

Facet Selection Algorithms for Web Product Search

Damir Vandic¹
vandic@ese.eur.nl

Flavius Frasincar¹
frasincar@ese.eur.nl

Uzay Kaymak²
u.kaymak@tue.nl

¹Erasmus University Rotterdam, P.O. Box 1738, 3000 DR Rotterdam, the Netherlands

²Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, the Netherlands

ABSTRACT

Multifaceted search is a commonly used interaction paradigm in e-commerce applications, such as Web shops. Because of the large amount of possible product attributes, Web shops usually make use of static information to determine which facets should be displayed. Unfortunately, this approach does not take into account the user query, leading to a non-optimal facet drill down process. In this paper, we focus on automatic facet selection, with the goal of minimizing the number of steps needed to find the desired product. We propose several algorithms for facet selection, which we evaluate against the state-of-the-art algorithms from the literature. We implement our approach in a Web application called *faccy.net*. The evaluation is based on simulations employing 1000 queries, 980 products, 487 facets, and three drill down strategies. As evaluation metrics we use the average number of clicks, the average utility, and the top-10 promotion percentage. The results show that the Probabilistic Entropy algorithm significantly outperforms the other considered algorithms.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Information filtering

General Terms

Algorithms, Experimentation

Keywords

facet selection, product search, information retrieval

1. INTRODUCTION

Online product search has nowadays become more important than ever, as consumers purchase more often on the Web [3]. One explanation for this is that the Web facilitates the user in finding products that better match their

needs and that it offers more variation than traditional selling channels. Not only do the users have access to more information (e.g., user reviews, exact product information), they also find it easier to shop from their homes. On the other hand, because of the many options, users are often overwhelmed and find it difficult to browse through the available products.

Multifaceted search, also sometimes referred to as ‘guided navigation’, is a popular interaction paradigm that allows users to navigate through multidimensional data [1]. One of the main uses of multifaceted search is in the domain of e-commerce, i.e., Web shops. It is being employed to solve the parametric product search problem for Web shops that have collected local offerings and product information. For example, in a Web shop the user might enter a query like ‘samsung, gps’ in order to search for a Samsung phone that has built-in GPS capabilities. After showing the initial result set, most Web shopping interfaces display the facets of the products in the result set, which can be used to further drill down into the results set. The facets in this case are product attribute/value combinations. Some examples of such product facets are ‘connectivity:hspda’ and ‘screen size:3-4 inch’.

An important problem of multifaceted search is the selection of facets that should be displayed for each query. Because products have so many attributes that could be displayed as facets, Web shops usually have some static business logic to display certain facets for each result set. Although this works for local Web shops that do not have many product categories, the creation of this business logic is a time consuming process and is not appropriate for Web-wide product search. One solution to this problem is to employ an optimized facet selection process. The goal of such an optimization process is to show facets that effectively partition the product search space so that the user can easily drill down and find its desired product. In literature, this is referred to as the *facet selection* problem, which can be expressed as the optimization of a hyperactive media link generation process [2].

In this paper, we propose new algorithms for the facet selection problem in product search. We evaluate several approaches and compare our proposed algorithms against several state-of-the-art facet selection algorithms from the literature. Our proposed algorithms aim to partition the space in the most effective manner and thus allow the user to drill down in the least amount of time. We perform the evaluation on a large data set and analyze the results, differently from previous works, across three different measures.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.
Copyright 2013 ACM 978-1-4503-2263-8/13/10 ...\$15.00.
<http://dx.doi.org/10.1145/2505515.2505664>.

2. RELATED WORK

The main goal of this paper is to analyze the performance of several facet selection algorithms in a product search environment. In the literature, facet optimization is a field that has gained popularity in the last few years [4, 8, 11, 13]. The author of [5] shares her experiences from experiments on (faceted) user interface design, as part of UC Berkeley’s Flamenco Search Interface project. Most of the studies investigate the optimization of the facet selection process. However, there is not much research that explores this topic in an e-commerce environment.

