Towards Semantic Faceted Search

Marcelo Arenas[†]
Dept. of Computer Science
PUC Chile

Bernardo Cuenca Grau[‡]
Dept. of Computer Science
University of Oxford

Evgeny Kharlamov[‡]
Dept. of Computer Science
University of Oxford

Šarūnas Marciuška[‡]
Dept. of Computer Science
University of Oxford

Dmitriy Zheleznyakov[‡]
Dept. of Computer Science
University of Oxford

ABSTRACT

In this paper we present limitations of conventional faceted search in the way data, facets, and queries are modelled. We discuss how these limitations can be addressed with Semantic Web technologies such as RDF, OWL 2, and SPARQL 1.1. We also present a system, SemFacet, that is a proof-of-concept prototype of our approach implemented on top of Yago knowledge base, powered by the OWL 2 RL triple store RDFox, and the full text search engine Lucene.

1. MOTIVATION AND PROPOSAL

Faceted search is a technique for accessing document collections that combines text search and *faceted navigation* applied to the documents' metadata. With faceted navigation, users can narrow down search results by incrementally applying multiple filters called *facets* [6]. During the last decade, faceted search has become a mainstream commercial technology, and it is ubiquitous in e-commerce websites and online libraries. Despite the numerous success stories, however, traditional faceted search models impose severe constraints in the way *(i)* faceted metadata is represented, *(ii)* facets are defined, and *(iii)* queries are formulated [4, 14]. Pushing the boundaries of faceted search beyond the current state-of-the-art requires addressing several challenges, which we discuss next. To make the discussion concrete, suppose we are looking in a travel website such as *TripAdvisor* for accommodation in Seoul to attend the WWW 2014 conference. We look for a 4-star or 5-star hotel with a Korean or Japanese vegetarian restaurant.

Limitations of the data model. Classical faceted search models assume that documents are not "linked" to each other. We can start our search in *TripAdvisor* by filling in an initial form to obtain all available hotel documents in Seoul during the conference dates. The search can then be further refined by using the facets "hotel class" and "amenities" to select 4-star or 5-star hotels with restaurants. To complete our query, we need additional constraints about restaurant documents; however, the relevant facets are associated to restaurants, and not to hotels. Thus, we switch to the interface for restaurants, where we can use the available facets to select Japanese or Korean vegetarian-friendly restaurants in Seoul. Although the hotel-specific and the restaurant-specific "views" have in common the information provided in the initial search form (i.e., city and dates), there is no link between hotel and restaurant documents and hence the constraints we imposed to restaurants are not transferred to the hotel view. Thus, although many hotels featured in *TripAdvisor* satisfy our query, narrowing down the search to only those hotels requires significant manual browsing effort.

Limitations of the facet model. In their most basic form, facets consist of a heading and a set of values; e.g., hotel star ratings in *TripAdvisor* are modelled as a facet having one value for each 1-star to 5-star rating. Many applications, however, also define facets that are *hierarchical*. For example, accommodation in *TripAdvisor* is divided into hotels, B&B, and rentals; hotels into luxury, business etc. Hierarchical facets provide *background domain knowledge* which can be exploited to improve faceted search; however, they are still rather limited. Although a hierarchical facet establishes dependencies between its values, the underlying semantic relationship (e.g., "is-a", "part-of") is undefined. There are also issues concerning dependencies between facets, which cannot be represented in such a simple model. E.g, the type of hotel and the star ratings are correlated (e.g., motels cannot be 5-star); these dependencies are typically implemented ad-hoc, which negatively impacts systems' maintainability, performance, and reliability.

Limitations of the query model. The limitations above affect queries that users can pose. In particular, facet values for different kinds of documents cannot be joined in a single query. Thus, in *TripAdvisor* our example query cannot be formulated: even if we can query for both hotels or restaurants independently, when we "switch view" from hotels to restaurants, the constraints imposed on hotels are lost. Similar limitations were observed for faceted search over interlinked documents [3, 14], webpages [12], databases [7], dataspaces [17], and knowledge bases [3, 12]. Orthogonally, there are issues with the meaning of queries, which affect the way they are processed in the backend and their results are interpreted by users. In a faceted search front-end, users are presented with facets and allowed to make a multiple choice within each facet. Typically, choices in one facet are understood as logical OR, and constraints for different facets are combined with logical AND. Thus, if a user chooses "2-star" and "3-star", they are looking for hotels with two *or* three stars. Multiple choice in a single facet could also be interpreted conjunctively, e.g., when the users choses "WiFi" and "parking" facilities. Ambiguity is resolved in the backend when queries are translated into operations over inverted indicies. This process is application dependent, and it is not grounded on a formal query model that can be independently studied.

Semantic Faceted Search. RDF has been proposed by many authors as a promising technology to overcome some of the lim-

[†] Email: marenas@ing.puc.cl

[‡] Email: firstname.middlename.lastname@cs.ox.ac.uk

WWW'14 Companion, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577381.

