Personalized Interactive Faceted Search

Jonathan Koren, Yi Zhang
Jack Baskin School of Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064, USA
{jonathan, yiz}@soe.ucsc.edu

Xue Liu
School of Computer Science
McGill University
Montreal, Quebec, Canada, H3A 2A7
xueliu@cs.mcgill.ca

ABSTRACT

Faceted search is becoming a popular method to allow users to interactively search and navigate complex information spaces. A faceted search system presents users with key-value metadata that is used for query refinement. While popular in e-commerce and digital libraries, not much research has been conducted on which metadata to present to a user in order to improve the search experience. Nor are there repeatable benchmarks for evaluating a faceted search engine. This paper proposes the use of collaborative filtering and personalization to customize the search interface to each user's behavior. This paper also proposes a utility based framework to evaluate the faceted interface. In order to demonstrate these ideas and better understand personalized faceted search, several faceted search algorithms are proposed and evaluated using the novel evaluation methodology.

Categories and Subject Descriptors:
H.3.3 [Information Search and Retrieval]: Information filtering
General Terms:
Algorithms
Keywords:
collaborative recommendation, faceted search, interactive search, user modeling, personalization, evaluation

1. INTRODUCTION

Helping users find documents/items with facets has become an important problem for Internet services. Facets are metadata that can define alternative hierarchical categories for the information space. Unlike traditional categories, facets allow a document to exist simultaneously in multiple overlapping taxonomies [17]. For example, a system containing documents on motion pictures could have facets such as *director*, *year of release*, and *genre*. By using these facets, users can easily combine the hierarchies in whatever way best meets their information needs.

Some search engines provide advanced search functionalities to help users to find documents/items with facets. These interfaces allow a user to put constraints over document facets, such as limiting the search results to specific areas of a site or limiting the results to a specific language. Generally, these advanced search functionalities are useful. For example, a user who wants to download a Kevin Smith movie can use the structured query `director:Kevin Smith filetype:mpeg` in order to narrow down the search results to the movies that Kevin Smith directed. However, average users are either not effective at issuing complex queries, or are unwilling to take the effort to do so. Thus, most users do not use advanced search functionalities and the mean query length is only about 2.4 words according to a study based on 60,000,000 searches [7].

An alternative method to solve this problem is *faceted search* [16]. Instead of waiting for the user to create structured queries from scratch, a faceted search interface allows the user to progressively narrow down the choices by choosing from a list of suggested query refinements. Faceted search is one of the prevailing e-commerce mechanisms. For example, a customer can narrow down the list of candidate products by putting constraints over the *category*, *price*, *brand*, and *age* facets at `toysrus.com`. However, the facet types (category, price, brand, etc.) and the possible values for each facet are usually manually defined for a specific e-commerce site. For general purpose retrieval, automatic facet and facet-value recommendations are needed.

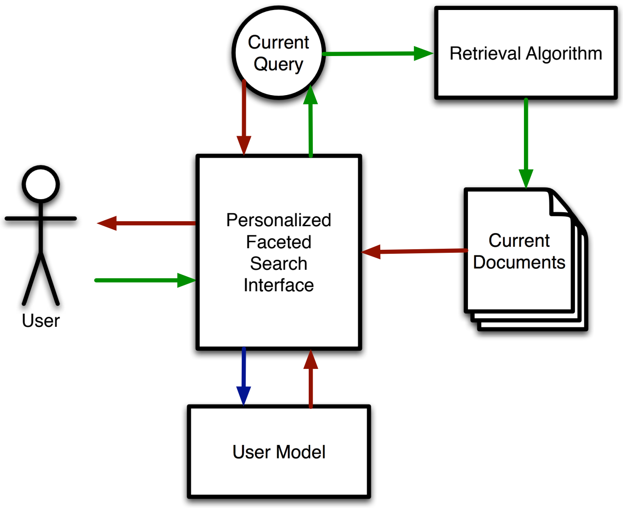
Faceted search interfaces share three characteristics. The interfaces present a number of facets along with a selection of their associated values, any previous search results, and the current query. By choosing from the suggested values of these facets, a user can interactively refine the query. The interface also provides a mechanism to remove previously chosen facets, thus widening the current search space. Studies have shown that users find faceted search interfaces intuitive and easy to use [5][15].

One key problem to building a faceted search interface is selecting which facets and facet-values to make available to the user at any one time. This is especially important when the document domain is very large. Some systems show users all available facets and facet-values. This approach can quickly overwhelm the users and lead to diminished user performance [15]. Other systems such as eBay Express[1] present a manually chosen subset of facets to the user, and the facet-values are ranked based on their frequency. Other systems, such as Flamenco [5], simply present the first few facet-values in an alphabetized list. For systems with a large number of facets, manually selecting and maintaining a system of "blessed" facets may be too time consuming. Also, a pre-defined interface may not serve all users of the system adequately. This approach may also be slow to react to changes in the users' behavior. What is needed is an automatic mechanism to select facets and facet-values for

WWW 2008, April 21–25, 2008, Beijing, China.
ACM 978-1-60558-085-2/08/04.

