

It is clear from the findings presented in the section so far that there are a broad range of different query classes for which the cohort modeling performs well. However, the performance of the ALL model was generally slightly lower than the other models. There may be a better way to combine the cohorts and in the next section we describe an approach to learn cohorts dynamically.

# 6. LEARNED COHORT MODELS

The results in the previous section show that our cohort modeling techniques using *pre-defined* features can more accurately estimate users' individual click preferences (as represented via an increased number of SAT clicks) than our competitive baseline method. A challenge of this approach is the tradeoff between the number of cohorts and the predictive power of cohorts on individuals. One can define more granular cohorts, for instance, including second or even lower levels of ODP, and changing locations from state to city or even the ZIP-code level. However, more cohorts result in fewer users in one cohort and less reliable CTR estimation. To overcome this challenge, we propose an alternative that generates cohorts automatically via clustering. The objective is to construct homogeneous clusters (cohorts) given a large number of features.

## 6.1 Clustering Method

In this section, we discuss how we learn cohorts automatically using $k$-means clustering. Each user is represented by a vector of contextual features $x_u \in R^d$, which is concatenated from the three sets of pre-defined cohort features on topic, location and top level domain. The dimension of the feature vector is 99 in our setting. The objective of the method is to assign users into cohorts. Given large data volumes, a map-reduce implementation of $k$-means algorithm is applied to cluster users into $k$ clusters (cohorts). We then define two implementations of customized cohort membership vector.

**Definition 7 (Learned Membership, Hard):** Given learned $k$-cohorts, a particular user's cohort membership vector is defined as a $k$-tuple $W(u) = [w(u,1), w(u,2), \ldots, w(u,k)]$. Membership in the $i$-th cohort depends on whether the user is assigned to the $i$-th cluster. That is, $w(u,i) = 1$ if the user is in the $i$-th cluster, otherwise $w(u,i) = 0$.

**Definition 8 (Learned Membership, Soft):** Given learned k-cohorts, a particular user's cohort membership vector is defined as a $k$-tuple $W(u) = [w(u,1), w(u,2), \ldots, w(u,k)]$. The membership to $i$-th cohort is determined by the minimum Euclidean distance between the user and the centroid. Let centroids learned by $k$-means be $\{\mu_1, \mu_2, \ldots, \mu_K\}$. Ideally the Gaussian Mixture Model could achieve the goal with additional computational overhead. In this large-scale study, we leverage the $k$-means results and assign cluster membership as follows:

$$w(u, C_j) = p(C_j|x_u) = \frac{\exp(-\frac{1}{\alpha^2} d(x_u, \mu_j)^2)}{\sum_{\{i=1\}}^{K} exp(-\frac{1}{\alpha^2} d(x_u, \mu_i)^2)} \quad (11)$$

where $d(x_u, \mu_j)$ is Euclidean distance between the user vector $x_u$ and the centroid $\mu_j$, and $\alpha$ is estimated from the average distance between centroids. This is a simplified implementation of the Gaussian Mixture Model having identity covariance.

With the hard membership assignment, each user has only one nonzero cohort membership, which may be preferable on many clusters with large $k$. For users with diverse preferences, it is natural to allow multiple cluster membership. Therefore soft membership may produce higher performance gain since it is capable of better capturing within-user variance in interests and intentions.

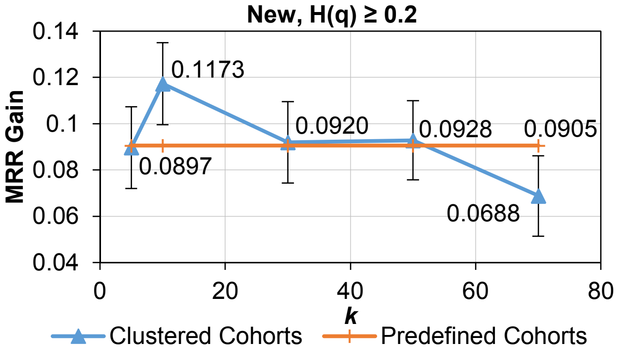

**Figure 4. Gains in MRR over the baseline for clustered cohorts versus pre-defined cohorts for different $k$. Note that this is for the New, $H(q) \geq 0.2$ query set (±SEM).**

