Automatic Ordering of Facets Based on Machine Learning

Diksha M
PG Scholar, Dept. Of CSE
CITech,B'lore

Prof.Raghavendra T.S
Dept. Of CSE,
CITech,B'lore

Abstract- Faceted browsing is widely used in Web shops and product comparison sites. In these cases, a fixed ordered list of facets is often employed. This approach suffers from two main issues. First, one needs to invest a significant amount of time to devise an effective list. Second, with a fixed list of facets it can happen that a facet becomes useless if all products that match the query are associated to that particular facet. In this work, we present a framework for dynamic facet ordering in e-commerce. Based on measures for specificity and dispersion of facet values, the fully automated algorithm ranks those properties and facets on top that lead to a quick drill-down for any possible target product. In contrast to existing solutions, the framework addresses e-commerce specific aspects, such as the possibility of multiple clicks, the grouping of facets by their corresponding properties, and the abundance of numeric facets. In a large-scale simulation and user study, our approach was, in general, favorably compared to a facet list created by domain experts, a greedy approach as baseline, and a state-of-the-art entropy-based solution.

Keywords— facet ordering, product search, user interfaces.

1.INTRODUCTION

Studies from the past have shown that other factors than the price play a role when a consumer decides to choose where to buy a product online . Therefore, online retailers pay special attention to the usability and efficiency of their Web shop user interfaces. Nowadays, many Web shops make use of the so-called faceted navigation user interface , which is in literature also sometimes referred to as 'faceted search'. Facets are used by some users as a search tool, while others use it as a navigation and/or browsing tool . One of the reasons why faceted search is popular among Web shops is that users find it intuitive. The term 'facet' has a rather ambiguous interpretation, as there are different types of facets. In this work, I refer to facets as the combination of a property and its value.Figure1 shows an example of a faceted search user interface.

Currently, most commercial applications that use faceted search have a manual, 'expert-based' selection procedure for facets , or a relatively static facet list . However, selecting and ordering facets manually requires a significant amount of manual effort. Furthermore, faceted search allows for interactive query refinement, in which the importance of specific facets and properties may change during the search session.

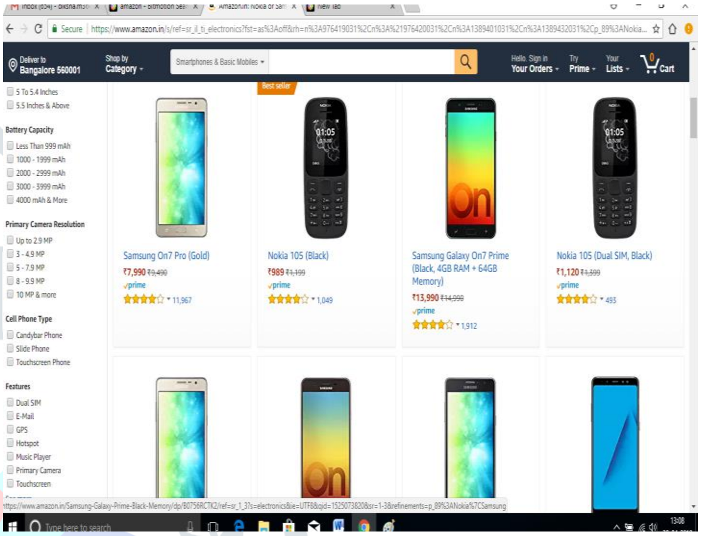

Figure 1: **A Screen shot of Amazon.com, showing a typical faceted search user interface in E-commerce.**

Therefore, it is likely that a predefined list of facets might not be optimal in terms of the number of clicks needed to find the desired product.

In order to deal with this problem, I propose an approach for dynamic facet ordering in the e-commerce domain. The focus of our approach is to handle domains with sufficient amount of complexity in terms of product attributes and values. Consumer electronics (in this work 'mobile phones') is one good example of such a domain. As part of our solution, we devise an algorithm that ranks properties by their importance and also sorts the values within each property. For property ordering, we identify specific properties whose facets match many products (i.e., with a high impurity). The proposed approach is based on a facet impurity measure, regarding qualitative facets in a similar way as classes, and on a measure of dispersion for numeric facets. The property values are ordered descending on the number of corresponding products. Furthermore, a weighting scheme is introduced in order to favour facets that match many products over the ones that match only a few products, taking into account the importance of facets. Similar to existing recommender system approaches , our solution aims to learn the user interests based on the user interaction with the search engine.

