Approximately Optimal Facet Selection

Sonya Liberman∗
Department of Computer Science
Technion, Haifa 32000, Israel
sonyal@cs.technion.ac.il

Ronny Lempel
Yahoo! Research
Matam, Haifa 31905, Israel
rlempel@yahoo-inc.com

ABSTRACT
Multifaceted search is a popular interaction paradigm for
discovery and mining applications that allows users to ana-
lyze and navigate through multidimensional data. A crucial
aspect of faceted search applications is selecting the list of
facets to display to the user following each query. We call
this the facet selection problem.

When refining a query by drilling down into a facet, doc-
uments that are associated with that facet are promoted
in the rankings. We formulate facet selection as an opti-
mization problem aiming to maximize the rank promotion
of certain documents. As the optimization problem is NP-
Hard, we propose an approximation algorithm for selecting
an approximately optimal set of facets per query.

We conducted experiments over hundreds of queries and
search results of a large commercial search engine, compar-
ing two flavors of our algorithm to facet selection algorithms
appearing in the literature. The results show that our algo-

Categories and Subject Descriptors
H.3.3 [Information Storage and Retrieval]: Information
Search and Retrieval

General Terms
Algorithms, Experimentation

Keywords
Search engines, multifaceted search, facet selection

1. INTRODUCTION
Multifaceted search, also known as guided navigation, is a
popular and intuitive interaction paradigm for discovery and
mining applications that allows users to digest, analyze and
navigate through multidimensional data. Many Digital Li-
braries and e-commerce Web sites implement faceted search
applications. Lately, facets have begun appearing also in
result pages of general Web search engines. As Web queries
are often short and ambiguous, facets can assist searchers in
disambiguating their precise information need, or intent.

The interaction with a faceted search interface involves in-
terleaved search and browse operations over semi-structured
documents. In addition to containing text, the documents are
associated with facets - nodes (i.e., values) of a mul-
dimensional facet hierarchy/taxonomy1. At any step in the
search session the user may either (1) modify the search
query, (2) browse (drill-down) into one of several displayed
facets that further narrow the context of the current query,
or (3) remove some facets from the context (roll-up), hence
generalizing the context. Note that when narrowing a query
by drilling down into a facet, search results are filtered to
contain only those documents associated with the facet. Thus,
results belonging to the clicked facet are promoted in the
rankings, as documents that previously ranked ahead of
them and are not associated with the facet, are filtered out.2

A crucial aspect of faceted search applications is selecting
the list of facets to display following each query. The role of
the chosen facets is to guide users toward satisfying their in-
tents. Thus, the key is to display facets that are well-aligned
with the (latent) intents, and that partition the informa-
tion space in a manner that facilitates intent fulfillment. We call
this the facet selection problem.

This paper presents a novel algorithm for facet selection.
The algorithm selects facets that optimize the rank promo-
tion of documents it believes to satisfy users’ intents. We
model this belief by postulating a distribution over search re-

1While it is also common to refer to the dimensions them-
selves as facets, we will restrict the use of the term to the
actual values.
2In this paper we further assume that the relative order of
documents that are associated with the clicked facet does
not change following the drill-down operation.
selection of non-correlated facets, that promote the ranks of nearly disjoint sets of documents. Experiments conducted with two flavors of our algorithm show it is superior to facet selection algorithms that appear in the literature.

The rest of this paper is organized as follows. We survey related work in Section 2. In Section 3 we model user interaction with faceted search applications, and formally define the optimization problem of facet selection. Section 4 proves that the optimization problem is submodular, and proposes a greedy algorithm that selects a set of approximately optimal facets. Sections 5 and 6 report on our experiments with the proposed new algorithms, and show that they outperform baseline algorithms that are referenced in the literature. We conclude in Section 7.

2. RELATED WORK

Faceted search applications require faceted data, namely the existence of facet hierarchies and the mapping of documents onto those hierarchies. In two related papers, Dakka et al. [8, 7] describe algorithms for extraction of facet hierarchies from a corpus based on lexical subsumption, and assignment of the documents to those facets. Stoica and Hearst [25] use synsets and hypernym relations to accomplish a similar task. Feinstein and Smadja [11] describe the RawSugar social tagging system, which supports faceted search over tag hierarchies. Kohlschütter et al. [15] use personalized PageRank values for multiple ODP categories to (1) infer dominant facets in Web search results, and (2) support drill-down operations on the result set. Anick and Tipirneni [1] map each document to a list of its terms of high lexical dispersion, and at query time display terms appearing in several top ranking documents.

Another crucial aspect in deploying faceted search is the user interface, whose purpose is to clearly present the multidimensional information space to users, guiding and enabling them to make informed decisions on their interleaved search and navigation steps within the space. Hearst [14] provides observations based on many years of experiments on interface design, most recently as part of UC Berkeley’s Flamenco Search Interface project. Kules [18] studies how organizing large result sets into categorized overviews helps users in exploring and understanding such result sets. Shen et al. [24] studies visualization techniques in Digital Libraries that support integrated browsing and searching. Zhang and Marchionini [26] demonstrate that certain faceted UI choices can significantly shorten the time users require to complete search tasks.

Many faceted applications present aggregated statistics (e.g. document counts) for each facet on its own. Beyond that, Schneiderman et al. [23] plot two-dimensional tables with hieraxes - axes of hierarchical categories. Meredith and Pieper’s inverted index based BETA system [19] also displays two-dimensional tables for correlating pairs of facets.

