Adaptive Faceted Search for Product Comparison on the Web of Data

Alex Stolz and Martin Hepp

Universitaet der Bundeswehr Munich, D-85579 Neubiberg, Germany
{alex.stolz,martin.hepp}@ebusiness-unibw.org

Abstract. In this paper, we study the appropriateness of adaptive faceted search as a search paradigm for e-commerce on the Web of Data. We provide preliminary evidence that the product space in a sample dataset narrows down logarithmically by the number of product features used in a query, and show that the usability of an adaptive, instance-driven faceted search interface is comparable to approaches with hard-wired product features, while improving the depth of product search and comparison.

Keywords: E-Commerce, Product Comparison, Faceted Search, Usability, SUS, HCI, Linked Data, Semantic Web, RDF, SPARQL

1 Introduction

In the recent years, companies have started to add structured e-commerce data published as RDFa and Microdata markup to HTML Web pages. Such product, store, and offer data, while mainly provided for major search engines like Google and primarily based on the GoodRelations and schema.org vocabularies, forms a promising data source for novel Web applications and services.

Unfortunately, the available means for exploring this giant RDF graph of e-commerce information are limited. The diversity of products and data sources, the inherent learning effects during search, the heterogeneity in terms of data semantics with the resulting need to align data schema elements on the go, and the sparsity of the graph of product information, create special requirements for product comparison solutions that are currently not met. On top of the technical challenges, products and services are typically characterized by a vast variety of product features that influence the overall utility of a certain product, trade-offs between such features, and a significant variation in item prices. Consequently, product comparison includes multi-dimensional, non-linear decisions.

Conventional search approaches fall short with structured product data at Web scale. Information retrieval, e.g. keyword search, essentially flattens multi-dimensional product descriptions to simple, one-dimensional term matches. On the other extreme, query formulation as with SPARQL is generally very complex and lacks mediation between the conceptual models of the data vs. the mental models of human users. Other methods suggested for browsing RDF data (e.g.

Tabulator [6]) are very low-level for serious product search. As a result of these shortcomings, consumers tend to narrow down the set of candidate offers very early in the search process, which bears the risk that potentially interesting product offers are eliminated prematurely. Also, results are highly biased towards a single product or offer dimension (e.g. low prices) [15].

In this paper, we show that *faceted search* ([19]; cf. [16]), a special form of exploratory search [11], is appropriate for product comparison on the Web of Data. Faceted search is well established both in practice (e.g. eBay[1] and Amazon[2]) and in academia as a way to guide users through option spaces (e.g. [23,8,13]). In a nutshell, it constitutes a multi-dimensional interaction paradigm based on facet-value pairs, e.g. product dimensions, that dynamically adapt with the actual data.

2 Requirements for Faceted Search

This section defines important requirements for product search that can to a large extent be readily met by faceted search interfaces over RDF data.

- *Regard multi-dimensionality of products:* The complexity and dynamics of products and services necessitate multi-parametric searches based on distinguishing properties and attributes of product entities, which, on the Web of Data, can be realized by considering the structure of the available data.
- *Support learning about the option space:* Search is an iterative, incremental learning process (e.g. [12, p. 9]) rather than a static, one-shot query. For example, users grasp new information about the option space in every search turn [5], possibly leading to changes in price expectation. Thus, users need a way to relax or refine their constraints and preferences based on how those modify the size of the option space.
- *Facilitate incremental, user-driven schema alignment:* For product search with incremental learning, it is not only vital to assist in navigating and pruning the option space, but also to actively engage the user in the search process. Since users are likely to learn about correspondences in the underlying product features during the user interaction, the approximate alignment of conceptual elements should be integrated in the iterative search process, and be fed back to the graph. E.g., a user interface could ask the user for approval of a possible match between two product features. In an RDF environment, corresponding axioms can be easily added to the existing data as named RDF graphs – potentially managed on a per-user basis.
- *Take into account the popularity of conceptual elements in the instance data:* A user interface that is solely based on the schema elements defined in the underlying ontologies is inefficient, because the user lacks information about the availability of matching data (e.g. whether a property is used at all) and the relevance of a constraint on the option space (e.g. whether products differ