## 6.2 Evaluating Clustered Cohorts

We compare the performance of re-ranking by clustered cohorts against the model with predefined ALL cohorts. We evaluate MRR change on a selected subset of queries. The subset of queries is new queries with entropy larger or equal to 0.2. This is set on which we observed a large performance gain with predefined cohort model, so it was a competitive dataset on which to assess the learned cohorts. We experiment learned cohorts using select probes of $k \in \{5, 10, 30, 50, 70\}$, with soft assignment of cohort membership, which is expected to be more performant. Figure 4 displays the experimental results at varying value of $k$. Error bars denote standard error of the mean. Note that the baseline ranker (where MRR gain = 0) already contained global CTR as a feature, which is equivalent to $k$=1. We therefore do not report performance at $k$=1 in Figure 4.

The figure shows that we can observe the largest MRR gain by the clustered cohorts model when $k$=10. The MRR gains are slightly larger than the model with predefined cohorts in $k$=30, 50, and slightly smaller in $k$=5. The MRR gain decreases sharply when $k$ becomes too large, e.g., 70. We did not perform a full sweep of $k$ given resource constraints, but the findings are still informative and the gains over predefined at $k$=10 are significant ($p < 0.001$). There may be a region $10 < k < 30$ where we may realize larger gains and we will explore that region in more detail in future work.

We also evaluated the model on other subsets of queries and observed similar results: the largest gain is obtained in small $k$ and largest $k$ has smaller gain than the predefined cohorts model. One possible explanation for this is that the user features are sparse and that as $k$ increases, the reliability the cohort signal in each cluster degrades. The fact that we can obtain strong performance by reducing the dimensionality of the features from 99 to 10 is promising for large-scale deployment (since it means compact user profiles). It also reveals the opportunity of profiling users with more subtle and sparse features than projecting to a few principal dimensions, as was done in the case of the pre-defined cohorts.

As mentioned previously, we can employ either hard or soft clustering, depending on whether we want users to reside within a single cohort only (hard) or appear in multiple cohorts potentially with different weights (soft). The implications of this include the nature of the profile stored by a search engine. In the analysis above, we employed soft clustering. One concern we had regarding hard clustering is that it may lead to an inaccurate CTR estimation for users who are far from the cluster centroid. To better understand the impact of this decision, we compare the performance of models with cohorts by hard-clustering, soft-clustering and predefined features

**Table 4. Gains in MAP and MRR over baseline for different clustering methods (hard ($k$=10) vs. soft) and vs. pre-defined.**

| Metric | Hard Membership | Soft Membership | Predefined Cohorts |
|---|---|---|---|
| ΔMAP | 0.0731±0.0158 | 0.1143±0.0170 | 0.0932±0.0172 |
| ΔMRR | 0.0737±0.0165 | 0.1173±0.0177 | 0.0905±0.0180 |

w.r.t. the baseline as in the other experiments presented in the paper thus far. Table 4 presents the findings of this analysis.

Table 4 shows that, as expected, cohorts generated using hard membership achieved the smallest performance gain, and are worse than those from predefined cohorts. Soft membership performs significantly better; other differences are not significant. This suggest that finding weights to assign to each cohort is important for estimating individual preference. Users also have variations inside a cohort, and their preferences cannot simply be generalized by one cohort.

## 6.3 Preference Analysis

Given that we have these different ways to identify cohorts, we were interested in understanding the relationship between existing search engine results and global/cohort preference. To show that our performance improvement on personalization is not simply caused by gathering more features for the ranking algorithm, we conduct analysis on search logs and investigate whether cohorts manifest unique preference, which is directed by users in the cohort. For a query with many candidate results, global CTR can offer a ranking of URL candidates. We refer to this here as *global choice*. In each identified cohort, cohort CTR can yield a ranking as well, and we refer to this as *cohort choice*. We focus on the difference between global choice and cohort choice of the top-ranked result.