Between the data model and the user interfaces lies the index. The Apache Solr open source project supports searching over a flat list of facets. Ben-YiTzhak et al. [3] describe an indexing scheme of hierarchical faceted data, along with the corresponding runtime algorithms for supporting various arithmetic and logical aggregations over the data.

Once the data model, the index structure and the UI paradigm have been decided and implemented, one reaches the main question addressed by this paper - which facets are the most appropriate ones for every given query. In many commercial applications of faceted search, the facets shown per query are set by business rules. For example, product families in an e-commerce site might have interesting facets defined for them, prompting facets in dimensions such as “optical zoom” and “megapixels” to be presented for “digital camera” searches. Such tailoring of facet selection is possible when both the content and query domains are small and well-understood, but cannot achieve Web-scale.

Krellenstein [17] describes a facet selection heuristic where facets are scored by the number of result documents associated with them, as well as the rank of those documents in the result set and the depth of the facet in the hierarchy.

Dash et al. [9] tackle facet selection from the point of view of an analyst who is interested in finding out the “surprising” aspects of the data. Given some baseline expectation on the data (e.g. sales data from a previous year), they display the facets that deviate the most from expectation. Arentz and Öhrn [2] select facets based on a combination of each facet’s a-priori, query independent usefulness and a dynamic, entropy-based measure of information content.

Koren et al. [16] examine means to personalize facet selection to maximize the utility of the selected facets to each individual searcher. We adopt several aspects of their user modeling and notions of utility in this paper, and will further expand on points of similarity in the upcoming sections. Both our paper and [16] can be seen as simplified instantiations of the Probability Ranking Principle for interactive IR as proposed by Fuhr [12]. There, users are presented with various ranked lists of options (search results and facets, in our case), and each user action (e.g. drilldown) is associated with a cost and a benefit. The task is to rank the various lists in a manner that optimizes the expected benefit to users.

2.1 Facet Selection vs. Interactive Query Refinement

The drill-down operation can be seen as the semi-structured counterpart to interactive query refinement [10]. In interactive query refinement, a list of terms - rather than structured constraints - are shown to the user, who then adds a subset of those to the query, causing a change in the set of relevant documents and their rankings. However, there are two pronounced differences between facet drilldown and query refinement that will cause the mathematical machinery of this paper to apply only to the former problem:

- In certain retrieval models, adding terms to the query increases the number of matching documents, whereas the results of a drilldown operation are always a subset of the original result set.
- When adding terms to queries, the relative ranking of documents that appear in both the original and refined result sets may change, whereas the relative order of any two documents that survive the filtering upon facet drilldown remains the same.

3. MODELING THE UTILITY OF FACETS

This section models the interaction between users and faceted search engines, and in particular the effort required
of users to satisfy their information need. Intuitively, a search experience is good whenever the user’s information need is fulfilled quickly. Faceted search enables users to navigate more effectively in the information space induced by their query by providing navigational “short-cuts”. We thus define the effort of a search session as (roughly) the number of results a user must scan before satisfying the information need. Through this modeling of effort, we define the utility – in terms of effort reduction – that a set of displayed facets provides to a user given a particular query.

Let \( D \) denote the document corpus, and let \( F \) denote the set of facets. We denote by \( C : D \rightarrow 2^F \) the classification function that maps each document into a subset of facets. Once a user \( u \) submits a query \( q \) to the search engine, the engine computes a ranked list of documents \( D_q \subseteq D \) according to its relevance function, and furthermore computes a subset of facets \( F_q \subseteq F \). Both facets and the top-ranked search results are returned to the user, in separate lists, each of which ordered by decreasing scores.

Our model of what happens next follows Koren et al. [16] in making the following four assumptions about the users.

**Rationality:** when a user \( u \) issues a query \( q \), there exists a single target document \( d^u_q \) that \( u \) wants to find. This target document is not necessarily known to \( u \) prior to submitting the query; however, \( u \) will be able to recognize it within the search results \( D_q \) returned by the engine. It is implicitly assumed that \( d^u_q \) is indeed returned by the engine, i.e. \( d^u_q \in D_q \), although perhaps it is ranked very deep in the result set.

**Practicality:** upon receiving search results, the user first scans some number \( m \) of top-ranking search results, and finishes the search session if encountering \( d^u_q \) there.

**Omniscience:** if \( d^u_q \) was not present in the top-\( m \) results of \( D_q \), the user examines the list of facets \( F_q \) returned by the engine, and it is further assumed that the user is able to precisely identify which (if any) of the facets in \( F_q \) is associated with \( d^u_q \).

**Optimality:** if some of the displayed facets are indeed associated with \( d^u_q \), the user drills down (in an optimal manner to be modeled shortly) and receives the filtered search results \( D_{q}^{\text{fil}} \); otherwise, the user remains with the original set of results \( D_q \).

The omniscience and optimality assumptions above seem particularly restrictive. However, while we use them to devise our facet selection algorithms, we will significantly relax them in the evaluation phase as our main variant of user simulation will not assume them (see Section 5.3).

Formally, we define the effort of a user \( u \) for the query \( q \) as zero if the engine returned \( d^u_q \) within its top-\( m \)-results. Otherwise, it is the rank of \( d^u_q \) in the list of results at the end of the above process - either the original results \( D_q \) or the filtered results \( D_{q}^{\text{fil}} \). In other words, the effort is the number of documents the user must scan through until encountering the target document whenever that document did not appear in the top-\( m \) search results. Hence, whenever the engine was not successful in answering the need of the user up-front, hopefully it returned facets that can surface the target document via drill-down operations. A good facet selection algorithm will thus display facets that, over many queries and many users, enable the promotion of many target documents to prominent positions in the filtered rankings.