From the perspective of user interface design, we distinguish between two main facet types: qualitative facets (e.g., WiFi:true) and numeric facets (e.g., Lowest price (e):64.00). We further distinguish between two types of qualitative facets: nominal facets and Boolean facets. Nominal facets are, for example, those for the property Display Type, and can have any nominal value. Boolean facets are for instance Multitouch, and have only three options from an interface perspective: true, false, or no preference.

2.PROPOSED SYSTEM

We propose an approach for dynamic aspect requesting in the internet business space. The focal point of our approach is to deal with areas with adequate measure of many-sided quality as far as item traits and qualities. Customer hardware (in this work 'cell phones') is one great case of such an area. As a feature of our answer, we devise a calculation that positions properties by their significance and furthermore sorts the qualities inside every property. For property requesting, we recognize particular properties whose aspects coordinate numerous items (i.e., with a high contamination). The proposed approach depends on an aspect pollution measure, with respect to subjective features likewise as classes, and on a measure of scattering for numeric features. The property estimations are requested dropping on the quantity of comparing items. Moreover, a weighting plan is acquainted all together with support aspects that match numerous items over the ones that match just a couple of items, considering the significance of features.

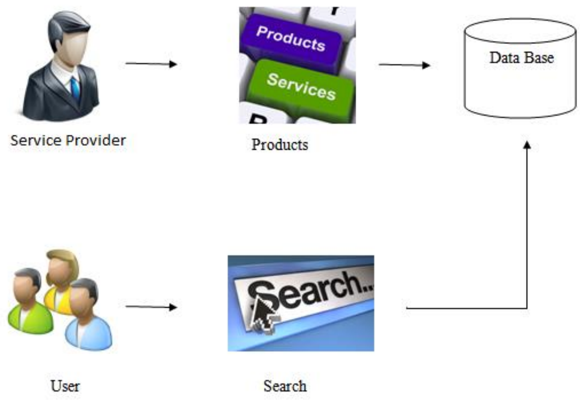

Figure 2: System Architecture

Our arrangement plans to take in the client intrigues in light of the client cooperation with the internet searcher.

2.1ADVANTAGES OF PROPOSED SYSTEM:

In our examination, we utilize the regular disjunctive semantics for qualities and conjunctive semantics for properties and consider the likelihood of penetrate ups. This implies result set sizes are relied upon to both increment and decline amid the hunt session, either by deselecting an aspect or picking an expansion feature in a property

In terms of the quantity of snaps, our approach appears to beat alternate techniques, with the exception of on account of the Best Facet Drill-Down Model, where each approach performs similarly well. Besides, for the Combined Drill-Down Model, our approach brings about the most minimal number of roll-ups and the most noteworthy level of fruitful sessions.

The moderately low computational time makes it reasonable for use in true Web shops, making our discoveries additionally pertinent to industry. These outcomes are likewise affirmed by a client based assessment think about that we moreover performed.

3. IMPLEMENTATION

3.1 SEARCH SESSIONS

A query in a search session is defined as a collection of previously selected facets. We have decided to apply disjunctive semantics to a selection of facets within a property. For facets across different properties, we use a conjunctive semantics. For example, selecting the facets Brand:Samsung, Brand:Apple, and Color:Black results in (Brand:Samsung OR Brand:Apple) AND Color:Black. Several ecommerce stores on the Web (e.g., Amazon.com and BestBuy.com) use the same principle, which, from a user experience point-of-view, is very intuitive. Our approach assumes that users can undertake two types of actions: drill-down and roll-up. A drilldown is defined as an action of selecting one or more facets, leading to a reduction of the result set size. A roll-up action increases the result set size, which is likely to happen when the user notices that the selected facets are too strict.

Figure 3: Initiates the two main approches (1)Computing the property scores. (2)Computing facet scores.

3.2 COMPUTING PROPERTY SCORES

In this module, now I discuss the details of computing property scores, shown as one of the first two processes. The outcome of the property scores is used to first sort the properties, after which the facet scores, are used to sort the values within each property. We zoom into the main steps of computing the property score. As shown by the diagram, the score for each property is computed separately and can thus be done in parallel. We designed the proposed algorithm in such a way that more specific facets and properties are ranked higher. To support the algorithm in identifying more specific facets, we introduce the disjoint facet count. This metric is used to compute the score for qualitative properties.