**Definition 9 (DiffTop):** DiffTop for query $q$ is an $m$-tuple vector $D(q) = [diff(q, C_1), diff(q, C_2), \ldots, diff(q, C_m)]$. Each value is a binary value to indicate whether a cohort has unique preference. We denote the top ranked URL domain $d$ by the cohort choice of $j$-th cohort as $d_{C_j}$, and by global choice as $d_G$. If $d_{C_j} \neq d_G$, we set $diff(q, C_j) = 1$, otherwise 0.

Among selected logs in profiling period, we choose queries with at least two distinct URL domains clicked, and count how many are inconsistent in the cohort choice and global choice. We find that 2% of distinct queries demonstrated unique preference by at least one cohort. The ratio appears small, but considering the query volume is large, and the fact that we focus on clicks on domain level in the top position only, it is still a strong signal of cohort potential.

There are cases that the values of cohort CTR for the top and the second top URL domains are very similar, e.g., equally small. This means that the top and the second top URL domains have similar cohort preference. To address such subtle scenarios, we defined a *weighted* DiffTop measure as follows.

**Definition 10 (Weighted DiffTop):** The weighted DiffTop for query $q$ is a $m$-tuple $w\_D(q) = [w\_diff(q, C_1), w\_diff(q, C_2), \ldots, w\_diff(q, C_m)]$. Each value measures the degree of unique preference by the cohort. We define the decrease delta (Δ) to measure the difference in the click probability between the top and the second top URL domain as follows:

$$\Delta(d_1, d_2, q, C_j) = \frac{\widehat{ctr}(d_1, q, C_j) - \widehat{ctr}(d_2, q, C_j)}{\widehat{ctr}(d_1, q, C_j)} \quad (12)$$

where $d_1$ is the url domain in top position, and $d_2$ the one is the second position. The DiffTop is then weighted by Δ as follows:

$$w\_diff(q, C_j) = diff(q, C_j) \cdot \Delta(d_1, d_2, q, C_j) \quad (13)$$

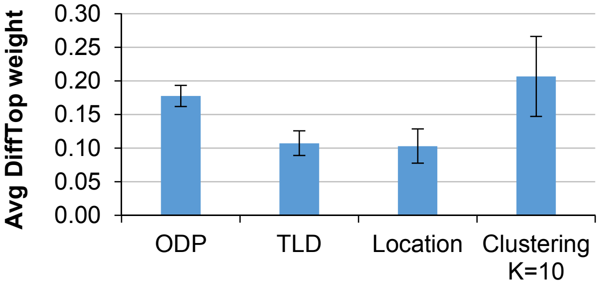

**Figure 5. Average DiffTop weight for each cohort (±SEM).**

If Δ equals zero, the top candidate has less cohort dominance, thus DiffTop a weaker signal about unique preference for this cohort.

To compare across cohorts, we average weighted DiffTop across queries for each cohort as $D(C_j) = \sum_q w\_diff(q, C_j)/diff(q, C_j)$. Such weight is then aggregated for a particular cohort type. Taking ODP cohorts for example, $D(C^O) = \sum_j D(C_j)/m$, where $m$ is the number of cohorts of type ODP.

If the average value is large, it implies that members of the cohort behave differently than non-members. We compare cohorts by ODP, TLD, Location features, also include clustered cohorts with $k$=10. Figure 5 shows the aggregated weighted DiffTop value. At least two insights can be made. The first is that all cohorts have high average DiffTop weights in general. This shows that our selected features are useful in distinguishing cohort choice and global choice. The second is that ODP and Clustered cohorts are more informative than TLD and Location, perhaps because they are denser.

# 7. DISCUSSION AND CONCLUSIONS

We have proposed an approach for using cohorts of searchers similar along one or more dimensions to enhance Web search personalization. To understand the value of these cohorts we performed an extensive set of experiments with predefined cohorts as well as cohorts dynamically learned from behavioral data, and for different query sets, including acronyms and queries previously unseen from a given user. These are scenarios where we would like to be able to employ personalization but often it does not succeed given insufficient data about the interests of individual users. The results of our experiments have clearly demonstrated the value of cohorts, especially for ambiguous and new queries from users, where our observed gains over a production ranker appear to be most significant.