The authors of [7] aim to make facet optimization personalized, a solution that should be able to deal with the large variety in user preferences towards facet drill down approaches. From this work we adopt several aspects of the used user modeling. The main idea is to have a collaborative filtering and personalization of the search interface, with respect to the user’s behavior. The authors propose a utility-based framework, which we also adopt in this paper. The authors of [9] investigate how facet optimization can be performed for general-purpose Web documents. Three different strategies for simulating a faceted search session are employed in the evaluation. In this paper, we benchmark our proposed methods against those presented in [9], using the same simulation strategies. The reason for this is that the strategies cover a wide range of type of users and that it allows for a more fair comparison than we would have come up with new simulation strategies.

Similar to [7] and [9], which are directly related to our work, there are other studies that have investigated the use of facets in information retrieval. The authors of [14] focus on column-wise faceted search. This approach allows the user to drill down in a hierarchical fashion, going from left to right. The authors claim that this has an advantage as users can make use of both directional and non-directional faceted search. We do not include this approach in our comparisons as we do not focus on directed faceted search.

The authors of [6] investigate how the use of Semantic Web technology can aid in multi-faceted product search. The authors propose an information search and retrieval framework based on a semantically annotated multi-faceted product family ontology. The key idea in their approach is to use a document profile model that suggests semantic tags for the annotation process. Because the approach presented by these authors requires a product family ontology, we did not include it in our comparisons (our approach is more flexible because we do not require product family ontologies).

The study presented in [10] focuses on an approach where both the querying and navigation of Web documents are considered as equally important activities. In this paradigm, querying is seen as a way to identify the starting points for navigation, and navigation is guided based on information obtained from the query. The authors present a formal model for this paradigm and report empirical results obtained from experiments over a large Web corpus. The results show that in the case of ambiguous queries, the proposed retrieval model identifies good starting points for post-query navigation. However, for queries that are less ambiguous, the output tends to match that of a conventional retrieval system. Because this approach deals with Web documents and hyper-links, we exclude it from the comparisons, as our approach does not involve hyper-links but instead just uses directly product/facet associations.

3. FACET SELECTION

The goal of this paper is to reduce the search effort of a user that is searching for a product that meets his needs. In this section, we first give the formal problem formulation and then we present the considered facet selection algorithms. The problem statement that we present here is based on previous works [7, 9]. We assume that the number of results scanned by a user (before finding his desired product) represents the search effort. Let \mathcal{D} denote the set of all products, \mathcal{F} the set of all facets, and $C : \mathcal{D} \rightarrow 2^{\mathcal{F}}$ the mapping of each product to a subset of facets. The main use case that we consider is that a user submits a query q to the product search engine. Next, the search engine computes a ranked list of products $D_q \subseteq \mathcal{D}$ and a set of facets $F_p \subseteq \mathcal{F}$ of size l that are to be displayed. Furthermore, the set $F_q \subseteq \mathcal{F}$ represents all facets that belong to all products in D_q .

Similar to [7], we incorporate four assumptions about the user in the considered simulation strategies for the evaluation. First, we assume that for each user u , who has submitted query q , there exists a single target product d_q^u that fulfills the user’s needs. This target product is assumed to always be present in the initial result set D_q , but can be ranked very low. Second, the user ends the search session if it finds d_q^u in the top- m results. Third, a user exactly knows which facets (if any) from F_p are associated to d_q^u . Last, if there are any facets that are associated with d_q^u , we assume that the user then also selects these.

In this paper, we also assume that multiple clicks (drill downs) can occur. More specifically, we assume that the above described process can repeat itself a maximum of k times (iterations). If the user finds the desired product in the top- m results in less than k iterations, then the search session ends prematurely, otherwise it ends after k iterations. If we let \mathcal{D} , \mathcal{F} , C , u , and q remain unchanged, then the result set at any iteration can be denoted as $D_{q,S}$, where $S \subseteq F_q$ represents the previously selected facets. Similarly, the proposed facets by the search engine at any iteration are denoted by $F_{p,S}$, where $F_{p,S} \subseteq F_q$.