[1] http://express.ebay.com

Figure 1: Personalized Interface Block Diagram



presentation to a user based on the behavior of the users if the system, and the expectation of the utility of that facet or facet-value to a user during a search.

This paper proposes using explicit user ratings to generate an "intelligent" faceted search interface that selects facets and facet-values automatically to create an interface tailored to a user's preferences. The idea of using explicit user feedback is motivated by the success of online reviewing/rating systems such as Flickr[2], Netflix[3], and YouTube[4], which have demonstrated that millions of users are willing to provide explicit ratings about items. Figure 1 shows the general structure of this system.

The idea of personalized browsing is not new. Li et al. briefly mentioned personalized browsing of social annotations based on cosine similarity [10]. Pandit et al. discussed navigation as a retrieval method [14]. Personal WebWatcher passively observed a user's browsing behavior in order to highlight links that matched the inferred task [12]. Features from the webpages that were visited by the user were used to develop a "related pages" advisor and an "interesting part" highlighter [3].

The work presented in this paper differs from these works by focusing on faceted search, which is not well studied. An efficient personalized faceted search mechanism can be used to 1) solve millions of e-commerce users' immediate information needs; 2) help users better understand the data, especially the data space relevant to the user; and 3) help users better understand how the engine works through the simple interactive interface, and thereby train users in how to make more effective use of the interface over time. The contribution of this paper is three-fold:

- We propose to personalize faceted search for searching and browsing an unknown repository. We propose a general probabilistic framework to build faceted document models and user relevance models.
- We propose and formulate an inexpensive methodology for evaluating interactive faceted search engines.

Table 1: Facet Types

Facet Type	Values	Facet
Nominal	Unique Tokens	Authors
Ordinal	Repeatable Ordered Tokens	Letter Grade
Interval	Repeatable Numbers Excluding Zero	Copyright Year
Ratio	Repeatable Numbers Including Zero	Page Count
Free-Text	Repeatable Tokens	Synopsis

- In order to demonstrate the proposed framework and evaluation methodology, several techniques to automatically generate faceted search interfaces are compared. Experimental results show that the utility of the system depends on the algorithm used for recommending facet-value pairs, the algorithm for generating the landing page, and how a user interacts with the system.

2. PROBABILISTIC FRAMEWORK FOR PERSONALIZED FACETED SEARCH

In order to properly construct an effective personalized faceted search interface, formal descriptions of both the contents of the documents being searched and the users' search needs are required.

This section proposes a probabilistic generative document modeling framework for faceted metadata. Then we also describe how to build a user relevance model for each individual users based on information from other users.

For the remainder of this paper, the following notations to represent the variables in the system.

$k = 1, 2, ..., K$**:** The index to an individual facet. K is the total number of facets.

$u = 1, 2, ..., U$**:** The index to an individual user. U is the total number of users.

θ_u**:** The user model parameter associated with user u.

2.1 Document Model

An important first step to solving any information retrieval problem is to express the structure of the document corpus in terms of probabilities. Faceted documents differ from traditional plain text documents in that they are made up of a series of keys (i.e. facets), each of which has a unique of values.

While a specific facet-value pair usually occurs at most once in a document, there may be multiple facet-value pairs that share the same facet or value, depending on the semantics of the facet and the value. Furthermore, some facets in a corpus may be mandatory, while others may be optional. This situation can be modeled by assuming that every document has every possible facet, with the number of values for each facet drawn from a multivariate normal:

$$\langle n_1, ..., n_K \rangle \sim MVN(\vec{\mu}, \Sigma) \qquad (1)$$

Where $\vec{\mu}$ is a K dimensional vector, with each dimension representing the mean of the number of values for the corresponding facet. Σ is the corresponding $K \times K$ covariance matrix.

[2] http://www.flickr.com
[3] http://www.netflix.com
[4] http://www.youtube.com

The distribution of the values for a facet depends the facet's type. There are five commonly used facet types: *nominal*, *ordinal*, *interval*, *ratio*, and *free-text* (Table 1).

A facet of nominal type has discrete and orderless values. The values of a nominal type facet can occur at most once per facet, and so we can group all nominal facets together and represent them as draws from a multivariate Bernoulli distribution.

An ordinal facet has discrete values, with each value having an implicit ranking. The interval and ratio facet types are similar to each other in that both have continuous and ordered values. However, they differ slightly in that values of a ratio facet contain a true zero to compare against, while interval facets do not have an absolute zero point. For simplicity, we can represent the distribution of the values for an ordinal, interval, or ratio facets can be grouped together as draws from a multivariate normal distribution.

The last facet type, free-text, is a special type. Unlike the other types, where the value in each facet-value pair is a single token, the values for free-text facets are made up of multiple, possibly repeating, tokens. This type is used to represent natural language text that does not easily map to any of the other previous types. This type can be represented by repeated draws from a multinomial distribution, which is similar to commonly used language models.

2.2 User Relevance Model

In order for the system to adapt to a user's needs, a user relevance model that can be used to predict the user's preferences must be learned from the user data. Faceted search engines essentially perform two separate, but related, retrieval tasks. Like all search engines, they must return the documents that best match a user's query. However, they must also provide useful suggestions for query refinement. Since the first task is heavily studied in the information retrieval community, this paper focuses on the problem of query suggestion in faceted search.