The missing piece in the model above is the manner of drill-down. In the next subsection we present two novel models of user drill-down. First, however, we present some additional notations, that are also summarized in Table 1.

As mentioned earlier, \( D_q \) denotes the set of documents that the engine considers as relevant to query \( q \). For all \( d \in D_q \) let \( r_q(d) \) denote the rank of document \( d \) with respect to query \( q \) (the most relevant document has rank 1). Now, define the random variable \( X \) as the effort invested by a user before reaching the target document. By the above,

\[
X = \begin{cases} 
0 & r_q(d^u_q) \leq m \\
\text{rank of } d^u_q \text{ in } D_q^{\text{fil}} & r_q(d^u_q) > m \text{ and no drilldown} \\
\text{otherwise} & \text{otherwise}
\end{cases}
\]

Let \( p(d = d^u_q) \) denote the probability, taken over all users, of \( d \) being the target document for \( q \). Note that when users are presented with ranked search results only, i.e. without any facets, the expected effort until reaching the target document for query \( q \) is

\[
E[X|q] = \sum_{d \in D_q, r_q(d) > m} p(d = d^u_q) r_q(d)
\]

When users are presented with both search results and a set of facets \( F \), we denote by \( E_M[X|q, F] \) the expected effort until reaching the target document under the drilldown model \( M \). We define the utility of a subset of facets \( F \) with respect to a query \( q \) and a drill-down model \( M \) as the decrease in the value of \( X \) induced by displaying the facet subset \( F \), denoting it by \( U_M^q(F) \):

\[
U_M^q(F \subseteq F) = E[X|q] - E_M[X|q, F]
\]

Finally, we formulate the \( k \)-facet selection problem with respect to query \( q \) and drill-down model \( M \) as:

\[
F_{k}^{OPT}(q, M) = \arg \max_{F \subseteq F : |F| \leq k} U_M^q(F)
\]

where \( k \) is the size of the facet subset to be shown to users.

### Table 1: Summary of Notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( D )</td>
<td>Document corpus</td>
</tr>
<tr>
<td>( D_q \subseteq D )</td>
<td>Documents returned for query ( q )</td>
</tr>
<tr>
<td>( F )</td>
<td>Set of facets available to the engine</td>
</tr>
<tr>
<td>( F_q \subseteq F )</td>
<td>Facets returned by the engine for query ( q )</td>
</tr>
<tr>
<td>( C : D \rightarrow 2^F )</td>
<td>the classification function that maps each document into a subset of facets</td>
</tr>
<tr>
<td>( d^u_q )</td>
<td>Target document of user ( u ) for query ( q )</td>
</tr>
<tr>
<td>( r_q(d), d \in D_q )</td>
<td>Rank of document ( d ) in ( D_q )</td>
</tr>
<tr>
<td>( p(d = d^u_q) )</td>
<td>Probability, over all users, of document ( d ) being the target document for query ( q )</td>
</tr>
<tr>
<td>( X )</td>
<td>Random variable indicating user effort</td>
</tr>
<tr>
<td>( E_M[X</td>
<td>q, F] )</td>
</tr>
<tr>
<td>( U_M^q(F) )</td>
<td>Utility of showing facet subset ( F ) for query ( q ) under drill-down model ( M )</td>
</tr>
<tr>
<td>( F_{k}^{OPT}(q, M) )</td>
<td>Optimal subset of up to ( k ) facets to show for query ( q ) under drill-down model ( M )</td>
</tr>
</tbody>
</table>
3.1 User Drill-Down Models

To recap the discussion above, once facets and search results are retrieved by the search engine for a given query, the user u scans the top m results. If the target document \(d^*_q\) is not found, u scans the displayed facets. Recall the assumption that u identifies the facets associated with \(d^*_q\).

**The Conjunctive Drill-Down Model**

The Conjunctive drill-down model, \(M_C\), presumes a user interface in which a check box is shown next to every facet in the displayed facet list. During facet exploration, the user may drill down into several facets simultaneously by marking their check boxes. Consequently, the new search results will contain only documents that are associated with all checked facets, and furthermore those documents will appear in the same relative order in which they appeared in \(D_q\).

Let \(D_q\) denote the set of facets the engine displays for query \(q\). For any document \(d \in D_q\), define \(F_{q,d} = F_q \cap C(d)\), i.e. the facets associated with \(d\) among those in \(F_q\). For a given target document \(d\) that ranked outside the top-m results, the conjunctive drill-down into \(F_{q,d}\) results in \(d\) being promoted over all documents that were listed above it and that are not associated with all the facets in \(F_{q,d}\). We denote the set of documents that \(d\) is promoted over by

\[
D_{q,d,F_q} = \{d' \in D_q : r(d') < r(d) \land F_{q,d} \not\subseteq C(d')\}
\]

Formally, the utility function for the Conjunctive model is thus defined as:

\[
U^{MC}_q(F_q) = E[X[q] - EM_C[X[q,F_q]] = \sum_{d \in D_q \cap C(d)} |D_{q,d,F_q}|
\]

**The Best Facet Drill-Down Model**

The Best Facet drill-down model, \(M_B\), assumes that when u is searching for \(d^*_q\) and is scanning \(F_q\), u identifies the single facet among \(F_q\) that will promote \(d^*_q\) the most. Hence, u identifies the “best drill-down” option\(^6\). Formally, the utility function of the Best Facet model is defined as follows:

\[
U^{MB}_q(F_q) = E[X[q] - EM_B[X[q,F_q]] = \sum_{d \in D_q \cap C(d)} \max_{f \in F_q_d} |D_{q,d,f}|
\]

where, similar in spirit to the definition above,

\[
D_{q,d,f} = \{d' \in D_q : r(d') < r(d) \land f \not\subseteq C(d')\}
\]

4. SELECTION ALGORITHMS

The previous section formulated facet selection as an optimization problem. However, finding an efficient optimization algorithm under both aforementioned drill-down models is unlikely, as for both models the problem is NP-Hard.