[1] `http://www.ebay.com/`

[2] `http://www.amazon.com/`

in that property). Due to a sparsely populated graph of product information on the Web, efficient user interfaces should thus adapt to the actual usage of schema elements in the data rather than be based on schema definitions.
– *Utilize metrics for the efficiency of the search process:* An efficient search interface presents choices to the user that help to quickly narrow down the option space, e.g. by proposing discerning features that partition the option space in the best possible way, or by suggesting properties that promise the highest utility to a given user need. The user dialog in faceted search is fundamentally a decision tree problem, where the user interaction steps are branches of the tree. Because the facets are orthogonal to each other, the decision tree can be constructed in any order [13]. However, if we want to optimize the search efficiency for the user, we have to create and, if necessary, update the resulting tree based on a "best split" strategy known from decision tree research in data mining [18, p. 158]. Popular algorithms from literature, e.g. ID3 [14], iteratively choose attributes maximizing the information gain. In this context, [10] mention some popular facet-pair suggestion strategies, namely relying on frequency, probability, and the information gain. The authors in [22] further give an overview over different metrics appropriate for product search to help decide which facets to present to the user.

3 Experiments

This paper investigates the appropriateness of faceted search interfaces for the Web of Data. To test for two fundamental aspects of search interfaces, namely search efficiency and usability, we first measure the impact of specificity in product search on the size of the result set using a simulation of random walks. Then, we conduct a usability study where we contrast a data-driven, adaptive faceted search interface with a second alternative with hard-wired product features.

3.1 Impact of Search Specificity on the Size of the Result Set

We simulated a number of product searches to find out how dispersed the search space for products is and how well a faceted search approach on average performs regarding partitioning the option space.

Method. We took a random sample of 875 automobile offers[3] from the *mobile.de* car listing Web site. We extracted the product features from the respective Web pages and populated an RDF graph via mapping product features to properties from the VSO ontology[4]. For the sake of simplicity, we did not take into account quantitative values for our simulation, but only qualitative and datatype properties. The variety of qualitative and datatype properties over the whole dataset

[3] More precisely, we took random result page numbers between 1 and 100 for random price ranges between 1 and 100,000 Euros.
[4] `http://purl.org/vso/ns`

Table 1. Variety of properties and values in automobile dataset

Property	**Variety of Values**
http://purl.org/vso/ns#bodyStyle	6
http://purl.org/vso/ns#color	24
http://purl.org/vso/ns#condition	5
http://purl.org/vso/ns#feature	60
http://purl.org/vso/ns#fuelType	10
http://purl.org/vso/ns#meetsEmissionStandard	5
http://purl.org/vso/ns#transmission	3

is shown in Table 1. These numbers give a total of 113 possible property-value pairs. From this range of possible property-value combinations, we drew one item at random and started from there 100 random walks with each simulating ten consecutive selection steps. After every selection step, we randomly picked a property-value pair from the reduced option space, which we obtained by issuing a proper SPARQL query.

Results. Figure 1 outlines the results of our simulation. At the beginning (step 0), the option space always entails the full range of 875 car offers. In search step 1, the median of the 100 iterations already goes down to circa 150 results, i.e. in 50% of the cases the first filtering step sorts out an average of more than 700 out of 875 automobiles. After having selected three product features, the median of the option space decreases to only three items.

As a possible constraint, our random walk does not include UNION clauses, i.e. the disjunctive selection of multiple facet values which would expand the option space (e.g. select a car that offers either manual or automatic transmission). However, we argue that this expansion operation does anyway occur rarely in practice when users seek interesting product offers.

Discussion. We can see clearly from the analysis that the space of possibly matching products decreases logarithmically with the number of features specified in a query. This confirms our assumption that learning about the option space, i.e. how relaxing and refining requirements and preferences based on the set of remaining choices, is a critical part of product search interaction. It also highlights that in specific branches of product search and thus sparsely populated decision trees, a search interface can benefit from being dynamically generated directly from the data about products and their characteristics.