In a faceted search system, a query is a list of facet-value pairs that jointly specify the required properties of a matching document. Recommended query refinements are presented as a list of facet-value pairs that a user can select from. In order to recommend relevant facet-value pairs to better match the user's preferences, the relevance of specific facet-value pairs to a user must be modeled.

The user modeling approach used in this paper focuses on modeling how a document's facet-values are generated. Let $P(rel|u)$ be the probability of any document being relevant to a user u. It is assumed that the user relevance model does not change over time, and thus a document is relevant to a user if it satisfies at least one of the previous information needs of the user.

A statistical user model assigns a probability to a document x. Assume that each facet-value pair x_k in a relevant document for user u is generated independent of all other facet-value pairs based on a distribution $\mathrm{P}(x_k \mid rel, u)$, and each facet-value pair x_k in a non-relevant document for user u is generated independent of all other facet-value pairs based on a different distribution $\mathrm{P}(x_k \mid non, u)$. The user model of a particular user u is represented as:

$$\theta_u = \{\mathrm{P}(rel \mid u), \mathrm{P}(x_k \mid rel, u), \mathrm{P}(x_k \mid non, u)\}$$

where $k = 1...K$.

For example, consider a document with only one free-text type facet. For a particular user u, assume there exists a set of training documents $\mathcal{X}_u = \langle \mathcal{X}_{u,rel}, \mathcal{X}_{u,non} \rangle$, where $\mathcal{X}_{u,rel}$ is a set of relevant training documents and $\mathcal{X}_{u,non}$ is a set of non-relevant training documents. Then the maximum likelihood estimation of the multinomial user model is very simple:

$$\mathrm{P}(rel \mid u) = \frac{\mid \mathcal{X}_{u,rel} \mid}{|\mathcal{X}_u|} \quad (2)$$

$$\mathrm{P}(x_k \mid rel, u)) = \frac{1}{|\mathcal{X}_u|} \sum_{x \in \mathcal{X}_{u,rel}} x_k \quad (3)$$

$$\mathrm{P}(x_k \mid non, u) = \frac{1}{|\mathcal{X}_u|} \sum_{x \in \mathcal{X}_{u,non}} x_k \quad (4)$$

This simple example is very similar to a commonly used relevance language model [19].

2.3 Collaborative User Relevance Model

Since a detailed user model is being learned for each individual user, it may take a while before the system can gather enough data from the user to estimate the model parameters reliably. However, good initial performance is the incentive for a new user to continue using the system.

Based on the assumption that a user shares similar criteria, preferences, or action patterns with some other user, information can be borrowed from others in order to aid a new user. The idea of learning from others is called "social learning" in psychology and can be traced back to 1940's [11]. The research on social learning is more focused on explaining human behavior *"learned observationally through modeling: from observing others one forms an idea of how new behaviors are performed, and on later occasions this coded information serves as a guide for action"* [1]. The same idea has been used in IR to develop "collaborative filtering" systems that make recommendations to a user by computing the similarity between one user's preferences and that of other users [9]. One well-received approach to improve recommendation system performance for a particular user is borrowing information from other users through a Bayesian hierarchical modeling approach. Several researchers have demonstrated that this approach effectively trades off between shared and user-specific information, thus alleviating poor initial performance for each user [20][18].

Previous work provided the motivation to use a hierarchical Bayesian modeling approach in order to help learn the current user's model. This leads to the assumption that the user model θ_u is a random sample from a prior distribution $\mathrm{P}(\theta)$.

For different facet types, the system needs to use a different functional form for user relevance model $\mathrm{P}(x_k \mid u, rel)$ and prior distribution $\mathrm{P}(x_k \mid rel)$. Tables 2 and 3 list each facet type, user model functional form, prior distribution and corresponding recommended EM update steps.[5] The recommended prior is a conjugate prior for the corresponding distribution, and are recommended for mathematical convenience.

In the previous example where a document contains only a single free-text type facet, the user model is a multinomial distribution, and so a Dirichlet prior is recommended. This

[5] The non-relevance user model $\mathrm{P}(x_k \mid u, non)$ and prior distribution $\mathrm{P}(x_k \mid non)$ are not listed since they are very similar.

Table 2: Posterior Probability Update Rules for Facets

Type	Distribution	Prior	E-Step
Free-Text	Multinomial	Dirichlet	$\mathrm{P}(x_k \mid rel, u) = \frac{m!}{x_{k,1} \ldots x_{k,n}} \prod_{j=1}^{n} (\alpha_{rel,k,j} + \sum_{x \in \mathcal{X}_{u,rel}} x_{k,j})^{x_{k,j}}$
Nominal	Multivariate Bernoulli	Multivariate Gamma	$\mathrm{P}(x_k \mid rel, u) = \frac{\alpha_{rel,k} + \sum_{x \in \mathcal{X}_{u,rel}} x_k}{\alpha_{rel,k} + \beta_{rel,k} + \lvert\mathcal{X}_u\rvert}$
Ordinal, Interval & Ratio	Normal	Normal	$\mathrm{P}(x_k \mid rel, u) = \frac{1}{\lvert\mathcal{X}_u\rvert \sigma_{rel,k} \sqrt{2\pi}}$ $\left(\sum_{x \in \mathcal{X}_u} exp(-(x_k - \mu_{rel,k})^2 / (2\sigma_{rel,k}^2)) \right)$