We now express the reduction in search effort in terms of the search engine proposed facets $F_{p,S}$. For this purpose we adopt the approach taken in [9] and modify it to allow for multiple clicks in the search process. We can define the *utility* of displaying a set of facets $F_p \subseteq \mathcal{F}$, proposed by a facets optimization approach M , with respect to a query q and a set of already selected facets S , as following:

$$\begin{aligned} U_{q,S}^M(F_p) &= E[X|q, S] - E_M[X|q, S, F_p] \\ &= \sum_{\substack{d \in D_{q,S} \\ r_q^S(d) > m}} p(d = d_q) r_q^S(d) - E_M[X|q, S, F_p] \end{aligned}$$

where $E[X|q, S]$ represents the expected search effort of a user that does not click on facets, $E_M[X|q, S, F_p]$ represents the search effort using the previously described drill down process with the facets proposed by approach M , X denotes the random variable that represents the search effort of a user for one click, $r_q^S(d)$ denotes the rank of d in the result set, and $p(d = d_q)$ is the probability of d being the target product for query q (taken over all users).

Using this definition, we can formulate the k facet selection problem as:

$$F_{p,S,M}^* = \arg \max_{\substack{F_p \subseteq \mathcal{F} \\ |F_p| < k}} U_{q,S}^M(F_p) \quad (1)$$

where k is the number of facets that are shown to the users. The optimization from Equation 1 is NP-hard (reduction from the Hitting Set problem [9]) and therefore it is difficult to have an exact optimal solution to this problem. The optimization problem can be solved by an iterative greedy algorithm that achieves a $(1 - 1/e)$ -approximation of the optimal solution [9].

3.1 Utility computation approaches

We consider four utility computation approaches in our study. Combining these with the previously described utility-maximization approach, we obtain four different facet selection algorithms. Two of these approaches have been proposed in [9] and the remaining two in this paper.

Best Facet I. Proposed by the authors of [9], this approach assumes that a user is fully aware of all facets that belong to d_q^u . Furthermore, it is assumed that, from the list of presented facets, the user selects the facet that promotes the target product the most, i.e., results in having the target being ranked higher in the filtered result set. Formally, this approach is denoted by M_{BI} and is defined as:

$$\begin{aligned} U_{q,S}^{M_{BI}}(F_p) &= E[X|q, S] - E_{M_{BI}}[X|q, S, F_p] \\ &= \sum_{\substack{d \in D_{q,S} \\ r_q^S(d) > m}} p(d = d_q) \max_{f \in F_{p,S,d}} |D_{q,S,d}^f| \end{aligned}$$

with

$$\begin{aligned} F_{p,S,d} &= F_{p,S} \cap C(d), \text{ and} \\ D_{q,S,d}^f &= \{d' \in D_{q,S} : r_q^S(d') < r_q^S(d) \wedge f \notin C(d')\} \end{aligned}$$

We can see that the utility is computed by iterating over each product that is not in the top- m results and multiplying the probability that it is the target with the maximum number of products that can be skipped (by selecting the ‘best facet’). The set $F_{p,S,d}$ represents the facets that are both in the presented facets list and associated with the target product.

Conjunctive. The Conjunctive utility computation approach is also proposed by the authors of [9]. Similarly to the Best Facet I approach, it also assumes that a user is fully aware of all facets that belong to d_q^u . The difference with the Best Facet I approach is that this approach assumes that the user does not select the best facet that promotes the target product the most, but that (s)he selects *all* the facets that belong to the target product. Formally, the utility is defined as:

$$\begin{aligned} U_{q,S}^{M_C}(F_p) &= E[X|q, S] - E_{M_C}[X|q, S, F_p] \\ &= \sum_{d \in D_{q,S} \wedge r_q^S(d) > m} p(d = d_q) \cdot |D_{q,S,d}^{F_{p,S,d}}| \end{aligned}$$

with $F_{p,S,d}$ remaining unchanged and

$$D_{q,S,d}^{F_{p,S,d}} = \{d' \in D_{q,S} : r_q^S(d') < r_q^S(d) \wedge F_{p,S,d} \not\subseteq C(d')\}$$