**Theorem 1.** \(F^{OPT}_k(q,M_B)\) and \(F^{OPT}_k(q,M_C)\) are NP-Hard problems.

The proof, which we omit here, is by reduction from the Hitting Set problem [13]. We now discuss two properties of both utility functions, submodularity and monotonicity, that give rise to a greedy \((1 - \frac{1}{e})\)-approximation algorithm for facet selection.

\[\text{This is a refinement of } myopic\ users\ in\ Koren\ et\ al.\ [16]\]

4.1 Submodularity and Monotonicity

**Definition:** Let \(F\) be a set and \(G \subseteq F \subseteq F\) and \(f \in F \setminus F\). A set function \(U : 2^F \rightarrow \mathbb{R}\) is submodular iff

\[
U(F \cup \{f\}) - U(F) \leq U(G \cup \{f\}) - U(G)
\]

Specifically, \(U\) is submodular when for every element \(f \in F\) its incremental value with respect to \(G\) is greater than or equal to its incremental value with respect to \(F\) when \(G \subseteq F \subseteq F\). Furthermore, \(U\) is monotonous non-decreasing iff

\[
\forall G \subseteq F \subseteq F, U(G) \leq U(F).
\]

Many classical problems (e.g. the NP-Hard Max Coverage Problem [13]) can be reduced to finding a subset of size \(k\) of maximal value with respect to a monotonous and submodular set function. An iterative greedy algorithm has been shown to achieve a \((1 - \frac{1}{e})\)-approximation of the optimal solution in non-decreasing, submodular functions [21, 20]. At every iteration, the algorithm adds to the solution the element resulting in maximal incremental gain. The algorithm runs in polynomial time provided that such an element can be identified in polynomial time. We thus show that the utility functions stemming from our two proposed drill-down models are non-decreasing and submodular, and adapt the greedy algorithm to the facet selection problem. A similar approach was recently applied by Chakrabarti et al. [5] in their work on quickselect link.

4.2 Submodularity of Drill-Down Models

We begin by showing that the utility function based on the Conjunctive model is submodular.

**Lemma 1.** The function \(U^{MC}_q(F)\) is non-decreasing and submodular.

**Proof sketch:** we should show that \(\forall G \subseteq F \text{ and } f \in F \setminus F, U^{MC}_q(F \cup \{f\}) - U^{MC}_q(F) \leq U^{MC}_q(G \cup \{f\}) - U^{MC}_q(G)\) and that \(U^{MC}_q(F) \geq U^{MC}_q(G)\).

Consider a set of facets \(S\) that are displayed to user \(u\) with target document \(d^*_q\). Since \(u\) identifies and drills down on all facets in \(S \cap C(d^*_q)\), the utility of \(S\) to \(u\) – i.e. the rank promotion that \(d^*_q\) experiences – is the number of documents ranking above \(d^*_q\) that are not associated with all the facets in \(S \cap C(d^*_q)\). Since \(G \subseteq F\), \(U^{MC}_q(G)\) is trivially non-decreasing. As for submodularity, when a facet \(f\) is added to the set of facets \(S\) already displayed to user \(u\), \(u\) will select facet \(f\) for drill-down in addition to the facets in \(S \cap C(d^*_q)\). Thus, the incremental rank promotion of \(d^*_q\) given \(f \in C(d^*_q)\) with respect to \(S\) is the number of documents that rank above \(d^*_q\) that are associated with all facets in \(S \cap C(d^*_q)\) and not associated with \(f\). Since every document ranking above \(d^*_q\) is associated with all facets in \(F \cap C(d^*_q)\), the incremental value of \(f \not\subseteq F\) with respect to \(G\) is not smaller than its incremental value with respect to \(F\).

**Lemma 2.** The function \(U^{MB}_q(F)\) is submodular and non-decreasing.

**Proof.** Observe that when \(G \subseteq F \subseteq F\), for every query \(q\) and \(d \in D_q\) it holds that

\[
\max_{f \in G_{q,d}} |D_{q,d,f}| \leq \max_{f \in F_{q,d}} |D_{q,d,f}|
\]
hence $U^M_B(F)$ is non-decreasing. Also note that if a user $u$ with a target document $d_u^*$ identifies $f$ as the best drill-down option when presented with the set of facets $F \cup \{f\}$, then $u$ will also drill-down on $f$ when presented with the set $G \cup \{f\}$. Furthermore, in both cases, the drill-down will result in the same rank promotion for $d_u^*$. Thus, the incremental value of $f$ with respect to $F$ is no greater than the incremental value of $f$ with respect to $G$, proving that $U^M_B$ is submodular.