Of course, the findings presented are currently based on a single sample data set of 875 cars, albeit those have been selected randomly from a very significant real dataset from a car sales portal. The effect of the number of features might be less significant if we took into account the correlation of features (e.g. that a stronger engine is likely to be found in combination with more seating capacity), which we deliberately abstracted from by selecting the features randomly. We would counter, however, that exactly these correlations between product features

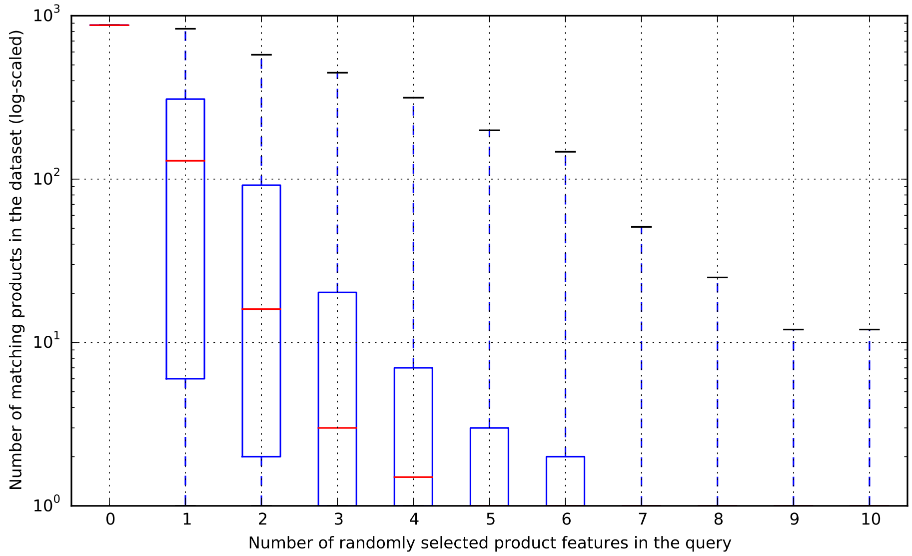

Fig. 1. Random walk simulation over a decision tree for 875 automobile offers

are unknown ex ante to a person exploring a product space and thus stress the importance of the learning effect of iterative product search.

3.2 Usability Studies of Faceted Search Interfaces for Products

Faceted search interfaces have recently attracted significant research interest. Various demonstrators, user studies, and evaluations repeatedly attest them superior usability in contrast with other search paradigms (e.g. [24,10,13,8]). In a survey in [23], the authors systematically compare faceted search with other popular search paradigms.

In here, we conduct a user study in order to find out whether an instance-driven search interface has a negative impact on usability, because hard-wired, consolidated user interfaces found in today's commercial faceted search applications have the advantage that the displayed facets can be based on popular mental models of human users. Instance-driven, adaptive faceted search interfaces bear the risk of being confusing to users, as the facets and facet names presented to users may change dynamically depending on the available data.

Method. In order to evaluate a potentially negative effect, we first developed an instance-driven, adaptive faceted search interface[5] for product comparison on the Web of Data, that addresses the requirements outlined in Sect. 2. Then, we prepared an identical search interface except for relying on hard-wired product

[5] `http://www.ebusiness-unibw.org/tools/product-search/`

Table 2. Results of SUS experiments

	Students		**Crowdsourcing**	
	A	B	A	B
No. participants	39	29	50	50
No. incorrect answers	5	3	13	9
No. answers considered	39	29	37	41
Avg. SUS score	66.54	72.59	65.00	68.75

features[6]. As the data to present in our two search interfaces, we used a random subset of 25 car offers out of the random sample of 875 car offers from *mobile.de*.