Best Facet II. In order to investigate the influence of the ranking on the suggested facets, we propose to modify the Best Facet I approach. In the Best Facet II approach, we do not maximize over the number of products that is skipped for each facet. Instead, we maximize over the *weighted* number

of products that is skipped. The utility is then defined as:

$$\begin{aligned} U_{q,S}^{M_{BII}}(F_p) &= E[X|q, S] - E_{M_{BII}}[X|q, S, F_p] \\ &= \sum_{\substack{d \in D_{q,S} \\ r_q^S(d) > m}} \max_{f \in F_{p,S,d}} p(d = d_q) \cdot |D_{q,S,d}^f| \end{aligned}$$

Probabilistic Entropy. We also investigate a hybrid method that combines an entropy-based approach and the utility-maximization approach using the Conjunctive utility computation. It aims to improve one important aspect of the utility maximization approaches, namely the strong dependency on the ranking mechanism of the product search engine. The ranking provided by the search engine may not be optimal with respect to the desired product, while the Conjunctive and Best Facet I approaches rely heavily on the ranking. These algorithms focus on the promotion of products that are not the desired one, under the assumption that a highly ranked product is the desired one. The Probabilistic Entropy algorithm escapes these local minima by selecting more facets that are possibly better facets than those of the highly ranked products.

With a probability of $\alpha \in (0, 1)$, the Probabilistic Entropy approach returns the same results as the Conjunctive approach. With a probability β , with $\beta < \alpha$, the Entropy approach (covered later on) is used to select the optimal facets. In this way, the local minima are likely to be escaped, as there is an alternative method that is being used that does not rely on the product ranking.

Probabilistic Conjunctive. This approach is similar to the Probabilistic Entropy. It uses the utility-maximization approach with the Conjunctive utility computation with a probability of α . With a probability of β , with $\beta < \alpha$, instead of using the Entropy approach like in the Probabilistic Entropy approach, it uses the Random facet approach (discussed later on).

Weighted Residuals Coverage. There are several baseline approaches that we consider in our evaluation. The first baseline approach is the Weighted Residuals Coverage algorithm and is also used in [9]. It is closely related to the *Most Probable* selection heuristic presented in [7]. The main idea is to use the postulated distribution $p(d = d_q)$ to determine the ‘importance’ of facets. The algorithm iteratively greedily selects the facet that occurs the most, weighted by the postulated distribution $p(d = d_q)$. An important aspect is that once a facet is chosen in an iteration, the weighted importances are updated by using only products that are not covered by the previously selected facets. In this way, the algorithm selects facets that cover mostly all of the higher ranked products.

Greedy Count. This baseline facet selection algorithm is also considered in [9]. It ranks the facets to be selected according to the number of top- x products in the result set and is similar to the *Most Frequent* selection heuristic from [7]. In [9], the parameter x is set to 100. The Greedy Count approach favors more popular facets and is likely to result in many drill downs, as the total number of products that will be filtered with each click will be relatively small.

Entropy. The Entropy baseline approach considers the problem more from an information theoretical perspective, using Shannon’s entropy [12]. In our case, the random variable is a binary one, indicating whether we know that a certain product is the target product or not. For each facet, we set $P(\text{yes})$ to the percentage of products that have the facet

and $P(\text{no})$ to the percentage that do not have the facet. The procedure for selecting the optimal set of facets is a greedy one, where in each iteration the facet with the highest entropy is added to the set of facets that is being displayed to the user.

Random. In order to put the results in perspective, we include the Random facet selection algorithm. This approach takes all possible facets F_q and selects randomly k facets that are then displayed to the user. The choice of the facets is random but nevertheless dependent on the query q and previously selected facets S .