3.3 PRODUCT COUNT WEIGHTING

For numeric properties, we have chosen to use the knowledge about the distribution of the numeric values for computing property scores. It is fairly straightforward to imagine that it may be useful to drill-down using a numeric property when the values for the result set are widely dispersed. When the facets are nearly uniformly distributed over the complete range of values, a drill-down using a user-defined range would lead to a large reduction of the result set. With the Gini impurity and the Gini coefficient, we now have metrics to score both qualitative and numeric properties. This score is independent from the number of products on which it is based. This could possibly lead to problems, as properties that occur within few products will obtain a relatively high score. To compensate for this, we introduce the product count weighting. The product count weighting is used to normalize the Gini indices, resulting in the final property score.

3.4 COMPUTING FACET SCORES

In this module, we have explained how we compute scores for properties. We now discuss the details of computing facet scores, shown as one of the first two processes. However, our approach also sorts the values within each property in order to reduce the value scanning effort. This is in contrast to for instance the shown as one of the first two processes.

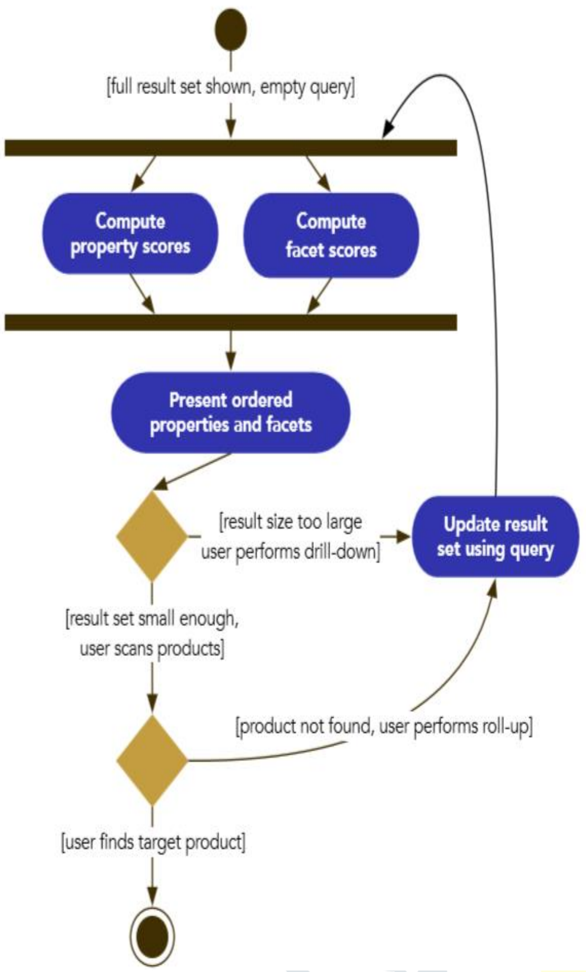

Figure 3 : **Activity diagram describing the main flow of a search engine**

However, our approach also sorts the values within each property in order to reduce the value scanning effort. This is in contrast to for instance the approach in exiting, which considers property ranking but disregards facets ranking. For numeric properties, value ordering is neglected, as these are often represented with a slider widget in user interfaces. The slider widgets, of which an example, give an indication of the minimum and maximum values for a property, and allow the user to freely define a range of facets within these boundaries. For qualitative properties our approach employs the facet count, ranking facets descending on count, per property. As the target product is unknown to the system, this will increase the chance that a facet matching the target product is placed on top.