### 4.3 A Greedy Selection Algorithm

Nemhauser et al. [20] show that given a non-decreasing, submodular set function that equals zero on the empty set, a greedy local improvement heuristic achieves, for $k > 1$, a $1 - \left(\frac{1}{k}\right)^k \geq 1 - \frac{1}{k}$ approximation of the optimal subset of size $k$. Both $U^M_B$ and $U^M_C$ satisfy these conditions, hence the following greedy algorithm selects a subset of $k$ facets that is a $(1 - \frac{1}{k})$-approximation of the optimal set (the same algorithm applies to both models):

**Algorithm 1 SelectFacets**

**Input:** a ranked set of search results $D_q$, a distribution $p(d = d_q)$ over $D_q$, an integer $m$, an integer $k$, a classifier function $C$, and a set of facets $F$ of size $> k$

**Output:** An approximately optimal subset of facets $F^*$ such that $|F^*| = k$

1. $F^* \leftarrow \emptyset$
2. for $i = 1$ to $k$ do
3.   $f_i \leftarrow \operatorname{argmax}_{f \in F \setminus F^*} \text{incrementalValue}(f, F^*)$
4.   $F^* \leftarrow F^* \cup \{f_i\}$
5. end for

The selection process begins with an empty set, and iteratively adds the facet generating maximal incremental value with respect to the already selected facets. Note that under each of the models $M_B$ and $M_C$, the incremental value of facets is calculated differently.

The algorithm’s complexity is linear in the time it takes to identify the facet with the highest incremental value at each iteration. Even a naive implementation can accomplish this in $O(|D_q| \cdot |F|)$ per iteration, or $O(|F| \cdot |D_q| \cdot k)$ overall.

**Discussion.** Algorithm 1 attempts to directly optimize the amount of user effort reduced by the selected set of facets. Its objective function guarantees a balance between selective facets, that are associated with few documents but that promote each such document significantly, and facets that cover many documents and hence promote them by moderate amounts. Furthermore, it naturally favors the selection of non-correlated facets, that promote (nearly) disjoint sets of documents. In contrast, previous work [17, 2, 16] did not attempt to directly optimize user effort; rather, those works proposed facet selection heuristics based on simple statistical properties of the documents-facets association that were thought to be correlated with positive user experience.

Note that Algorithm 1 requires the distribution of target documents $p(d = d_q)$ over $D_q$. The engine doesn’t know this distribution; however, its score function represents its best estimate for the needs of the users, and may take into account usage statistics such as clicks on results. Therefore, the engine can infer a distribution that hopefully is “close enough” to the true, underlying distribution $p(d = d_q)$.

### 5. EXPERIMENTAL SETUP

This section presents an automatic and scalable evaluation of the ability of facet selection schemes to quickly surface (promote) relevant documents that the search engine failed to return in its top-10 results for a query.

Our evaluation is based on a set of 1000 randomly selected queries submitted to a major commercial search engine (denoted by $E$), each returning at least 100 search results. The documents in $E$’s index (and hence its search results) are each classified - with various confidence levels - into a business-oriented taxonomy of 6000 nodes as described in [4]. We trimmed this taxonomy to its top two levels, resulting in 230 categories. On average, each document was classified into 3.34 categories, and we considered (up to) the five categories of highest confidence per document as its facets.

We use the search results returned by engine $E$ for the query $q$ to postulate a probability distribution $p(d = d_q)$ of target documents, and use this distribution to perform facet selection by several schemes (Section 5.1). To measure the effectiveness of each facet selection scheme, we model user intent in the form of test documents - documents (relevant to the query) that are chosen independently of the postulated distribution of target documents used for the selection process (Section 5.2). We then simulate several flavors of faceted search sessions (5.3) and compute the promotion of the test documents induced by each set of facets (5.4).

While the association of facets to documents uses proprietary data, the rest of the experimental protocol is reproducible by accessing public interfaces of Web search engines.

#### 5.1 Selection Schemes

We evaluate four facet selection schemes - two schemes resulting from applying Algorithm 1 to the Conjunctive model $M_C$ and the Best-Facet model $M_B$, and two baseline algorithms (see below).

For both $M_B$ and $M_C$, we set $m = 10$, and postulate the following Zipfian distribution to estimate the target documents’ identities: $p(d_j = d_q) = \gamma \cdot j^{-2}$, where, for $j = 11 \ldots 100$, $d_j$ is the document returned by engine $E$ at rank $j$ for query $q$, and $\gamma$ is a normalization constant. This distribution conveys the belief of engine $E$ that documents it ranks higher are more likely to be targeted by users.

We compare $M_B$ and $M_C$ to two baseline models:

- **The Weighted Residual Coverage** scheme $B_{RC}$ weights documents according to the postulated distribution $p(d = d_q)$ described above. It then selects facets in iterations, where each iteration greedily selects the facet associated with the highest probability mass of documents that were not yet covered, i.e., are not associated with any of the facets selected in the previous iterations. This is closely related to the **Most Probable** selection heuristic presented in [16].

- **The Greedy Count** scheme $B_{GC}$ ranks facets simply according to the number of top-100 documents in $D_q$ they are associated with (the more, the better). This is equivalent to the **Most Frequent** selection heuristic in [16].

We note that for all four schemes, considering the top-1000 documents did not produce qualitatively different results.

#### 5.2 Modeling User Intent

This section defines the documents on which we will test the effectiveness of the four selection schemes. As facets are mainly useful for surfacing documents that do not appear
in the top search results, these test documents should (1) be relevant and represent valid user intents, while (2) not appearing too high in E’s search results. A third, methodologi- cal requirement is to model user intent independently from the distribution $p(d = d_q)$, postulated by the engine and used for the selection process.

We thus used the top results of a different commercial search engine, $E^*$, as surrogates for user intents. Specifically, we submitted each of the 1000 queries in our sample to $E^*$, and considered its top-10 results per query to be potential target documents. We then intersected those documents with documents ranked 11 through 1000 by E to define the test documents for each query.