4. EVALUATION

In this section, we first discuss the used data set and the experimental design, before presenting the evaluation results. We use a data set that is gathered from Kieskeurig.nl, the largest price comparison site in the Netherlands. This service does not only provide price comparisons, but also has very detailed information on products. For this evaluation, we focused on consumer electronics and chose mobile phones to be the category of products that we use in the experiments. The data set contains 980 products for which we have key/value pairs, i.e., product attributes. All product information is in Dutch, but should be understandable also for non-Dutch speaking people because of the frequently used English terminology in the product attributes.

Using the product attributes, we created the facets using the following rules. A facet is a combination of a product attribute and a value (or range of values). Binary product attributes, such as ‘GPS’, were mapped to one facet that was in the form of ‘GPS:yes’. For product properties that represent multivalued qualitative values, such as ‘Supported Video formats’, we created a binary facet for each value. Similarly, for single-valued qualitative product attributes, we created a single facet for each value. For all the quantitative properties, we manually defined the ranges that would represent the different facets. As a result of this facet creating process, we obtained 487 facets for the 980 products. The size and variety of this data set allows for a thorough evaluation of the facet selection algorithms that we propose in this paper.

Experiments. For the evaluation, we simulate a user that is in a faceted search session. There are two aspects that are important in this type of simulation. First, we need a way to generate queries that are sufficiently realistic for the experiments. Second, we need one or more simulation strategies of users in order to simulate the clicking on a facet. Before we go into the details of these two aspects, let us first explain on a high level how we have designed the simulation.

Given a query, we submit it to the product search engine, after which, for every product that we consider as a possible target product, we simulate a faceted search session. In this paper, the set of possible target products consists of the first 100 products after the top- m products. We set $m = 10$, which results in performing the simulation with each product ranked in the range [11, 111] as a target product. The reason for this is that we want to measure how the algorithms perform for many different target products. Next, the ranked search results are obtained and a faceted faceted search session is simulated, where a user is aware of the target product, but is only able to recognize it when it appears in the top-10 results. The user keeps clicking on a facet (described shortly) until either the target product ap-

pears in the top-10, or the target product disappears from the result set. The latter can happen if the user simulation strategy does not assume that the user has perfect knowledge of the target product and therefore sometimes clicks on a facet that does not belong to the target product. In each simulation of a faceted search session, we record how many products the target product has been promoted over for each click. We repeat this process for $l \in \{3, 5, 7\}$, i.e., the number of facets to be displayed.

For the estimation of the target document identity probability $p(d = d_q)$, we employ a Zipfian distribution, as the authors of [9] have done. For $j \in \{11, \dots, 111\}$, we have $p(d_j = d_q) = \gamma \cdot j^{-2}$, where d_j is the product returned by the search engine at rank j for query q , and γ is a normalization constant such that $\sum_{d_j} p(d_j = d_q) = 1$.

Queries. For the queries, we used the product attributes to manually create a list of terms of all sensible qualitative values, such as ‘iOS’, ‘autofocus’, ‘flash’, and ‘led’. Using this terms list, we created 1000 queries that each consisted of three terms. We ensured that no queries were generated that had an empty result set. In order to avoid trivial drill down, we chose to filter out queries that have a result set with a size of less than 20. In our experiments, we use disjunctive semantics for the queries in which at least one of the terms needs to be present in a product for it to appear in the search results. Products that match on more terms are ranked higher.

User Simulation Strategy. For the user simulation strategies, we adopt the three approaches used in [9]. The first simulation strategy that we consider is the *Conjunctive User Strategy*. This strategy assumes that the user has perfect knowledge about the target product and that it selects *all* facets in one drill down. In a user interface, this is usually achieved by allowing the user to select multiple facets using a check box. The *Best Facet User Strategy* is the second simulation strategy that we consider. It also assumes that the user has perfect knowledge, however, instead of choosing all facets that correspond to the target product, it selects the single facet that promotes the target product the most. The third simulation strategy that we consider is the *Probabilistic User Strategy*. This strategy differs from the other two because it does not assume that the user has perfect knowledge. It involves a probabilistic process where the user scans the facets in the order that they are presented and the choice whether to click or not is modeled using two Bernoulli distributions, depending on whether the facet is actually associated to the target product or not. For facets that are associated to the target product we set $p = 0.9$ and for facets that are not associated to the target product we set $p = 0.1$. This means that a user might select a facet that does not belong to the target product with 10% probability.