4.RESULTS

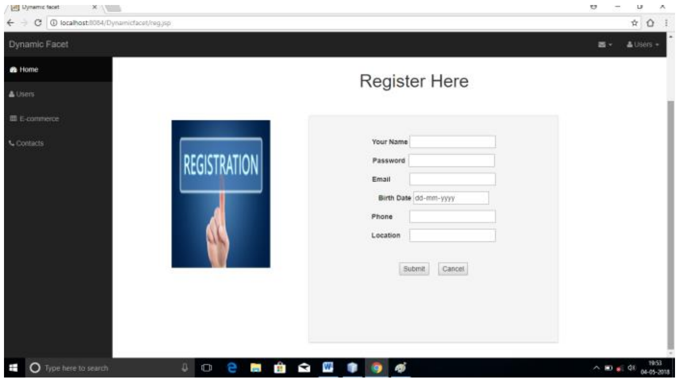

Figure 4: User Registration

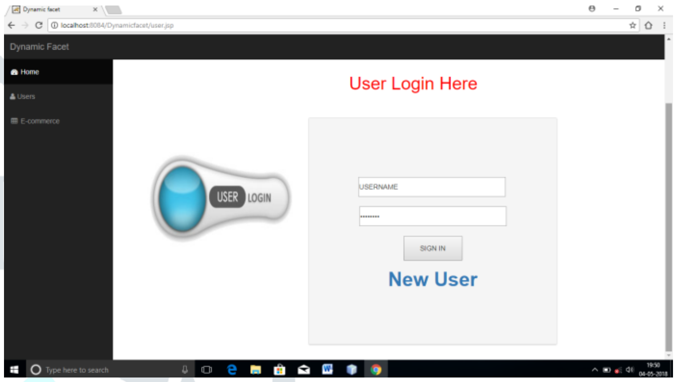

Figure 5: User login

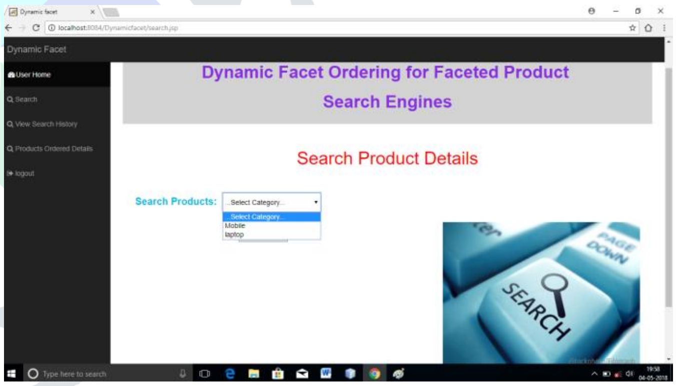

Figure 6:Search Product Details

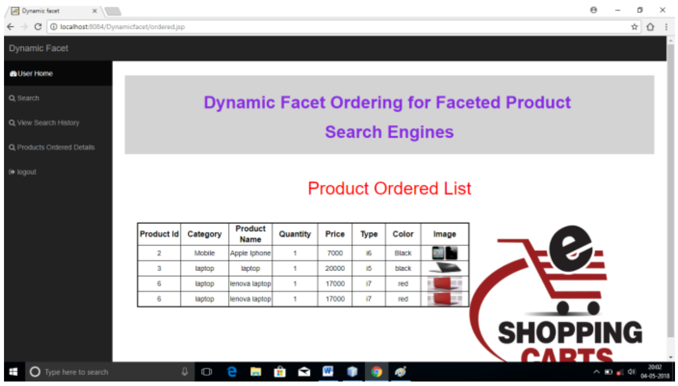

Figure 7: Product Ordered list

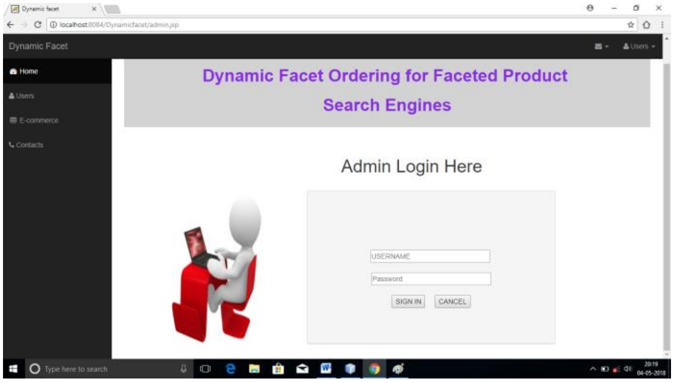

Figure 8: **Admin login**

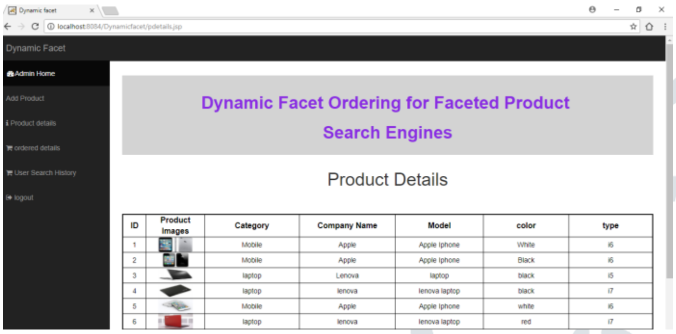

Fig 9: Admin Product Details

By drill down approach and ranking facets properties by their importance and sorting the values and attributes , the result set decreases and facets are dynamically arranged in such a way that any individual product could be found quickly and effectively .If the target product is found,the search session is completed and considered successful. If the target product is not found ,the user will perform roll up action which will increase the result set size ,and the same process repeats again.