Table 3: Prior Probability Update Rules for Facets

Type	Distribution	Prior	M-Step
Free-Text	Multinomial	Dirichlet	$\overrightarrow{\alpha}_{rel,k} = \frac{1}{\lvert\mathcal{U}\rvert} \sum_{u=1}^{\lvert\mathcal{U}\rvert} \sum_{x \in \mathcal{X}_u} \overrightarrow{p}_{rel,k,u} \overrightarrow{x_k} I_{\{x\ is\ rel\ to\ u\}}$
Nominal	Multivariate Bernoulli	Multivariate Gamma	$\alpha_{rel,k} = \frac{1}{\lvert\mathcal{U}\rvert} \sum_{u=1}^{\lvert\mathcal{U}\rvert} \sum_{x \in \mathcal{X}_u} p_{rel,k,u} x_k I_{\{x\ is\ rel\ to\ u\}}$ $\beta_{rel,k} = \frac{1}{\lvert\mathcal{U}\rvert} \sum_{u=1}^{\lvert\mathcal{U}\rvert} \lvert\mathcal{X}_u\rvert - \sum_{x \in \mathcal{X}_u} p_{rel,k,u} x_k I_{\{x\ is\ rel\ to\ u\}}$
Ordinal, Interval & Ratio	Normal	Normal	$\mu_{rel,k} = \frac{1}{\lvert\mathcal{U}\rvert} \sum_{u=1}^{\lvert\mathcal{U}\rvert} \mu_{rel,k,u} I_{\{x\ is\ rel\ to\ u\}}$ $\sigma_{rel,k}^2 = \frac{1}{\lvert\mathcal{U}\rvert} \sum_{u=1}^{\lvert\mathcal{U}\rvert} \sigma_{rel,k,u}^2 I_{\{x\ is\ rel\ to\ u\}}$

is very similar to the commonly used relevance language model with Dirichlet smoothing [19].

3. PROPOSED EVALUATION METHODOLOGY

Evaluating personalized faceted search is a very challenging problem that has not been well studied. A commonly used interactive system evaluation methodology is to carry out user studies. Although undoubtedly useful, this approach is limited because: 1) user studies are usually very expensive; 2) it is very hard for the research community to repeat a user study; 3) the number of subjects in user studies are usually limited; and 4) a personalization system may not succeed until a user has interacted with it for a long period of time. As a result, many prior user studies on personalization generate inconclusive or contradictory results.

This section proposes an alternative evaluation methodology for faceted search based on user simulation and explicit relevance feedback collected over a long period of time from many actual users. This methodology is designed around the concept of measuring the utility of an interface based on which actions a user performs in a search session. This new methodology can be used to evaluate personalized faceted search interfaces in a quick, cheap and repeatable manner.

3.1 User Interface Utility

While the specifics vary between individual faceted search interfaces, every interface shares certain characteristics. In general, a faceted search interface is divided into three parts. The first part is a list of the facets in the document collection. Each facet has a list of available values associated with it. When there are a large number of values for a facet, interfaces tend to display only a fraction of the available values, but allow the user to view the complete list upon request. The user can restrict the current query by selecting facet-value pairs from this list.

The second part of the interface is the display of current query's criteria. This display not only reminds the user what subset of the search space the user is examining, but also allows the user to broaden the current query by removing some previously selected facet-value pairs.

The final and most prominent part of a faceted search interface is the document list. The document list displays the most relevant documents to the current query. The user can scan the entire list of matching documents (usually a subset at a time), and then select a specific document for retrieval.

Given this broad description, an evaluation measure can be defined to quantify an interface's search effectiveness. The effectiveness of an interface should be measured with respect to how well the user's information need is satisfied compared to the amount of effort required by a user.

A user's information need can be expressed as a subset of the documents in the corpus being searched, and the number of documents of this subset that must be retrieved by the user. Given this assumption, the success of a user in meeting his/her information need can be measured by the number of relevant documents successfully retrieved or recalled. Since the number of documents required for a completely successful search can exceed one, and faceted search interaction is essentially a series of query refinements, evaluation must be with respect to a search session, instead of an individual query.

For each user search session, assume the user takes a sequence of T actions $(a_1, a_2, ..., a_T)$. At each time t, the user takes an action a_t, which changes the state of the user interface from q_t to q_{t+1}. The user utility for this session is defined as:

$$\mathrm{U} = \sum_{t=0}^{T} \mathrm{R}(q_{t+1}, a_t, q_t) \qquad (5)$$

q_t is represented as a combination of the current query, the suggested facet-value pairs, and the system suggested documents at time t. $\mathrm{R}(q_{t+1}, a_t, q_t)$ represents the reward that the system receives when the user transitions from state q_t to q_{t+1} via action a_t. For example, the reward could be

-1 if the user clicks a link and does not find any relevant documents. Conversely, if the user clicks a link and finds a relevant document, the reward could be 100.