Implementation. We provide a faceted search engine implementation (<http://faccy.net>) that supports the same query semantics we used in the experiments, as well as all the algorithms that we have evaluated in this paper. The user can enter a query, separated by comma’s to perform a disjunctive query, after which the facets will be shown. By default, the method ‘All Facets’ is used, which simply shows all possible facets. The user can click at any point in time on a different facet selection algorithm, shown at the top of the page. Both the experiments and the Web application have been implemented in Python, using MongoDB as the database. The Web application is implemented in

Django 1.5 and the experiments have been run on the LISA cluster, consisting of several hundreds multi-core nodes, provided by SARA Computing and Networking Services.

4.1 Results

After performing the experiments, we obtain a data set where we have determined, for each user simulation strategy (3), for each facet selection algorithm (9), for each query (1000), for each considered target product (100), for each click (1 to maximum 10), the *utility*. In this context, the utility is computed by counting the number of products a target product was promoted over in a single click. We perform the evaluation separately for each user simulation strategy and we use three measures to perform the analysis.

First, we consider the average number of clicks, representing after how many clicks, on average, the faceted search session ends. Second, the average *total* utility is considered, i.e., the total number of products that a target product has been promoted over in a search session. Third, for each facet selection algorithm, we measure how many times the target product appeared in the top-10 after 1 click. This is a measure that shows us how good a facet selection algorithm is at ‘getting it immediately right’.

We compare the means of two facet selection algorithms using a statistical test. For all three measures that we analyze we have used a paired two-sample t-test in order to determine whether a difference with respect to a measure is significant or not. We use a significance level of 5%, but because we perform many comparison, we apply a Bonferroni correction to the significance level. In total, we have 972 comparisons, which means that the corrected significance level becomes $0.05/972 = 5.144033 \times 10^{-05}$ ($972 = 36$ pairs for a strategy \times 3 strategies \times 3 values for $l \times 3$ measures). We have found almost all of the comparisons to be significant. For this reason, we only highlight the ones that are not significant. Furthermore, due to space limitations, we will discuss and conclude our findings without presenting all the data obtained from the experiments. However, we would like to point out that for the presented conclusions, we have carefully analyzed all of this data.

Average number of clicks. For this measure, we found that the Greedy Count facet selection algorithm scores the worst (i.e., a high number of clicks) for all user simulation strategies and numbers of facets displayed. We can conclude that this algorithm is not effective in reducing the number of drill downs. Because all the presented facets are highly popular, even for the Probabilistic Strategy, where there is the possibility that a search session ends because the user clicks on the wrong facet, the Greedy Count algorithm has an average close to 10 (the maximum number of clicks). The Probabilistic Conjunctive and Probabilistic Entropy algorithms gave the best performance on this measure, with less than 3 clicks on average for all three strategies.

We also found that with the Conjunctive Strategy, the average number of clicks decreases with the number of facets displayed for all algorithms. This is likely because the drill down occurs more efficiently when there are more facets available to select from. Because the Conjunctive Strategy simulates a user that selects all facets associated with the target product, the average promotion of the target product increases (which decreases the number of clicks).

Table 1 shows the comparisons for which we found no evidence for a significant difference, i.e., these were not statisti-

cally significant when performing a paired t-test. For example, for $l = 3$ and for the Conjunctive Strategy, the average number of clicks for the Weighted Residuals algorithm and the Random algorithm are not statistically different from each other (second row).