In our experiments, we used a competitive baseline a ranking algorithm that already had personalization signals based on a number of personal and contextual features for individual searchers. Despite such attention to representing the individual user's interests, the cohort-based models presented in this paper were still able to enhance the strong personalization baseline and achieve significant gains. This is promising as it suggests that we can learn how to integrate the cohort signals and make decisions about when to use them in combination with individual signals when both are present, or in isolation when only cohort signals are available. That said, further experiments are necessary with other personalization models to assess the generalizability of our findings to other settings.

The pre-defined cohorts have the disadvantage that they require system designers to select important features manually in advance. Using unsupervised clustering we circumvented this problem and learned cohorts dynamically. We are pleased that using cluster-generated cohorts that outperformed the pre-defined cohorts. However, the success of any clustering method is dependent on the features that are used. In this paper we used a set of features associated with topical preference, location, and top-level domain preference, but there are other viable alternatives (e.g., demographics, social network cliques) and we need to explore their effectiveness in detail.

We have shown that that best performance from cohorts learned via $k$-means clustering is attained when we set $k$=10. In a production search engine handling millions of users and billions of queries, the amount of space that can be devoted to each user is minimal. We have shown that for each user we would only have to store a small amount of additional information about their cohorts in each user's profile, e.g., a single membership bit for each of the 10 cohorts.

Overall, it is clear that there is significant potential value from modeling cohorts in search personalization. Unlike most existing work that learn from each of similar individuals, our approach focuses on learning from the whole group(s). Our modeling has two main components: cohort construction and cohort behavior modeling. One direction of future work is enhancing each of these components, for example, leveraging other sources of data beyond query-click logs (e.g., browsing signals, social network information) for cohort construction, and considering relationships between cohort members (e.g., group dynamics) for cohort behavior modeling. Another direction is investigating generalized cohort models (e.g., employing a Bayesian framework with a cohort prior $P(C_j)$), and other clustering algorithms (e.g., hierarchical clustering).