When determining a user's total effort in manipulating the interface, it is important to note that this is not simply the number of actions performed, but also what type of actions are performed. What follows is a list of the actions that a user can perform and how each action rewards the system when a user executes it[6]. This paper assumes that the goal of the system is to minimize the total user effort required to fulfill his/her information need. Therefore, a negative reward is assigned to most actions.

Select Facet-Value Pair This action allows the user to place an additional constraint on the current query. After performing this action, the user receives a new list of relevant documents, along with a new list of facet-value pairs that may be useful for further query refinement. The system receives a penalty every time the user performs this action.

De-select User Selected Facet-Value Pair By performing this action, the user removes a previously imposed constraint from the current query. Depending on the reasons why the user previously selected this facet-value pair, the system may or may not be punished when the user performs this action.

De-select System Selected Facet-Value Pair This action is similar to other de-select action, except with this action, the user de-selects a constraint to the current query that was automatically added by the system. Since the system was responsible for imposing the constraint, the system receives a penalty.

View More Facet-Value Pairs With this action, the user requests to see more suggested values for a specific facet. A user could execute this action if none of the initially suggested facet-value pairs appeared relevant to the current information need. The system receives a penalty for when the user performs this action.

Mark Document as Relevant This action indicates that the user examined a document and found that it at least partially satisfied his/her information need. This action is used to provide positive feedback to both the facet-value suggestion mechanism and the underlying document ranking mechanism. Since this action represents success, its reward must be positive and large enough to overwhelm all the penalties accumulated during the search session up until this point.

Mark Document as Non-relevant This action indicates that the user examined a document, and found that it does not satisfy his/her information need. This action indicates that the user initially believed that a document was relevant, perhaps based on the document's title or snippet, but upon further inspection determined that it was not relevant to the current information need. The system receives a negative reward for this action.

View More Documents This action allows the user to view additional documents that may be relevant to the current query. Since this action does not impact which facet-value pairs are suggested to the user for refinement, this action has a reward of zero.

End Session This is a dummy action that is used to indicate when the user terminated his/her search session. This is always the final action of any search session and will transition the user interface to a dummy stop state. Since this action has no impact on the user experience, its reward is zero.

Each action not only affects the behavior of the search engine, but also impacts the user's perceptions of the search experience [6][5][2]. For example, if a user is forced to reformulate each query several times in order to fully explore a topic, he/she may feel frustrated or overwhelmed by the task. Conversely, if a search interface provides useful query suggestions, the effort on the part of the user is reduced to simply clicking a link instead of coming up with all new search terms by him/herself. In this sense, manually reformulating a query is more expensive than clicking a recommended query. This is because the user must draw upon his/her domain knowledge and exert greater physical effort to carry out the reformulations. Thus an action that requires much user effort, such as rewriting a query, should be penalized more in the valuation framework.

3.2 Expected Utility Based Evaluation

A faceted search system should maximize the expected utility for all users $\mathcal{U}$ and all information needs $\mathcal{D}$. This is expressed as:

$$E[\mathrm{U}()] = \sum_{u\in\mathcal{U}} \sum_{D\in\mathcal{D}} E[\mathrm{U}(u,D)]\mathrm{P}(D \mid u)\mathrm{P}(u) \tag{6}$$

with:

$$\begin{aligned} E[\mathrm{U}(u,D)] = \sum_{t=0} \sum_{a\in\mathcal{A}_t} & \mathrm{R}(q_{t+1},a,q_t) \\ & \mathrm{P}(q_{t+1} \mid a,q_t,u) \\ & \mathrm{P}(a \mid q_t,q_{t-1},...,q_0,u,D) \\ & \mathrm{P}(q_t \mid q_{t-1},...,q_0,u,D) \end{aligned} \tag{7}$$

where D represents user u's information need for a specific search session, and $\mathcal{A}_t$ is the set of possible actions a user can take at time t.

Assuming that the Markov property holds, Equation 7 simplifies to:

$$\begin{aligned} E[\mathrm{U}(u,D)] = \sum_{t=0} \sum_{a\in\mathcal{A}_t} \mathrm{R}(q_{t+1},a,q_t)\mathrm{P}(q_{t+1} \mid a,q_t,u) \\ \mathrm{P}(a \mid q_t,u,D) \\ \mathrm{P}(q_t \mid q_{t-1},u,D) \end{aligned} \tag{8}$$

3.3 User Interaction Assumptions

It is difficult, if not impossible, to express $\mathrm{P}(a \mid q_t,u,D)$. The correct probabilistic functional form is unknown. One could imagine that given enough resources, a system could observe many potential users for a long enough period of time and then use this data to estimate the conditional probability. However, such kind of data is system and user dependent, which makes it impossible to create a benchmark

[6] Some actions, such as reformatting a query, are not included since they are not always supported by faceted search engines. It is straightforward to extend the evaluation framework to more actions.

data set to compare different systems. A more practical approach is to make some reasonable assumptions about the users and estimate the expected utility based on simulated users.