Table 1: Comparisons for the ‘average number of clicks’ measure that are not statistically significant.

| l | Algorithm 1 | Algorithm 2 | Strategy |
|-----|---------------|-----------------|------------|
| 3 | Best Facet I | Conj. | Best Facet |
| 3 | Random | Weighted Resid. | Conj. |
| 3 | Best Facet I | Conj. | Prob. |
| 3 | Best Facet I | Prob. Conj. | Prob. |
| 5 | Best Facet II | Random | Best Facet |
| 7 | Prob. Conj. | Prob. Entropy | Best Facet |
| 7 | Best Facet I | Prob. Conj. | Prob. |
| 7 | Best Facet II | Conj. | Prob. |

Average utility. When we analyzed the results for this measure, we did not notice any significant patterns across the number of facets to be displayed. We did notice, however, that the Random algorithm gives the same performance as the Best Facet II algorithm for $l = 3$. From this we can conclude that methods that rely on a useful ranking (such as Best Facet II) are not suited when little facets need to be shown to the user.

Second, we find that the Probabilistic Entropy and Conjunctive algorithms give the best performance, with an utility score between 50 and 60 (except for the Probabilistic Strategy, with a score between 25 and 30). We also noticed that the use of the Entropy approach significantly improves the Conjunctive algorithm, as we found that the Probabilistic Entropy always outperforms the Conjunctive algorithm. As for the significance tests, we found only two comparisons not significant: (1) the Best Facet II and the Random algorithm (with the Best Facet User strategy), and (2) the Conjunctive and the Probabilistic Entropy algorithm (with the Probabilistic User strategy).

Top-10 Promotion Percentage. The last measure that we use to analyze the results is the the percentage of target products that were promoted into the top-10 products after one click. With respect to the Probabilistic Conjunctive and Probabilistic Entropy algorithms, we find that they score the highest, with the Probabilistic Entropy algorithm being the winner. This shows that the Conjunctive algorithm, which was shown to be superior in the work of [9], can be improved by introducing a hybrid approach that allows to escape from local minima. However, the added value of the Probabilistic Conjunctive and Probabilistic Entropy algorithms seems to decrease with the number of facets to be shown. This suggests that hybrid approaches, such as the Probabilistic Conjunctive and Probabilistic Entropy algorithms, are most suited when not many facets need to be displayed.

Table 2 shows the comparisons that are not significant for this measure. We can notice that many comparisons result in a non-significant difference. We should point out, however, that most of these insignificant differences apply only to the Probabilistic Strategy, as there are only two comparisons that are not significant for the other strategies. From this, we can conclude that there is no clear winner for this measure with the Probabilistic Strategy.

Table 2: Comparisons for the ‘top-10 promotion’ measure that are not statistically significant.

| l | Algorithm 1 | Algorithm 2 | Strategy |
|-----|---------------|-----------------|------------|
| 3 | Best Facet I | Conj. | Prob. |
| 3 | Best Facet I | Prob. Conj. | Prob. |
| 3 | Best Facet I | Prob. Entropy | Prob. |
| 3 | Best Facet I | Random | Prob. |
| 3 | Conj. | Random | Prob. |
| 3 | Prob. Conj. | Prob. Entropy | Prob. |
| 3 | Prob. Conj. | Random | Prob. |
| 3 | Prob. Entropy | Random | Prob. |
| 5 | Entropy | Weighted Resid. | Best Facet |
| 5 | Best Facet I | Random | Conj. |
| 5 | Best Facet I | Prob. Conj. | Prob. |
| 7 | Best Facet I | Best Facet II | Prob. |
| 7 | Best Facet I | Prob. Conj. | Prob. |
| 7 | Conj. | Prob. Entropy | Prob. |
| 7 | Prob. Conj. | Prob. Entropy | Prob. |

5. CONCLUSIONS

In this paper, we focused on automatic facet selection in the domain of e-commerce, for the purpose of minimizing the number of steps required by the user in order to find its desired product. We proposed several facet selection algorithms, which we evaluated against the state-of-the-art algorithms from literature. Furthermore, we implemented all considered facet selection algorithms in a freely available Web application called *faccy.net*.