5.CONCLUSION

In this work, proposed an approach that automatically orders facets such that the user finds its desired product with the least amount of effort. The main idea of our solution is and then, additionally, also sort the facets themselves. We used computing property scores and computing facet scores. For property ordering we want to rank properties descending on their impurity, promoting more selective facets that will lead to a quick drill-down of the results. Furthermore, we employ a product weighting scheme based on the number of matching products to adequately handle missing values and take into account the property product coverage. We evaluate our solution using an extensive set of simulation experiments, comparing it to three other approaches. While analyzing the user effort, especially in terms of the number of clicks, we can conclude that our approach gives a better performance than the benchmark methods and in some cases even beats the manually curated 'Expert-Based' approach. In addition, the relatively low computational time makes it suitable for use in real-world Web shops, making our findings also relevant to industry. These results are also confirmed by a user-based evaluation study that we additionally performed.

REFERENCE

[1] H. Zo and K. Ramamurthy, "Consumer Selection of E-Commerce Websites in a B2C Environment: A Discrete Decision Choice Model," IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 39, no. 4, pp. 819–839, 2009.

[2] M. Hearst, "Design Recommendations for Hierarchical Faceted Search Interfaces," in 29th Annual International Conference on Research & Development on Information Retrieval (ACM SIGIR 2006). ACM, 2006, pp. 1–5.

[3]D.Tunkelang,"FacetedSearch,"SynthesisLecturesonInformation Concepts, Retrieval, and Services, vol. 1, no. 1, pp. 1–80, 2009.

[4] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst, "Faceted Metadata for Image Search and Browsing," in Proceedings of the SIGCHI Conference on Human factors in Computing Systems. ACM, 2003, pp. 401–408.

[5] J. C. Fagan, "Usability Studies of Faceted Browsing: A Literature Review," Information Technology and Libraries, vol. 29, no. 2, p. 58, 2010.

[6] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen, and K.P. Yee, "Finding the Flow in Web Site Search," Communications of the ACM, vol. 45, no. 9, pp. 42–49, 2002.

[7] B. Kules, R. Capra, M. Banta, and T. Sierra, "What Do Exploratory Searchers Look at in a Faceted Search Interface?" in 9th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2009). ACM, 2009, pp. 313–322.

[8] Amazon.com, "Large US-based online retailer," http://www. amazon.com, 2014.

[9] V. Sinha and D. R. Karger, "Magnet: Supporting Navigation in Semi-structured Data Environments," in 24th ACM SIGMOD International Conference on Management of Data (SIGMOD 2005). ACM, 2005, pp. 97–106. [10] Kieskeurig.nl, "Major Dutch price comparison engine with detailed product descriptions," http://www.kieskeurig.nl, 2014.

[11] Tweakers.net, "Dutch IT-community with a dedicated price comparison department," http://www.tweakers.net, 2014.

[12] Q. Liu, E. Chen, H. Xiong, C. H. Ding, and J. Chen, "Enhancing Collaborative Filtering by User Interest Expansion via Personalized Ranking," IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 42, no. 1, pp. 218–233, 2012.

[13] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An Algorithmic Framework for Performing Collaborative Filtering," in 22nd Annual International Conference on Research and Development in Information Retrieval (ACM SIGIR 1999). ACM, 1999, pp. 230–237.

[14] J. Koren, Y. Zhang, and X. Liu, "Personalized Interactive Faceted Search," in 17th International Conference on World Wide Web (WWW 2008). ACM, 2008, pp. 477–486.

[15] G. M. Sacco and Y. Tzitzikas, Dynamic Taxonomies and Faceted Search. Springer, 2009, vol. 25.

[16] D. Dash, J. Rao, N. Megiddo, A. Ailamaki, and G. Lohman, "Dynamic Faceted Search for Discovery-Driven Analysis," in Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM 2008). ACM, 2008, pp. 3–12.

[17] S. Liberman and R. Lempel, "Approximately Optimal Facet Value Selection," Science of Computer Programming, vol. 94, pp. 18–31, 2014.

[18] D. Vandic, F. Frasincar, and U. Kaymak, "Facet Selection Algorithms for Web Product Search," in 22nd ACM International Conference on Information and Knowledge Management (CIKM 2013). ACM, 2013, pp. 2327–2332.

[19] H.-J. Kim, Y. Zhu, W. Kim, and T. Sun, "Dynamic Faceted Navigation in Decision Making using Semantic Web Technology," Decision Support Systems, vol. 61, pp. 59–68, 2014.

[20] Y. Zhu, D. Jeon, W. Kim, J. Hong, M. Lee, Z. Wen, and Y. Cai, "The Dynamic Generation of Refining Categories in OntologyBased Search," in Semantic Technology, ser. Lecture Notes in Computer Science, 2013, vol. 7774, pp. 146–158.