In order to make the evaluation possible and repeatable, two assumptions are made. First, it is assumed that each user is searching for exactly one document that is guaranteed to be found within the document corpus. This document is called the "target document". Second, it is assumed that the user has perfect knowledge of the target document. This is not meant to say that the user can reconstruct the document, but rather that the user can recognize the target document from a list of documents, and can distinguish between the suggested facet-value pairs that match the target document, from those do not match. These assumptions are not always true. However, because there are a large number of cases where these assumptions are almost true. These assumptions were made so that the research was manageable. Making these assumptions narrows the focus of the evaluation to a subset of possible user cases.

A user begins by examining the first page of the list of documents returned by the current query. If the target document is in this list, the user selects it, and then ends the search session. If the target document is not found, then the user examines the current recommended query for any facet-value pairs that are not contained in the target document. If any are found, the user removes one, and then resubmits the query. If the query contains only terms in the target document, the user scans the list of suggested facet-value pairs for any that are contained within the target document. If one is found, then the user selects one by some mechanism, and resubmits the query. If the none of the presented facet-value pairs are contained to the target document, the user chooses a facet at random and examines all of its values and selects one for inclusion in the current query. If the user does not find any additional relevant facet-value pairs after examining all possible facet-value pairs, the user begin to scan through the complete list of the documents returned by the current query.

How does a user select a facet-value pair from all matching pairs suggested by the search engine? Four different heuristics are proposed.***Stochastic users*** randomly select from the presented facet-values pairs that are contained in the target document from a uniform distribution. ***First-match users*** scan the list of suggested facet-value pairs and select the first pair that is contained in the target document. The intuition for this heuristic is that facet-value pair selection may mimic how users select documents for retrieval from a list of candidate documents. ***Myopic users*** select the facet-value pair that is contained in the least number of documents and is also contained in the target document. The intuition for this heuristic is that if users are searching for a single document, they would like "zero in" on that document quickly by reducing the search space as much as possible. ***Optimal users*** have detailed knowledge of what documents are contained within the corpus and how the underlying retrieval system works. They use this knowledge to select the facet-value pairs that will maximize the interaction utility. Since finding the optimal set of interactions for each relevant document for a user is computationally intensive, optimal users are not simulated in our experiments and are only included for completeness.

4. PERSONALIZED FACET-VALUE PAIR RECOMMENDATION

As stated in the introduction, one of the keys to building an effective faceted search interface is presenting the user with facet-values that that are relevant to the user's current search task. If the presented facet-values are not relevant to the task, the user could be forced to spend extra effort to find his/her document(s), or in the worst case not find his/her document(s) at all. This section describes several possible algorithms to select facet-value pairs.

4.1 Facet-Value Pair Suggestions

After a user performs an action in the middle of a session, the system needs present a list of facet-value pairs. We can view this as a feature selection task, and the following is an incomplete list of algorithms that can be used:

Most Frequent This is the simplest suggestion method. In this method, the facet-value pairs that are found in the currently selected documents are counted, and the most frequent values for each facet are presented to the user for query refinement. This method is popular among many commercially available faceted search interfaces, and thus provided an appropriate baseline for comparison.

Most Probable In this method, the facet-value pairs in the currently selected documents are ranked according to their probability of being included in a document relevant to the user. These probabilities can either be determined by the relevance judgments by the community of users (*Collaborative Prob.*), or personalized for each individual user (*Personal Prob*). This method was examined as it can be easily integrated into adaptive and personalized retrieval algorithms.

Mutual Information The pointwise mutual information between the presence of a facet-value pair appearing in a document and a document's relevance is calculated. The most informative values are then presented to the user for query refinement. Mutual information was considered as a facet-value suggestion method since it is a common method used for feature selection.

4.2 Starting/Landing Page for Faceted Search

A faceted search system needs to present a good starting/landing page for a user. Since a user's profile describes the qualities that define a relevant document for that user, this information can be used by the system to initially place each user nearer to his/her relevant documents even before the user begins to formulate a query. This is accomplished by examining the user's profile and automatically constructing an initial query based on facet-value pairs that are likely to be contained in the user's relevant documents. This automatically constructed query, along with the documents returned by this query, creates the *start state.* The following three methods to determine user's start state are proposed.

Null Start State This is the simplest start state creation method. In this method, each user begins in a state with no facet-value pairs selected by default and no pre-fetched documents. This method served as a baseline for comparing the other start state creation methods as this is the most prevalent method for beginning user initiated searches in information retrieval systems.

Table 4: Reward Function

Action	Reward
Select FVP	-1
De-select User FVP	0 (not used)
De-select System FVP	-1
View More FVPs	-1 per page
Mark Document as Relevant	+100
Mark Document as Non-Relevant	-1 (not used)
View More Documents	0 per page
End Session	0

Collaborative Start State In this method the system automatically issues a query containing the facet-value pairs that are the mostly likely to be contained in a relevant document as determined by the common Bayesian prior. This query is then issued to the underlying retrieval algorithm and the matching documents are initially suggested to the user. Since the start state is created by the common prior, every user is presented the same start state.

Personalized Start State This method is similarly to the method above, except that the default query is determined by each user's profile.

