FacetX: Dynamic Facet Generation for Advanced Information Filtering of Search Results

Work-in-Progress Paper

Raffael Affolter
Institute of Applied Information Technology
Zurich University of Applied Sciences
Winterthur, Switzerland
affolraf@students.zhaw.ch

Andreas Weiler
Institute of Applied Information Technology
Zurich University of Applied Sciences
Winterthur, Switzerland
andreas.weiler@zhaw.ch

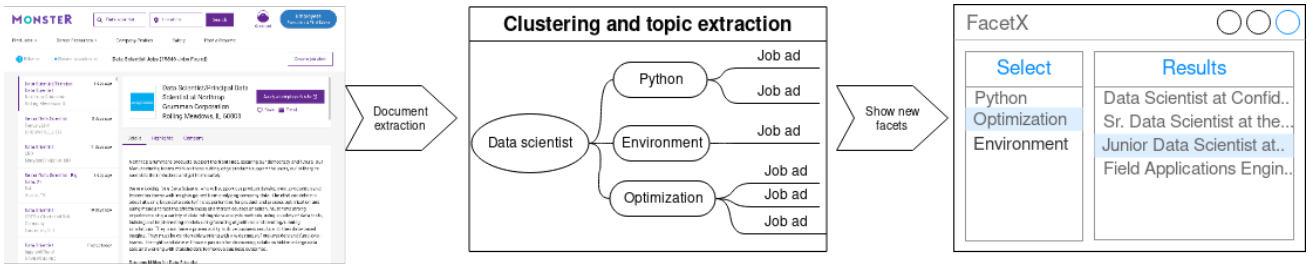

Figure 1: Exemplary demonstration of the FacetX application during a job search for "data scientist"

ABSTRACT

Searching for information is an important part of our daily life. People are searching for information about jobs, recipes, entertainment, places and much more. Many information systems try to support the users in finding the most relevant information to their information need by providing pre-built categories as filter mechanism. However, in most of the systems this support is designed in a very static way and does not consider the dynamic content of the documents in their collections. In this paper we introduce FacetX, an application for the dynamic generation of filter facets for advanced information filtering of search results.

1 INTRODUCTION AND MOTIVATION

The number of searches for information in the internet is steadily increasing. For example, Google [9] receives over 63,000 searches per second on any given day or people search for jobs in the network of Monster.com [7] about 8,000 times per minute. At the same time the number of results for the search queries increases as well. Most of the times the search query executor is overwhelmed by the large amount of results and the accompanying information overflow. For example, by searching on the platform of Monster.com for the job title "Data Scientist", the search result consists of more than 15,000 jobs. One solution for supporting the information seekers in finding their requested information in very large search results is to provide them navigational structures like product categories or price ranges in e-commerce platforms. With the support of the so-called faceted search [10] the information seekers are able to narrow down their search results to specific properties. However, the facets for filtering the search results are most of the times static and do not adapt themselves dynamically to the corresponding search results. For example, the filter facets for the search results on the platform of Monster.com for the job titles "Data Scientist" or "Gardener" are the same (*e.g.*, city or job status) even though the content of the search results is completely different.

Several systems for creating dynamic facets for supporting users in their search process have been proposed in the past. For example, Kim Hak-Jin *et al.* [3] use semantic web technologies to create facets from the ontologies of the data. After an initial search the system presents the resources and determined categories to the user. The user then selects a category and a value of it. The system then updates the result collection and presents the new determined categories and the corresponding resources. This process is repeated until the user finds the desired item. Similar to our approach is the guidance of the system through a graph which connects the search result together. Another example is proposed by Tvarozek *et al.* [11], which supports the user to overcome information overload by generating personalized dynamic facets for a user in multimedia collections. They present how a typical facet browser can be extended to support the generation of dynamic facets. With the use of a provided domain ontology and the analysis of the user behavior they try to generate the most relevant facets for the corresponding users. However, in contrast to our work, both systems are built on pre-defined ontologies, which are not necessary for FacetX.

In this paper we introduce FacetX, an application for the dynamic generation of filter facets for advanced information filtering of search results. FacetX can be applied to any domain, in which the results of a search are returned as a collection of documents. In contrast to previous work our application generates facets without the need of a predefined domain ontology and is therefore adaptable and usable in any context. In the following, we present the methodology behind our current work-in-progress implementation and several case studies, like searching for a job, recipe, or movie. These case studies show the effect of FacetX to search processes, which are daily undertaken by millions of users.