We set up a usability study according to the System Usability Scale (SUS) [7] score. The questionnaire encompasses ten brief questions where each response is represented by a five-point Likert scale ranging from *strongly agree* to *strongly disagree*. SUS questions are designed to alternate between positive and negative statements. In addition, we included a gold question to filter out unreliable candidates based on an incorrect response. We placed the gold question at the end of the questionnaire. Otherwise, we feared that participants would possibly give up too early, because it required a bit of effort to look at the information displayed in the search interface. Finally, we asked for optional feedback, which we used in a later analysis for interpreting the results. We put the questionnaire online so that users could test the search interface and answer to questions remotely.

We conducted two separate usability studies. The first one we ran with undergraduate students from our University, who specialize in business management or related fields. They were asked to assess the usability of the original, dynamic search interface A and, later, to repeat the same task with the amended search interface B. Our second experiment was harnessing crowd workforce from the CrowdFlower platform. As compared to the students experiment, we ran the usability test for both search interfaces A and B in parallel with two distinct groups of participants.

Results. In the following, we report on the empirical results obtained from the two usability studies, as summarized in Table 2.

Usability Experiment with Students. The task completion rate (cf. [17]) for students was $34/39 = 87\%$ for search interface A, and $26/29 = 90\%$ for search interface B. For students' ratings, we decided against eliminating incorrect answers to the gold question, because a closer investigation of individual responses revealed that students were not fooled by the alternating pattern of SUS questions rotating between positive and negative statements.

Search interface A achieved an average SUS score of 66.54, which is slightly below the average of 68[7], which was the mean SUS score among 500 system

[6] `http://www.ebusiness-unibw.org/tools/product-search-static/`

[7] `http://www.measuringu.com/sus.php`

usability studies. Taking on the qualitative, "adjective" rating introduced in [4], the search interface is considered *good* (SUS score close to 71.4). By comparison, search interface B obtained an average SUS score of 72.59. We stated the following null hypothesis to test the difference in the usability scores for significance:

Null Hypothesis. *There is no difference among SUS scores for search interfaces A and B obtained by two student samples from the same population.*

A Shapiro-Wilk test revealed that we cannot assume that both SUS score samples are normally distributed (p-values of 0.03 and 0.06), thus we compared the two samples using a non-parametric statistical test, the Wilcoxon rank-sum test.

The average usability scores assigned by our students to search interface A ($median = 70.00$) did not differ significantly from usability scores assigned to search interface B ($median = 75.00$), $W = -1.45$, $p = 0.15$, $r = -0.18$.

Usability Experiment with Crowdsourcing. Unlike in the previous experiment, we did only accept contributions by crowd workers who correctly answered the gold question. The task completion rate for crowd workers was $37/50 = 74\%$ for search interface A, and $41/50 = 82\%$ for search interface B.

Search interface A achieved an average SUS score of 65.00, which is below 68, but still *good* according to [4]. Search interface B obtained an average SUS score of 68.75. The null hypothesis below was used to test whether the usability scores significantly differ:

Null Hypothesis. *There is no difference among SUS scores for search interfaces A and B obtained through two different samples of crowd workers.*

A Shapiro-Wilk test revealed that we cannot assume that both SUS score samples are normally distributed (p-values of 0.13 and 0.01), thus we compared the two samples using a non-parametric statistical test, the Wilcoxon rank-sum test.

The average usability scores assigned by the first group of crowd workers to search interface A ($median = 65.00$) did not differ significantly from usability scores assigned to search interface B by the second group of crowd workers ($median = 73.75$), $W = -1.30$, $p = 0.19$, $r = -0.15$.

Discussion. This analysis shows that, in principle, a fully dynamic search interface directly based on product features found in the data, is not systematically less intuitive for users than one based on established, hard-wired product features used in existing car portals. However, we see a small negative effect in usability, which we expected, because the static, hard-wired set of search dimensions allows a higher degree of users' familiarity with the terminology and conceptual model of a search interface. We conclude from that small negative effect that a data-driven search interface for products comes at a cost, which must be compensated for by additional gains in precision, recall, and eventually the utility of the finally selected product.