5. EMPIRICAL EVALUATION

5.1 Data

In order to demonstrate these ideas, a set of experiments were carried out using documents from the Internet Movie Database (IMDB) corpus [8] along with real user relevance judgments for each document from the MovieLens [4] and Netflix Prize [13] corpora. These data sets were chosen since they provided documents containing facets, such as *director* and *actor*, along with real user relevance judgments on these documents. The IMDB corpus was trimmed to contain only documents found in either the MovieLens or Netflix corpora. This led to approximately 8,000 documents containing 367,417 facet-value pairs spread among 19 facets. Both the MovieLens and Netflix corpora were reduced to approximately 5,000 unique users through uniform sampling, giving 742,036 and 633,257 user judgments respectively. In both data sets users expressed a preference for retrieved documents based on a 1 to 5 rating scale. Each rating was converted into Boolean relevance judgments by assuming all movies that were rated 4 or greater were relevant, and all movies rated 3 or less were non-relevant. These user judgments were randomly divided 90% for training and 10% for testing. For simplicity, all facets were assumed to be nominal. Each user model was a multivariate Bernoulli distribution, with the shared prior being a multivariate Gamma distribution. Without loss of generality, a simple reward mechanism shown in Table 4 is used for evaluation. The interface was configured to return a maximum of 10 matching documents per page, and a maximum of 5 values for each facet.

5.2 Experimental results

Tables 5 and 6 list the average number of interactions required by a user to find the his/her target document. Note that with a unit penalty for each action and a large credit for each relevant document found, a smaller average number of

Table 5: Search Efficiency for MovieLens Users.

Interaction Model	Start State	FVP Suggest Method	Ave Num Actions
First Match	Null	Frequency	5.4
First Match	Null	Collab Prob	6.3
First Match	Null	Personal Prob	5.6
First Match	Null	PMI	30.4
First Match	Collab	Frequency	3.7
First Match	Collab	Collab Prob	7.5
First Match	Collab	Personal Prob	6.2
First Match	Personal	Personal Prob	3.6
First Match	Collab	PMI	25.7
Myopic	Null	Frequency	4.3
Myopic	Null	Collab Prob	9.1
Myopic	Null	Personal Prob	5.9
Myopic	Null	PMI	121.3
Myopic	Collab	Frequency	4.3
Myopic	Collab	Collab Prob	8.5
Myopic	Collab	Personal Prob	5.6
Myopic	Personal	Personal Prob	5.0
Myopic	Collab	PMI	110.1

Table 6: Search Efficiency for Netflix Users

Interaction Model	Start State	FVP Suggest Method	Ave Num Actions
First Match	Null	Frequency	6.1
First Match	Null	Collab Prob	7.5
First Match	Null	Personal Prob	6.5
First Match	Null	PMI	52.3
First Match	Collab	Frequency	4.3
First Match	Collab	Collab Prob	5.1
First Match	Collab	Personal Prob	4.5
First Match	Personal	Personal Prob	6.5
First Match	Collab	PMI	44.1
Myopic	Null	Frequency	5.7
Myopic	Null	Collab Prob	11.5
Myopic	Null	Personal Prob	7.5
Myopic	Null	PMI	202.4
Myopic	Collab	Frequency	5.6
Myopic	Collab	Collab Prob	10.8
Myopic	Collab	Personal Prob	7.1
Myopic	Personal	Personal Prob	7.5
Myopic	Collab	PMI	178.4

actions corresponds to a bigger user utility. Four noteworthy conclusions can be made from these results.

First, pointwise mutual information (PMI) significantly under performed when compared to the other facet-value selection methods. Mutual information measures the correlation between two random variables, in this case the presence or absence of a facet-value pair and a document's relevance. Pointwise mutual information rather than complete mutual information was used because only facet-value pairs that have a positive correlation should be suggested to the user.

PMI breaks down when there are facet-value pairs that are strongly correlated with relevant documents, but occur in only tiny fraction of all relevant documents. For example, if all films containing the facet-value pair *genre=filmnoir* are ranked highly, then *genre=filmnoir* will be suggested early for query refinement, even if *genre=filmnoir* is contained in less than 1 percent of all relevant documents. These results show that correlation measures such as mutual information are not a good choice for this type of selection problem, since the probability of utilizing the suggested features is more important than how tightly correlated they are with relevance.

The second conclusion that can be drawn is that first match users tended to find their documents faster than myopic users. This seems counter-intuitive, since the myopic user shrinks the set of candidate documents as quickly as possible. Examining which facet-value pairs were suggested at each point in the interaction revealed that the system would present multiple relevant facet-value pairs. However, after the myopic user would select the least frequently occurring pair, the other previously presented relevant pairs would be re-ranked and no longer presented. Meanwhile, the least frequently occurring pair is more likely to recommended by the faceted search engine as a result of over fitting the training data. This revealed that a simple greedy approach to utilizing facet-values for retrieval might not actually be the best choice when interacting with a faceted search interface. However, one should not lose sight of the fact that a search interface should be optimized for real world users, and thus designers have little control over what selection method users actual employ when interacting with the system. An analysis of user behavior should be carried out, and this knowledge used to create more accurate simulated users.