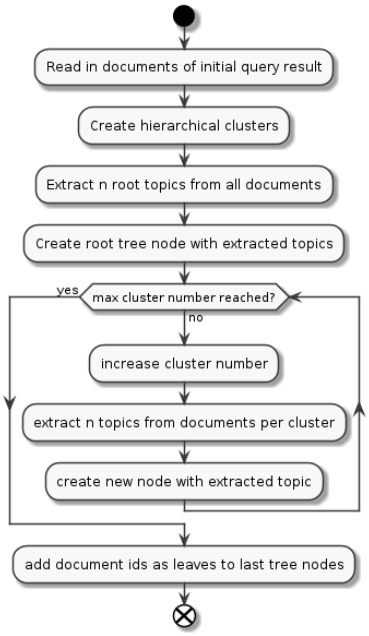

Figure 2: Process flow of the FacetX application.

2 METHODOLOGY

In this section, we describe the methodology behind the dynamic generation of the facets for the retrieved search results. Our approach follows a process flow with several successive phases, which can be seen in Figure 2. FacetX is implemented with the libraries of the Weka [4] toolkit and uses the StringToWordvector and the Rainbow stop word eliminator. The clustering is implemented with the use of the Weka hierarchical clusterer. To extract the topic, we used the parallel topic model of the Machine Learning for Language Toolkit (Mallet [6]). The input of the first phase are the results of a user defined search query which is executed by the information seeker. Therefore, we have a set of N documents per query as basis for the generation of the dynamic facets. To extract the features for the next step the content of the search result is tokenized, all stop words are removed from the documents and then term frequency-inverse document frequency(*tf-idf* [8]) is applied. In the next phase, we cluster the documents by applying agglomerative hierarchical clustering with complete-linkage as linkage criteria between the individual clusters. The result of this phase is a dendrogramm, which acts as a search tree for the creation of the facets for each branch. Since every branch in the dendrogram represents a cluster we can take every document per cluster to extract topics from it with the Latent Dirichlet Allocation (*LDA* [1]). The resulting number of clusters also decides how deep the search tree is. If the number of clusters is equal to the number of documents the leaves would contain only one document with the main topics in the parent node. In our application we decided to take the most important word for each topic as a facet suggestion.

3 CASE STUDIES

In the following, we describe three case studies, in which FacetX is able to support users in the filtering step of their daily life search processes. The first one is using FacetX to create facets for movies from the internet movie database (IMDB) [5]. In the second case study we create facets for a recipe search on the food platform epicurious [2] and in the third one we show the application of FacetX to a job search on Monster.com [7].

3.1 Movie Search

The internet movie database lists information about millions of movies and tv series on their platform. If a user is interested in movies of a specific genre like action or mystery the search is very simple and the results are presented as a sorted list (*e.g.* by user rating or year) to the user. The sorting and filter properties are always the same for all genres and other search results on the platform. However, we claim that users could definitively benefit from dynamically generated facets, which are for example based on the story line of the movies. For example, it would be possible to group together more similar movies like Star Wars episode 6 and 7, by extracting facets based on their synopsis. For this case study we extracted the 150 top ranked sci-fi movies sorted by user ratings from IMDB. The topic extraction was applied with the contents of the synopsis of the movies. We created 25 clusters with the best term of 6 topics as facets. As result we received some interesting topics (*cf.* Figure 4). For example, the topic around “astronauts” and “constructions” refers to the movies “Moon”, “The Martian” and “2001: A Space Odyssey” as seen in Figure 4a. Another interesting example is the topic consisting of “facehuggers”, “salvage”, and “romulans” which refers to the movies “Alien”, “Aliens”, and “Star Trek” as seen in Figure 4b.

3.2 Recipe Search

For this case study, we chose to search for recipes on epicurious by using the advanced search functionality of the platform. While epicurious offers a lot of facets to choose from, like technique or ingredient, a search still can result in a very high number of recipes. For example, the search for the technique barbecue returns a total of 1687 results and the search for the ingredient soup/stew returns a total of 1955 results. Although, it is possible to reduce the number of results further by selecting additional facets, mostly still a high number of recipes remain in the results.