# REFERENCES

1. Agichtein, E., Brill, E., and Dumais, S. (2006). Improving Web search ranking by incorporating user behavior information. *SIGIR*, 19–26.
2. Almeida, R. and Almeida, V. (2004). A community-aware search engine. *WWW*, 413–421.
3. Bennett, P.N., Radlinski, F., White, R.W., and Yilmaz, E. (2011). Inferring and using location metadata to personalize web search. *SIGIR*, 135–144.
4. Bennett, P., Svore, K., and Dumais, S. (2010). Classification-enhanced ranking. *WWW*, 111–120.
5. Bennett, P., White, R.W., Chu, W., Dumais, S., Bailey, P., Borisyuk, F., and Cui, X. (2012). Modeling the impact of short and long-term behavior on search personalization. *SIGIR*, 185–194.
6. Berger, A.L. and Lafferty, J. (1999). Information retrieval as statistical translation. *SIGIR*, 222–229.
7. Bilenko, M. and White, R.W. (2008). Mining the search trails of surfing crowds: identifying relevant websites from user activity. *WWW*, 51–60.
8. Burges, C.J.C., Ragno, R., and Le, Q.V. (2006). Learning to rank with non-smooth cost functions. *NIPS*, 193–200.
9. Chapelle, O., Chang, Y., and Liu, T.-Y. (2010). The Yahoo! learning to rank challenge. *http://learningtorankchallenge.yahoo.com.*
10. Cheng, H. and Cantú-Paz, E. (2010). Personalized click prediction in sponsored search. *WSDM*, 351–359.
11. Freyne, J. and Smyth, B. (2006). Cooperating search communities. *AH*, 101–110.
12. Fox, S., Kuldeep, K., Mydland, M., Dumais, S., and White, T. (2005). Evaluating implicit measures to improve Web search. *ACM TOIS,* 23(2*)*: 147–168
13. Goldberg, D., Nichols, D., Oki, B.M., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *CACM*, 35(12): 61–70.
14. Ieong, S., Mishra, N., Sadikov, E., and Zhang, L. (2012). Domain bias in Web search. *WSDM*, 413–422.
15. Joachims, T. (2002). Optimizing search engines using clickthrough data. *KDD*, 133–142.
16. Kautz, H., Selman, B., and Shah, M. (1997). Referral Web: combining social networks and collaborative filtering. *CACM*, 40(3): 63–65.
17. Lee, Y-J. (2005). VizSearch: A collaborative web searching environment. *Computers and Education*, 44(4): 423–439.
18. Mei, Q. and Church, K. (2008). Entropy of search logs: How hard is search? With personalization? With backoff? *WSDM*, 45–54.
19. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Reidl, J. (1994). GroupLens: An open architecture for collaborative filtering of netnews. *CSCW*, 175–186.
20. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M., and Ungar, D. (2002). Methods and metrics for cold-start recommendations. *SIGIR*, 253–260.
21. Shen, S., Hu, B., Chen, W., and Yang, Q. (2012). Personalized click model through collaborative filtering. *WSDM*, 323–332.
22. Shen, X., Tan, B., and Zhai, C.X. (2005). Implicit user modeling for personalized search. *CIKM*, 824–831.
23. Smyth, B. (2007). A community-based approach to personalizing Web search. *IEEE Computer*, 40(8): 42–50.
24. Sontag, D., Collins-Thompson, K., Bennett, P.N., White, R.W., Dumais, S.T., and Billerbeck, B. (2012). Probabilistic models for personalizing web search. *WSDM*, 433–442.
25. Sugiyama, K., Hatano, K., and Yoshikawa, M. (2004). Adaptive Web search based on user profile constructed without any effort from users. *WWW*, 675–684.
26. Sun, J.-T., Zeng, H.-J., Liu, H., Lu, Y., and Chen, Z. (2005). CubeSVD: A novel approach to personalized Web search. *WWW*, 382–390.
27. Tan, B., Shen, X., and Zhai, C. (2006). Mining long-term search history to improve search accuracy. *KDD*, 718–723.
28. Taneva, B., Cheng, T., Chakrabati, K., and He, Y. (2013). Mining acronym expansions and their meanings using query click log. *WWW*, 1261–1272.
29. Teevan, J., Adar, E., Jones, R., and Potts, M.A.S. (2007). Information re-retrieval: repeat queries in Yahoo's logs. *SIGIR*, 151–158.
30. Teevan, J., Morris, M.R., Bush, S. (2009). Discovering and using groups to improve personalized search. *WSDM*, 15–24.
31. Teevan, J., Dumais, S.T., and Horvitz, E. (2005). Personalizing search via automated analysis of interests and activities. *SIGIR*, 449–456.
32. Teevan, J., Liebling, D.J., and Geetha, G.R. (2011). Understanding and predicting personal navigation. *WSDM*, 85–94.
33. Weber, I. and Castillo, C. (2010). The demographics of Web search. *SIGIR*, 523–530.
34. White, R.W., Bailey, P., and Chen, L. (2009). Predicting user interests from contextual information. *SIGIR*, 363–370.
35. White, R.W., Bennett, P.N., and Dumais, S.T. (2010). Predicting short-term interests using activity-based search context. *CIKM*, 1009–1018.
36. White, R.W. and Buscher, G. (2012). Characterizing local interests and local knowledge. *SIGCHI*, 1607–1610.
37. White, R.W., Chu, W., Hassan, A., He, X., Song, Y., and Wang, H. (2013). Enhancing personalized search by mining and modeling task behavior. *WWW*, 1411–1420.
38. White, R.W., Dumais, S.T., and Teevan, J. (2009). Characterizing the influence of domain expertise on web search behavior. *WSDM*, 132–141.
39. Wu, Q., Burges, C.J.C. Svore, K.M., and Gao, J. (2008). Ranking, boosting and model adaptation. *Microsoft Research Technical Report MSR-TR-2008-10.*
40. Xiang, B., Jiang, D., Pei, J., Sun, X., Chen, E., Li, H. (2010). Context-aware ranking in web search. *SIGIR*. 451–458.