Formally, denote the set of test documents for query $q$ by $D_q^{test}$, and by $D_q^E(i \ldots j)$ the results ranked in the range $i \ldots j$ by engine $E$ for the $q$. By these notations,

$$D_q^{test} = D_q^{E^*}(1 \ldots 10) \cap D_q^E(11 \ldots 1000)$$

(2)

As our evaluation is based on surfacing target documents using facets, the definition above ensures that the test documents exist in E and can (hopefully) be promoted. Since for some queries this resulted in $D_q^{test} = \emptyset$, we were left with 810 queries having at least one test document. All test documents per query are considered equally probable in our ensuing analysis, regardless of their ranks in either $E^*$ or $E$.

We note that repeating our experiments with two alternatives for $E^*$ had no qualitative difference on the outcome.

5.3 Simulating Faceted Search Sessions

Once facets are selected for the 810 test queries, our experiments proceed by simulating search sessions according to three user drill-down strategies. Each strategy models the behavior of a user $u$ with target document $d_q^u \in D_q^{test}$.

The main drill-down model we consider is the Probabilistic Strategy $S_P$. It models user drill-down on a displayed set of facets as a random process guided by two principles: (1) users are influenced by the order in which facets are presented, and (2) while users do not have perfect knowledge of the facets associated with their target documents, they tend to select facets associated with their information need with higher probability than facets not associated with it. Thus, this drill-down strategy relaxes the omnisciency and optimality assumptions of Section 3 on which selection models $M_B$ and $M_C$ are based.

In the random process, $u$ cyclically scans the selected facets, in the order by which they were ranked and presented by the selection algorithm. Upon any scan of facet $i$, $u$ drills down with probability $\alpha$ if $i$ is associated with $d_q^u$, and with probability $\beta < \alpha$ if $i$ is not associated with $d_q^u$. This process continues until $u$ ultimately drills down, yielding a drill-down distribution over the displayed facets. The utility from drilling down into each facet is weighted by its associated probability. For example, if $u$ is shown the set of facets $f_1, f_2, f_3$ and $C(d_q^u) = \{f_1, f_3\}$, then $u$ drills down into $f_1$ with probability proportional to $\alpha$, into $f_2$ with probability proportional to $(1-\alpha)\beta$, and into $f_3$ with probability proportional to $(1-\alpha)(1-\beta)\alpha$. In our experiments we set $\alpha$ to 0.8, and $\beta$ to 0.1.

In addition to $S_P$, we simulate two other drill-down strategies - the Conjunctive Strategy $S_C$ and the Best Facet Strategy $S_B$. These strategies follow the assumptions our selection schemes are based upon, that users precisely identify which (if any) of the displayed facets is associated with $d_q^u$.

and act accordingly in an optimal manner. Specifically:

The Conjunctive Strategy $S_C$ assumes user $u$ simultaneously drills-down into all facets associated with $d_q^u$ as described in Section 3.1. Intuitively, selection scheme $M_C$ should perform well under this strategy.

The Best Facet Strategy $S_B$ simulates $u$ drilling-down on the single facet that maximizes the rank promotion of $d_q^u$. Intuitively, selection scheme $M_B$ should perform well under this strategy.

5.4 Utility of Selected Facet Sets

We measure the utility of a displayed set of facets within each simulation using two metrics. The first, following Equation 1 and denoted by $U_{avg}$, is the average rank promotion of the test documents induced by the set of facets. Essentially, selection schemes $M_B$ and $M_C$ optimize for this measure.

We also evaluate the simulations by the number of test documents promoted into the top-10 results after drill-down, denoted by $U_{top-10}$. This reflects the fact that users rarely scan beyond the top-10 search results, and that promoting a target document helps only if it then appears on the first page of results. Note that this utility function is unrelated to what selection schemes $M_B$ and $M_C$ optimize for.

5.5 Evaluation Methodology

Our evaluation methodology is inspired by Radlinski et al. [22]. We perform pairwise comparisons of the four selection schemes $\{M_B, M_C, B_{RC}, B_{GC}\}$, under the three drill-down strategies $\{S_B, S_C, S_P\}$ and the two utility measures $\{U_{avg}, U_{top-10}\}$ - 6 x 3 x 2 comparisons overall. Given a drill-down strategy and a utility measure, and for a pair of schemes $A$ and $B$, denote by $Q_{A>B}$ the number of queries where the facets selected by $A$ had higher utility than those selected by $B$, and by $Q_{A<B}$ the number of queries where both schemes performed equally well. For every pair of schemes, we examine the quantities $\{Q_{A>B}, Q_{B>A}, Q_{A=B}\}$ and perform a Pearson’s Chi-Square test [6] to evaluate whether the gap $|Q_{A>B} - Q_{B>A}|$ is statistically significant.

6. RESULTS

We evaluate our proposed facet selection algorithms by comparing them to the two aforementioned baseline selection schemes. The comparisons measure the utility of the various algorithms in simulated faceted search sessions. Section 6.1 examines the performance of the algorithms when selecting a set of five facets. Section 6.2 then examines the sensitivity of the results to other sizes of facet sets. Section 6.3 examines whether the empirical behavior of our proposed algorithms match their theoretical approximation guarantees.