To show the effectiveness of FacetX for the recipe search on epicurious we searched for recipes containing chicken as an ingredient and reduced the number of the first results with the additional facet “healthy” to the final number of 286 recipes. We then used the ingredients for clustering and topic extraction. With 50 clusters and 6 topics per node the results seem very interesting. The resulting topics (*cf.* Figure 5a) included some useful information to a dish like “boneless”, “skinless” or “skin-on”. Other topics were not as helpful like “oil”, “tablespoon”, “cup” since those are terms which are present in almost every ingredient list. An interesting finding can been seen under the topic “parsley”, “carrots”, and “celery”, where the recipe for “Jambalaya” appears, which is probably very unknown to most of the users. Note with the current filter settings of epicurious it would not have been possible to select the ingredients “parsley” and “celery”. Another interesting finding can be seen in Figure 5b where the ingredients “Bok Choy”, “Yams” and “Hoisin” appear in the recipe names which are probably also unknown to a wide range of users. To improve the generated facets of FacetX it might help to include more information like how the dish is prepared, how many calories it contains, or the reviews of users about the recipe.

3.3 Job Search

In this case study, we apply FacetX to the domain of job search on the job advertisement platform Monster.com. Figure 3 shows

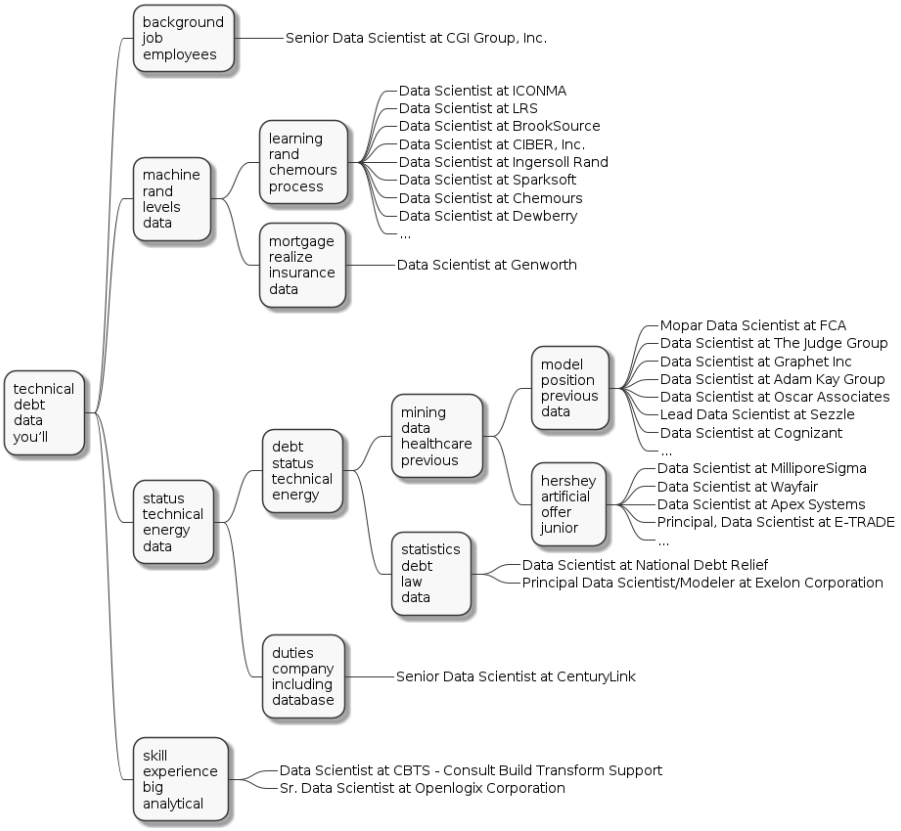

Figure 3: Example result of FacetX applied to the search results for the query "data scientist" on Monster.com