We would also like to stress that a usability-based evaluation of novel search interfaces has a systematic weakness, because it only analyzes how well a user can handle the interface, but not the quality of the choices eventually made (e.g. how well the finally selected product meets the user's needs). As we have shown in the first part of this section, the sparsity and heterogeneity of the product space indicates that a more precise navigation in the option space can return much better product matches.

4 Related Work

Within the frame of this work, we deem mostly relevant three research directions, namely (1) adaptive faceted search interfaces, (2) faceted search over RDF data, and (3) faceted product search on the Semantic Web.

4.1 Adaptive Faceted Search

In adaptive faceted search interfaces, user controls dynamically adapt to the actual data restricted by the current selection. An adaptive faceted search interface was proposed in [1] to investigate content within Twitter streams. Facets and facet values are computed based on semantic enrichment of Twitter messages. The search interface adapts according to frequency, user profile, temporal context, and diversification. In [20], the author aims to facilitate information access on the Web via an adaptive, exploratory search relying on multiple search paradigms. Another work related to personalized faceted search over Web document metadata was proposed in [10], where facet views adapt according to user ratings.

4.2 Faceted Search over RDF Data

As an easy-to-use alternative for SPARQL querying, faceted search gained wide attraction as a search paradigm for RDF data. Faceted search as a means to navigate over arbitrary datasets with structured data was formalized in [13]. A similar approach develops a formal model for question answering based on faceted queries and regards also ontological reasoning [2]. The work in [8] combines the ease-of-use of faceted search with the expressive power of the SPARQL query language. In comparison to the two other works that operate on set operations over resources, this approach provides navigation through query transformations at the syntactic level. Some large-scale faceted search interfaces over real RDF datasets were suggested in [9] and [3]. In [9], the authors built a faceted search interface over structured Wikipedia infobox data (DBPedia). The work in [3] studies limitations of conventional faceted search systems, and presents a faceted search interface over Yago.

4.3 Faceted Search over Structured E-Commerce Data

The work in [21] presents a faceted product search interface over structured e-commerce data from the Web. The data store[8] presently contains a selection of

[8] `http://xploreproducts.com/`

product offers along with review data from selected online stores. In comparison to our research that proposes fully data-driven product search, this work only supports basic commercial properties of product offers, and categorizes products into a rigid category structure.

5 Conclusions

In this paper, we have studied the appropriateness of data-driven, adaptive faceted search interfaces for navigating the sparse graph of Linked Open Data for e-commerce on the Web with explicit support for user learning about the option space. We have provided preliminary evidence that the selection steps in faceted search interfaces drill down the option space logarithmically, and have shown that the usability loss of a dynamic, instance-driven faceted search interface in comparison to an approach with hard-wired product features is insignificant.

The small-scale usability study in this paper also indicates that users apparently have gotten used to search interfaces that expose rigid navigation structures optimized for individual application domains. While viable in smaller and controlled settings, it is not feasible for e-commerce over Linked Open Data, where diverse and dynamic product domains need to be consolidated. A large-scale evaluation with real e-commerce data from the Web is planned for future work.

Acknowledgments. The authors would like to thank Florian Ott for kindly supporting us in setting up and hosting the necessary infrastructure of our search systems, and all the undergraduate students at Universitaet der Bundeswehr Munich who participated in the usability study.

References

1. Abel, F., Celik, I., Houben, G.J., Siehndel, P.: Leveraging the Semantics of Tweets for Adaptive Faceted Search on Twitter. In: Proceedings of the 10th International Semantic Web Conference (ISWC 2011). pp. 1–17. Springer Berlin Heidelberg, Bonn, Germany (2011)
2. Arenas, M., Cuenca Grau, B., Kharlamov, E., Marciuška, S., Zheleznyakov, D.: Faceted Search over Ontology-Enhanced RDF Data. In: Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM 2014). pp. 939–948. ACM, Shanghai, China (2014)
3. Arenas, M., Cuenca Grau, B., Kharlamov, E., Marciuška, S., Zheleznyakov, D.: Towards Semantic Faceted Search. In: Poster Proceedings of the 23rd International World Wide Web Conference (WWW 2014 Companion). pp. 219–220. ACM, Seoul, Korea (2014)
4. Bangor, A., Kortum, P., Miller, J.: Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. Journal of Usability Studies 4(3), 114–123 (2009)
5. Bates, M.J.: The Design of Browsing and Berrypicking Techniques for the Online Search Interface. Online Information Review 13(5), 407–424 (1989)

6. Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A., Sheets, D.: Tabulator: Exploring and Analyzing Linked Data on the Semantic Web. In: Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI 2006). Athens, GA, USA (2006)
7. Brooke, J.: SUS - A Quick and Dirty Usability Scale. In: Jordan, P., Thomas, B., Weerdmeester, B., McClelland, I. (eds.) Usability Evaluation in Industry, pp. 189–194. Taylor & Francis (1996)
8. Ferré, S., Hermann, A.: Semantic Search: Reconciling Expressive Querying and Exploratory Search. In: Proceedings of the 10th International Semantic Web Conference (ISWC 2011). pp. 177–192. Springer Berlin Heidelberg, Bonn, Germany (2011)
9. Hahn, R., Bizer, C., Sahnwaldt, C., Herta, C., Robinson, S., Bürgle, M., Düwiger, H., Scheel, U.: Faceted Wikipedia Search. In: Proceedings of the 13th International Conference on Business Information Systems (BIS 2010). pp. 1–11. Springer Berlin Heidelberg, Berlin, Germany (2010)
10. Koren, J., Zhang, Y., Liu, X.: Personalized Interactive Faceted Search. In: Proceedings of the 17th International World Wide Web Conference (WWW 2008). pp. 477–485. ACM, Beijing, China (2008)
11. Marchionini, G.: Exploratory Search: From Finding to Understanding. Communications of the ACM 49(4), 41–46 (2006)
12. Morville, P., Callender, J.: Search Patterns. O'Reilly Media (2010)
13. Oren, E., Delbru, R., Decker, S.: Extending Faceted Navigation for RDF Data. In: Proceedings of the 5th International Semantic Web Conference (ISWC 2006). pp. 559–572. Springer Berlin Heidelberg, Athens, GA, USA (2006)
14. Quinlan, J.R.: Induction of Decision Trees. Machine Learning 1(1), 81–106 (1986)
15. Sacco, G.M.: The Intelligent E-Store: Easy Interactive Product Selection and Comparison. In: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology (CEC 2005). pp. 240–248. IEEE, Munich, Germany (2005)
16. Sacco, G.M., Tzitzikas, Y.: Dynamic Taxonomies and Faceted Search: Theory, Practice, and Experience. Springer Berlin Heidelberg (2009)
17. Sauro, J., Lewis, J.R.: Estimating Completion Rates from Small Samples Using Binomial Confidence Intervals: Comparisons and Recommendations. Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting (HFES 2005) 49(24), 2100–2104 (2005)
18. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley, 1 edn. (2005)
19. Tunkelang, D.: Faceted Search. Morgan & Claypool (2009)
20. Tvarožek, M.: Exploratory Search in the Adaptive Social Semantic Web. Information Sciences and Technologies Bulletin of the ACM Slovakia 3(1), 42–51 (2011)
21. Vandic, D., van Dam, J.W., Frasincar, F.: Faceted Product Search Powered by the Semantic Web. Decision Support Systems 53(3), 425–437 (2012)
22. Vandic, D., Frasincar, F., Kaymak, U.: Facet Selection Algorithms for Web Product Search. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM 2013). pp. 2327–2332. ACM, San Francisco, CA, USA (2013)
23. Wei, B., Liu, J., Zheng, Q., Zhang, W., Fu, X., Feng, B.: A Survey of Faceted Search. Journal of Web Engineering 12(1), 41–64 (2013)
24. Yee, K.P., Searingen, K., Li, K., Hearst, M.A.: Faceted Metadata for Image Search and Browsing. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2003). pp. 401–408. ACM, Fort Lauderdale, Florida, USA (2003)