6.1 Selection Scheme Comparison for k=5

We begin by presenting a pairwise comparison of the four selection schemes as shown in Figures 1-3. Each of the 12 plots compares a pair of schemes, as judged by a specific utility function, under all three user drill-down strategies. In all plots, the y-axis is the percentage (out of 810 queries) achieved by the quantities $Q_{A>B}, Q_{B>A}$ and $Q_{A=B}$. A check sign above each experiment (i.e. drill-down strategy) indicates that the difference $|Q_{A>B} - Q_{B>A}|$ is statistically significant for a significance level of 5%.
Figure 1: Model Based Selection Schemes $M_C$ and $M_B$ vs. Weighted Residual Coverage Baseline Scheme $B_{RC}$

Figure 2: Model Based Selection Schemes $M_C$ and $M_B$ vs. Greedy Count Baseline Scheme $B_{GC}$

Figure 3: Comparing Model Based Selection Schemes ($M_C$ to $M_B$, left) and Baseline Selection Schemes ($B_{RC}$ to $B_{GC}$, right)
First, from Figure 1 we observe that the model-based selection schemes $M_C$ (left) and $M_B$ (right) outperform the baseline Weighted Residual Coverage scheme $B_{RC}$ under all drill-down strategies with respect to both evaluation metrics. The major difference between the model-based schemes and $B_{RC}$ is that the latter considers only the documents covered by each facet, disregarding the potential new ranked list of results induced by each facet. Furthermore, $B_{RC}$ favors facets that cover many documents for which $p(d = d_k)$ is high, i.e. high-ranking documents according to the postulated distribution $p(d = d_k)$. However, $B_{RC}$ ignores the risk that drilling down into such facets may yield search results that are quite similar to the original list. On the other hand, the “result-oriented” $M_B$ and $M_C$ schemes recognize the low utility of “dense” facets when considering the rank promotion induced by the drill-down operation.

Figure 2 compares the Greedy Count baseline scheme $B_{GC}$, with $M_C$ (left) and $M_B$ (right). The results here are not as clear-cut. We first observe, from the bottom two plots, that $M_C$ and $M_B$ outperform $B_{GC}$ under all drill-down strategies with respect to the $U_{top-10}$ utility measure. Furthermore, from the rightmost comparison in each of the four plots, we see that $M_C$ and $M_B$ outperform $B_{GC}$ under the Probabilistic drill-down strategy $S_P$ regardless of utility measure. Note that both $U_{top-10}$ and $S_P$ are, in a sense, foreign to $M_B$ and $M_C$ that optimize for $U_{avg}$ and drilldown strategies $S_B$ and $S_C$ respectively. Surprisingly, however, the Greedy Count baseline scheme outperforms $M_B$ and even $M_C$ under the Conjunctive drill-down strategy $S_C$ when evaluated by average rank promotion $U_{avg}$. The comparisons $B_{GC}$ vs. $M_B$ and $B_{GC}$ vs. $M_C$ under the Best-Facet drill-down strategy $S_B$ are inconclusive when evaluated by $U_{avg}$.

Figure 3 (top-left) shows that $M_C$ outperforms $M_B$ under the $U_{avg}$ utility function and both $S_C$ and $S_B$ drill-down strategies. While $M_C$ winning under $S_C$ is hardly surprising, the fact that $M_C$ outperformed $M_B$ under the best-facet drill-down strategy $S_B$, on which $M_B$ is based, is surprising. When measuring utility according to $U_{top-10}$ (bottom-left plot), $M_B$ outperformed $M_C$ under both the $S_B$ and $S_P$ strategies. No significant differences were detected under the $U_{avg}$ utility function and the Probabilistic strategy $S_P$ and under $U_{top-10}$ with the Conjunctive strategy $S_C$.

Finally, Figure 3 (right) shows that in the battle of baselines, $B_{GC}$ outperforms $B_{RC}$ in four of six comparisons, with one test being inconclusive and with $B_{RC}$ outperforming $B_{GC}$ only once. This is consistent with the findings in [16].

### 6.2 Effect of the Number of Selected Facets

This section examines the relative performance of the various selection schemes as the number of selected facets per query, $k$, varies from 1 to 10. Table 2 compares the Conjunctive model $M_C$ to the baseline schemes $B_{RC}$ and $B_{GC}$.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$U_{avg}$</th>
<th>$U_{top-10}$</th>
<th>$U_{avg}$</th>
<th>$U_{top-10}$</th>
<th>$U_{avg}$</th>
<th>$U_{top-10}$</th>
<th>$U_{avg}$</th>
<th>$U_{top-10}$</th>
<th>$U_{avg}$</th>
<th>$U_{top-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>2</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>3</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>4</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>5</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>6</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>7</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>8</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>9</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>10</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
<td>$M_C$</td>
<td>$M_B$</td>
</tr>
</tbody>
</table>

Table 2: Comparing the Conjunctive Selection Scheme $M_C$ to Baseline Selection Schemes $B_{RC}$ and $B_{GC}$ with Varying $k$ Under All User Strategies and Both Utility Functions (table entry indicates winning scheme)

<table>
<thead>
<tr>
<th>$k$</th>
<th>$S_C$</th>
<th>$S_B$</th>
<th>$S_P$</th>
<th>$S_C$</th>
<th>$S_B$</th>
<th>$S_P$</th>
<th>$S_C$</th>
<th>$S_B$</th>
<th>$S_P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>2</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>3</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>4</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>5</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>6</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>7</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>8</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>9</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
<tr>
<td>10</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
<td>$M_B$</td>
</tr>
</tbody>
</table>

Table 3: Comparing the Best Facet Selection Scheme $M_B$ to Baseline Selection Schemes $B_{RC}$ and $B_{GC}$ with Varying $k$ Under All User Strategies and Both Utility Functions (table entry indicates winning scheme)
Table 3 compares the Best Facet model $M_B$ to the baselines. The entries in each table indicate the selection scheme that performed better for a specific (drill-down strategy, utility function) pair; all results are statistically significant unless where indicated by “(NS)”.