Finally, simply suggesting the most frequent values for each facet performed well when compared to the personalized suggestion methods. There are two possible reasons for this. First, the frequency of facet-value pairs in the documents is correlated with users' idea of what makes a document relevant. In general this may not be the case, and thus frequency may not be a good facet-value selection mechanism for when the users' expectations do not closely match what is contained in the document repository. In this case frequency would fail to provide good suggestions, and the personalized probability models would be shown to be superior. Second, the *Personal Prob* algorithm used to select facet-value pairs is not good enough and far from optimal. This is not surprising since the probabilistic model proposed in this paper is based on strong assumptions that may not be true on the evaluation data sets.

6. CONCLUSIONS

This paper explored how to use explicit user ratings to design a personalized faceted search interface. Several algorithms to generate personalized faceted search interfaces were proposed and evaluated. This paper compares different algorithms for the same simple faceted search interface, which is not a well studied topic.

Researchers have tried to tell whether faceted search or structured query recommendation helps, or whether personalization helps. This paper showed that the algorithms used to build a faceted search interface, to recommend structured queries, and to personalize the interface make a difference. The expected user satisfaction differs significantly for the different algorithms. Famous algorithms for feature selection such as mutual information work poorly.

This paper also proposed a cheap evaluation methodology for personalized faceted search research. The experimental environment is repeatable and controllable, which makes it a benchmarkable evaluation environment. Although the simulated users differ from real users, the evaluation methodology does provide insight into understanding how various faceted interface design algorithms perform. This paper does not intend to claim whether this evaluation method is better or worse than user studies. Instead, the outlined approach serves to complement user studies by being cheap, repeatable, and controllable.

How to select a set of facet-value pairs at each step of the interaction process to optimize a user utility is a more fundamental that requires future research. This paper serves a first step towards personalized faceted search. The facet-value pair selection algorithms examined in this paper are far from optimal. For example, the semantic relationship among facets were not examined, nor was how the possibility of how a user's interests changing over time could affect the performance of the learned interface. These are future research topics.

7. ACKNOWLEDGMENT

This research was funded by National Science Foundation IIS-0713111, the Petascale Data Storage Institute under Department of Energy award DE-FC02-06ER25768, Los Alamos National Laboratory under ISSDM, and the industry sponsors of the Information Retrieval and Knowledge Management Lab at University of California Santa Cruz. Any opinions, findings, conclusions or recommendations expressed in this paper are the author's, and do not necessarily reflect those of the sponsors.

8. REFERENCES

[1] A. Bandura. *Social Learning Theory.* General Learning Press, 1977.

[2] B. B. Bederson. Interfaces for staying in the flow. *Ubiquity*, 5(27):1–1, 2004.

[3] Z. Chen, L. Wenyin, F. Lin, P. Xiao, L. Yin, B. Lin, and W.-Y. Ma. User modeling for building personalized web assistants. In *Proceedings of the 11th International World Wide Web Conference (WWW '02)*, 2002.

[4] GroupLens. Movielens. http://www.grouplens.org/taxonomy/term/14/, 2006.

[5] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen, and K.-P. Yee. Finding the flow in web site search. *Communications of the ACM*, 45(9):42–49, 2002.

[6] M. A. Hearst. User interfaces and visualization. In R. Baeza-Yates and B. Ribeiro-Neto, editors, *Modern Information Retrieval*, chapter 10, pages 257–323. ACM Press, New York, NY, USA, 1999.

[7] H. Inan. *Search Analytics: A Guide to Analyzing and Optimizing Website Search Engines*. Book Surge Publishing, 2006.

[8] Internet Movie Database Inc. Internet movie database. http://www.imdb.com/interfaces/, 2006.

[9] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

[10] R. Li, S. Bao, Y. Yu, B. Fei, and Z. Su. Towards effective browsing of large scale social annotations. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 943–952, New York, NY, USA, 2007. ACM.

[11] N. Miller and J. Dollard. *Social Learning and Imitation*. Yale University Press, 1941.

[12] D. Mladenić. Personal web watcher: Design and implentation. Technical report, J. Stefan Institute, Department for Intelligent Systems, Ljubljana, Slovenia, 1998.

[13] Netflix. Netflix prize. http://www.netflixprize.com, 2006.

[14] S. Pandit and C. Olston. Navigation-aided retrieval. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, 2007.

[15] V. Sinha and D. R. Karger. Magnet: Supporting navigation in semistructured data environments. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD '05)*, pages 97–106, 2005.

[16] D. Tunkelang. Dynamic category sets: An approach for faceted search. In *ACM SIGIR '06 Workshop on Faceted Search*, August 2006.

[17] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings for the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*, pages 401–408, 2003.

[18] K. Yu, V. Tresp, and S. Yi. A nonparametric hierarchical bayesian framework for information filtering. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*, pages 353 – 360, New York, NY, USA, 2004. ACM Press.

[19] C. Zhai and J. Lafferty. Two-stage language models for information retrieval. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02)*, New York, NY, USA, 2002. ACM Press.

[20] P. Zigoris and Y. Zhang. Bayesian adaptive user profiling with explicit and implicit feedback. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM 2006)*, pages 397 – 404, New York, NY, USA, 2006. ACM, ACM Press.