The evaluation was performed with simulations employing 1000 queries, 980 products, 487 facets, and three drill down strategies. We used three different evaluation metrics. First, we consider the average number of clicks, i.e., after how many clicks, on average, the faceted search session ends. We can conclude that the Greedy Count algorithm performs the worst for this metric as it has a relatively high average number of clicks, while the Probabilistic Conjunctive and Probabilistic Entropy algorithms are performing the best for all three user simulation strategies. Second, we consider the average *total* utility, i.e., the total number of products that a target product has been promoted over in a search session. The main conclusion that we draw from this analysis is that the hybrid approaches that we propose (Probabilistic Conjunctive and Probabilistic Entropy) perform much better with respect to average utility. Third, for each facet selection algorithm, we measure how many times the target product appeared in the top-10 after one click. We observe that for the Probabilistic Strategy, there is no clear winner. For the other two simulation strategies, i.e., the Best Facet Strategy and the Conjunctive Strategy, we can conclude that both the Probabilistic Conjunctive and Probabilistic Entropy give good results, with the Probabilistic Entropy being the best of the two.

In future work, we would like to integrate the user selected facets in the optimization process. Such an approach might be able to learn from the clicking behavior of a user and improve the overall performance. Furthermore, we plan to evaluate our proposed approaches in a user-based study, where we can measure other performance aspects, such as the total time of a drill down process.

Acknowledgment

Damir Vandic is supported by an NWO Mosaic scholarship for project 017.007.142: Semantic Web Enhanced Product Search (SWEPS). The experiments have been conducted on the LISA Computing Cluster at SARA Amsterdam.

6. REFERENCES

- [1] O. Ben-Yitzhak, N. Golbandi, et al. Beyond basic faceted search. In *Proceedings of the International Conference on Web Search and Web Data Mining (WSDM 2008)*, pages 33–44. ACM, 2008.
- [2] P. Brusilovsky. Adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 11(1):87–110, 2001.
- [3] B. J. Corbitt, T. Thanasankit, and H. Yi. Trust and e-commerce: a study of consumer perceptions. *Electronic Commerce Research and Applications*, 2(3):203–215, 2003.
- [4] J. English, M. Hearst, R. Sinha, K. Swearingen, and K. Yee. Hierarchical faceted metadata in site search interfaces. In *Extended Abstracts on Human Factors in Computing Systems (CHI 2002)*, pages 628–639. ACM, 2002.
- [5] M. Hearst. Design Recommendations for Hierarchical Faceted Search Interfaces. In *ACM SIGIR Workshop on Faceted Search*, pages 1–5, 2006.
- [6] S. Johnson Lim, Y. Liu, and W. Lee. Multi-facet product information search and retrieval using semantically annotated product family ontology. *Information Processing & Management*, 46(4):479–493, 2010.
- [7] J. Koren, Y. Zhang, and X. Liu. Personalized interactive faceted search. In *Proceedings of the 17th international conference on World Wide Web (WWW 2008)*, pages 477–486. ACM, 2008.
- [8] B. Kules, R. Capra, M. Banta, and T. Sierra. What do exploratory searchers look at in a faceted search interface? In *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2009)*, pages 313–322. ACM, 2009.
- [9] S. Liberman and R. Lempel. Approximately optimal facet selection. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing (SAC 2012)*, pages 702–708. ACM, 2012.
- [10] S. Pandit and C. Olston. Navigation-aided retrieval. In *Proceedings of the 16th International Conference on World Wide Web (WWW 2007)*, pages 391–400. ACM, 2007.
- [11] G. Sacco and Y. Tzitzikas. *Dynamic Taxonomies and Faceted Search*, volume 25. Springer, 2009.
- [12] C. Shannon, W. Weaver, R. Blahut, and B. Hajek. *The Mathematical Theory of Communication*, volume 117. University of Illinois press Urbana, 1949.
- [13] D. Tunkelang. Faceted search. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 1(1):1–80, 2009.
- [14] M. Wilson, P. André, et al. Backward highlighting: Enhancing faceted search. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST 2008)*, pages 235–238. ACM, 2008.