The results reinforce those reported in Section 6.1 for $k = 5$. First, the left halves of both tables show that $M_B$ and $M_C$ outperform $B_{GC}$ in a statistically significant manner under all drill-down strategies and utility functions for all examined values of $k$. This mirrors the plots in Figure 1.

Second, in a manner consistent with the findings of Figure 2, the right halves of the tables shows that both $M_B$ and $M_C$ outperform the Greedy Count scheme $B_{GC}$ under the Probabilistic drill-down strategy $S_P$ and both utility function, as well as under the $U_{top10}$ utility function and all three drill-down strategies. Of the 80 corresponding comparisons, 76 are statistically significant. Third, as in the previous section, $B_{GC}$ outperforms both $M_B$ and $M_C$ (for $k \geq 3$) under the Conjunctive drill-down strategy $S_C$ under the $U_{avg}$ utility function. When coupling $U_{avg}$ with the Best-Facet drill-down strategy $S_B$ (4th column from the right), $M_B$ and $M_C$ perform roughly the same as $B_{GC}$ - 7 of the 10 results in each table are not statistically significant.

Overall we do not see sensitivity to the number of selected facets in the relative performance of the four tested facet selection schemes, at least for the range $k = 1, \ldots, 10$.

6.3 Revisiting Approximate Optimality

Section 4 proved that selection schemes $M_B$ and $M_C$ are approximately optimal for the utility function $U_{avg}$ under certain assumptions on users’ information needs and drill-down behavior. However, our experimental framework relaxed those assumptions, and so the theoretical approximation ratio of $1 - \frac{1}{2}$ is not guaranteed to hold.

The first assumption that does not hold in our experiments concerns the distribution over target documents. $M_B$ and $M_C$ select facets under the postulated Zipfian distribution over the top-100 results of engine $E$ as described in Section 5.1. However, utility is experimentally measured on a uniform distribution over all test documents that may not appear in $E$’s top-100 results, as explained in Section 5.2. Second, while $M_B$ is approximately optimal for drill-down strategy $S_B$ (and similarly $M_C$ for $S_C$), we test each scheme under all three drill-down strategies. In particular, drill-down strategy $S_P$ violates the omniscience and optimality assumptions underlying both schemes.

To measure how well $M_B$ and $M_C$ approximated an optimal algorithm OPT that is somehow aware of the true user needs, we denote by $F_{Diff}$ the facets associated with any of the test documents for $q$. This set contains at most 50 facets, as we consider at most 5 facets for each of at most 10 test documents. OPT enumerate all subsets of size 5 of $F_{Diff}$, and for each drill-down strategy, selects the subset that maximizes the utility under $U_{avg}$. In the case of $S_P$, where the order of facets matters, OPT examines ordered subsets. We then averaged the ratio of utilities achieved by $M_B$ (resp. $M_C$) and OPT over the 810 test queries. The results are shown in Table 4.

The results are consistent with those reported in the top-left plot of Figure 3 - $M_C$ outperforms $M_B$ with respect to $U_{avg}$. Observe that under drill-down strategies $S_B$ and $S_C$, both selection schemes are better than $1 - \frac{1}{2} = 0.632$ optimal, with each scheme slightly favoring its corresponding drill-down strategy ($S_B$ for $M_B$ and $S_C$ for $M_C$). Not surprisingly, the results under $S_P$, which deviates the furthest from the models’ assumptions, are lower. We did not check the corresponding ratios under $U_{top10}$ since that utility function is not submodular.

7. CONCLUSIONS AND FUTURE WORK

This paper addressed the problem of selecting the set of facets to be displayed alongside search results from a utility perspective, aiming to minimize the effort of users in finding their target documents. We modeled the manner by which users interact with faceted search engines, and proposed efficient algorithms that select approximately optimal facets under those models, as selecting the optimal set is NP-Hard.

Simulations of faceted search sessions for hundreds of queries showed that our algorithms outperform baseline selection algorithms. In particular, this holds even when users are modeled as less knowledgeable than what our algorithms assume, as well as for a utility function we do not optimize for, that only considers a drill-down successful if the test document was surfaced in its top-10 search results.

The following directions are left for future work. First, our current selection algorithms do not explicitly consider the hierarchical relationships among facets. For example, the algorithm might output both a facet and its ancestor, despite its preference to select non-correlated facets. We plan to study various aspects of adding hierarchical constraints to the set of selected facets. Second, from an implementation point of view, the main requirement of our algorithms was to obtain a list of 100 top-ranking documents per query, along with the facets associations with each of those documents. We plan to tackle the performance implications of this requirement in large scale, distributed search engines operating under heavy query loads. Third, we plan to investigate additional user models and utility functions under which the facet selection problem can still be efficiently approximated. The utility function $U_{top10}$: e.g., is not submodular and approximating it remains a challenge.

Acknowledgments

We thank Ravi Kumar and Andrew Tomkins for valuable discussions throughout the course of this work.

8. REFERENCES


Table 4: Average Utility Ratios of $M_B$, $M_C$ to OPT for each Drill-Down Strategy

<table>
<thead>
<tr>
<th>Strategy</th>
<th>$S_B$</th>
<th>$S_C$</th>
<th>$S_P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_B/OPT$</td>
<td>0.6475</td>
<td>0.6465</td>
<td>0.5549</td>
</tr>
<tr>
<td>$M_C/OPT$</td>
<td>0.6605</td>
<td>0.6694</td>
<td>0.5626</td>
</tr>
</tbody>
</table>