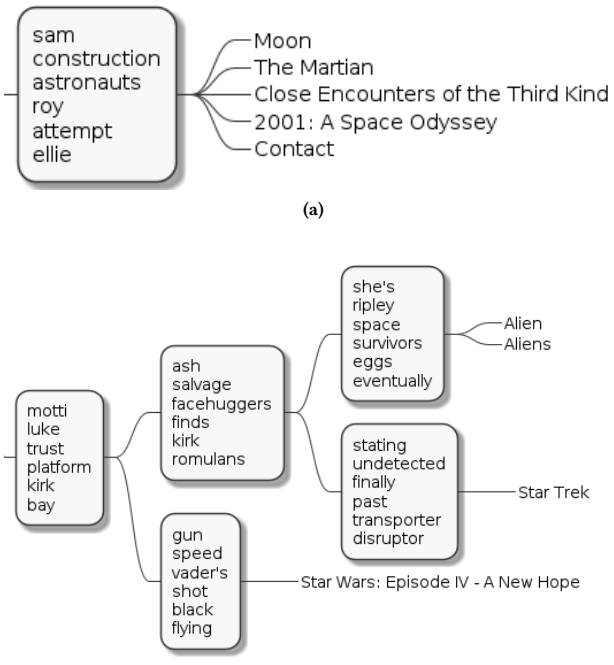

Figure 4: Generated facets of one cluster for the IMDB case study.

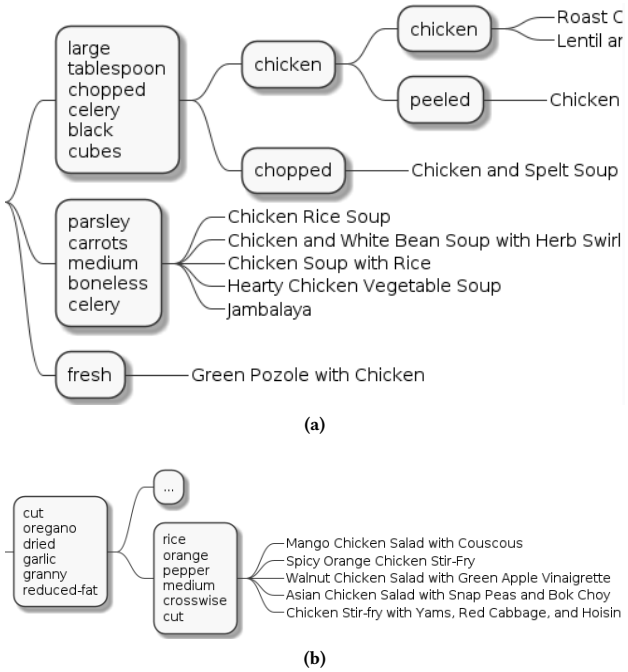

Figure 5: Sample facets of the recipe search case study for recipes with the ingredient chicken.

an example, which is the result of applying FacetX to the search results for the query "data scientist". From the resulting documents, the job title and the description were extracted for further processing. In this example a cluster size of eight was chosen and only the first four topics per node were extracted. For the modeling of the topics the job description of the results were used. The title of the job offerings is put at the leaves to give some idea how many documents would appear after those facets were chosen. After the search for job offerings with search terms on any online platform the users are always confronted with the same filter facets, also if they search for very different working areas or specific fields. These filter facets mostly contain the city or region of the workplace, the salary range, type (*e.g.*, beginner, experienced, manager) of the job offering, or the percentage of the workload. However, to be able to really understand what the job offering contains, the user needs to browse through every job description individually.

For example, if we search with the search term "gardener" without any further restrictions on Monster.com we would have to browse through a total of about 2500 job descriptions and for the search term "data scientist" it would be even more with a total of about 17000. Since Monster.com just offers filter facets like company, city, job status, or date posted we have no possibilities to further narrow down the results. But to be able to find the very best match between a user and a job offering it would be preferable for the job seeker, as well as for the company, to be able to use dynamic filter facets which are contained in the content of the job offering to drill down in the result set. With FacetX we would be able to dynamically create the filter facets based on the content of the job offerings. Furthermore, as seen in Figure 3 the user would be able to refine the results further by choosing the facets of the sub clusters. In this case study we searched for job offerings as a data scientist on Monster.com and extracted 52 results. We used FacetX to create facets for those results with different number of clusters and topics. With this a user can traverse down the hierarchical tree and reduce the search results by selecting the node with the topics, which seems the most interesting one. To get the best insight into a job offering the topics at the leaves seemed to be the most meaningful. There were topics like "healthcare", "federal" or "insurance" which can help in the decision whether pursuing the search further for those job offerings or not. The more topics are extracted per node the more information a user might gain about the content of the offerings but also more time has to be spent to read through the topics. Also, not all topics might be useful to the user, as for example the terms "job" or "position".

In this case study we also wanted to evaluated how FacetX performs in a more complex setting. We therefore searched for both jobs ("gardener" or "data scientist") on Monster.com. Note, with the filter functionality of Monster.com it would not be possible to separate the two very different job types from each other after the first search results are returned to the user. However, with the support FacetX additional filter facets were generated which successfully divided the job offerings from "data scientist" and "gardener" from each other (*cf.* Figure 6). Only two job offerings for gardener could be found in the cluster with mainly data scientists and in the cluster for gardener jobs 3 for data scientist were found. The extracted topics for the two branches included words like "deep", "model" and "process" for the data scientist branch and "landscape", "work" and "tree" for the other one.

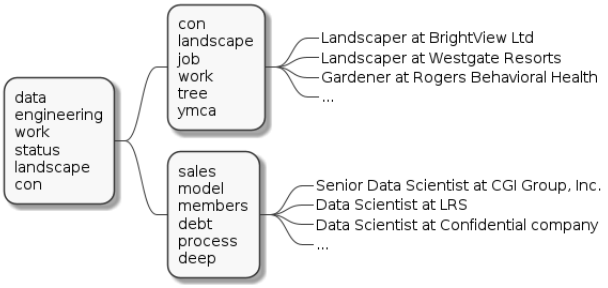

Figure 6: The first two branches of the dendrogram for the search term "gardener" and "data scientist" of the job search case study.

4 CONCLUSIONS AND FUTURE WORK

In this work-in-progress paper we introduce FacetX, a tool for the dynamic generation of facets for advanced information filtering. With FacetX it is possible to create dynamic facets for the result of an initial search query and gain more information about the retrieved documents without specific domain knowledge. FacetX could be integrated with any search engine as additional tool for the user to support the search and limiting the information overflow. In those settings where it is not possible to create meaningful facets the application still could be used to support the users in keeping the overview about the results. Future work includes the improvement of the facet generation by domain-specific stop word lists. Furthermore, other topic extraction techniques could be investigated for documents with less content. For example, the case study with the recipes demonstrated the support of FacetX is limited by the small amount of words in the recipes. Additionally, the number of topics for topic extraction could be made dynamically as the number of documents shrinks with increasing cluster numbers. The clustering part also needs more research in the case of how many clusters would be a good fit for a given number of documents and which linkage criteria is preferable to use.

REFERENCES

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3 (2003), 993–1022.

[2] Epicurious.com. 2019. *Epicurious Recipes, Menu Ideas, Videos & Cooking Tips.* Condé Nast Publications. Retrieved December 9, 2019 from https://www.epicurious.com/

[3] Kim Hak-Jin, Zhu Yongjun, Kim Wooju, and Sun Taimao. 2014. Dynamic faceted navigation in decision making using SemanticWeb technology. *Decision Support Systems* 61 (2014), 59–68.

[4] Mark Hall, Frank Eibe, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.* 11, 1 (2009), 10–18.

[5] IMDB.com. 2019. *Internet Movie Database, Ratings and Reviews for New Movies and TV Shows.* IMDb.com, Inc. Retrieved December 9, 2019 from http://www.imdb.com/

[6] Andrew Kachites McCallum. 2002. *MALLET: A Machine Learning for Language Toolkit.* http://mallet.cs.umass.edu/

[7] Monster.com. 2019. *Monster.com About.* Monster Worldwide, Inc. Retrieved December 9, 2019 from https://www.monster.com/about/

[8] Gerard Salton and Michael J. McGill. 1986. *Introduction to Modern Information Retrieval.* McGraw-Hill, Inc., New York, NY, USA.

[9] Seotribunal.com. 2019. *63 Fascinating Google Search Statistics.* Retrieved December 9, 2019 from https://seotribunal.com/blog/google-stats-and-facts/

[10] Daniel Tunkelang. 2009. Faceted Search. *Synthesis lectures on information concepts, retrieval, and services* 1, 1 (2009), 1–80.

[11] Michal Tvarozek and Maria Bielikova. 2007. Personalized Faceted Navigation for Multimedia Collections. In *Second International Workshop on Semantic Media Adaptation and Personalization (SMAP 2007).* IEEE, Piscataway, New Jersey, US, 104–109.