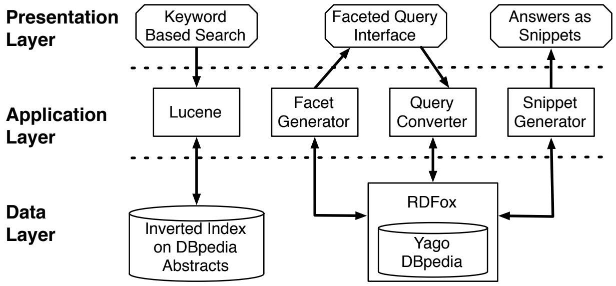

Figure 1: Architecture of SemFacet

itations of faceted search systems [2, 3, 5, 9, 10, 11, 12, 13, 15, 16, 18]. Although several RDF-based faceted search systems have been developed, there is a lack of rigorous understanding of the underlying data and query models. We aim at providing solid foundations to *semantic faceted search*—the extension of the faceted search paradigm with Semantic Web technologies. RDF was designed as a language for the representation of loosely-structured metadata, and it provides the required flexibility to semantically link different documents in arbitrary ways. OWL 2 can be used to provide rich domain knowledge on top of faceted metadata: OWL 2 axioms can capture hierarchical facets, and complex dependencies between facets in a declarative and semantically unambiguous way (e.g., business hotels cannot be 2-star, every 5-star hotels has a restaurant etc.). Finally, faceted queries can be captured by SPARQL 1.1, which provides well-understood semantics, computational properties, and powerful for query processing. Also, Semantic Web technologies provide important additional benefits. First of all, semantic facets and faceted query interfaces can be automatically generated from RDF and OWL 2 ontologies. Then, OWL 2 axioms can be used to specify which facets and values to display at each step of query refinement, thus providing valuable guidance to users. These techniques are orthogonal and complementary to the facet ranking mechanisms.

OWL 2 can also be used to simplify the annotation of documents with faceted metadata and deal with sparse and incomplete annotations. E.g., annotating data items with hierarchical facets is cumbersome since data must contain a value for each level of the hierarchy; in contrast, by representing hierarchies in OWL 2, we only need annotations for the most specific relevant values since the remaining ones can be automatically derived. Finally, semantic facets give a mechanism to query semantically related data sources, and hence are a natural query paradigm for ontology-enhanced linked data. We refer the reader to [1, 8] for more details on our approach.

2. THE SEMFACET SYSTEM

Our approach is general and can be used to provide faceted search over any RDF and OWL 2 ontology. To illustrate its potential in practice and assess the feasibility of our techniques, we implemented a prototypical faceted search system, called SemFacet (see [1, 8] for details), on top of (a fragment of) Yago [19] ontology and DBpedia containing around 15 million triples altogether.

A general architecture of SemFacet is in Figure 1. The backend relies on Lucene for keyword based search and RDFox, a massively parallel in-memory RDF triple store, for storing RDF triples, performing reasoning, and answering queries. SemFacet is implemented in such a way that both Lucene and RDFox can be substituted with any other software that provide the same functionality.

The front-end of SemFacet, by relying on nesting of conventional facets, allows users to formulate tree shaped SPARQL queries over RDF and OWL 2. The process of constructing queries is (see [8] for details): the first step is to provide a set of keywords, which leads to a set of initial answers and initial facets. Query refinement is then an iterative process, where users can either choose available facet values, or refocus the query to a different facet. In response the system updates the query answers as well as the facets available (they are automatically generated from the underlying RDF and OWL 2 ontology) to continue query refinement.

SemFacet also exploits OWL 2 axioms to enrich RDF data with implicit triples. This helps in addressing *sparsity* of annotations and modelling of *hierarchical* facets. Moreover, OWL 2 axioms help in avoiding "dead ends" (i.e., facet value selections that lead to queries with the empty answer). In conventional faceted search applications, the detection of such dead ends is data driven, in the sense that the interface does not display facet values for which no document exists. Axioms provide an alternative, declarative, way to detect dead ends during faceted search, e.g., by exploiting axioms expressing disjointness between classes of objects.

SemFacet is available as a Web service [1] and runs on a machine with 1vCPU, 4Gb of memory, and 20Gb of disk space. Although we have not formally evaluated our system, preliminary experiments show typical response time comparable with well known conventional faceted search systems.

3. REFERENCES

[1] SemFacet: Semantic Faceted Search Project. `http://www.cs.ox.ac.uk/isg/projects/SemFacet/`.

[2] T. Berners-Lee and et al. Tabulator redux: Browsing and writing linked data. In *LDOW*, 2008.

[3] S. Buschbeck and et al. A demonstrator for parallel faceted browsing. In *EKAW'12*, 2012.

[4] E. Clarkson, S. B. Navathe, and J. D. Foley. Generalized formal models for faceted user interfaces. In *JCDL'09*.

[5] J. Diederich, W.-Tilo Balke, and U. Thaden. Demonstrating the semantic growbag: automatically creating topic facets for FacetedDBLP. In *JCDL*, page 505, 2007.

[6] D.Tunkelang. *Faceted Search*. Morgan & Claypool Pubs.'09.

[7] G. H. L. Fletcher and et al. Towards a theory of search queries. *ACM Trans. Database Syst.*, 35(4):28, 2010.

[8] B.Cuenca Grau, E.Kharlamov, Š.Marciuška, D.Zheleznya−kov, M.Arenas, and E.Jimenez-Ruiz. Semfacet: Semantic faceted search over yago. In *WWW (Companion Volume)'13*.

[9] R. Hahn and et al. Faceted wikipedia search. In *BIS*, 2010.

[10] M. Hildebrand, J. van Ossenbruggen, and L. Hardman. /facet: A browser for heterogeneous semantic web repositories. In *ISWC'09*.

[11] D. Huynh and et al. Piggy bank: Experience the semantic web inside your web browser. *J. Web Sem.*, 2007.

[12] David F. Huynh and David R. Karger. Parallax and companion: Set-based browsing for the data web. www.davidhuynh.net.

[13] E. Hyvönen, S. Saarela, and K. Viljanen. Ontogator: Combining view- and ontology-based search with semantic browsing. In *XML Finland*, 2003.

[14] A. Jameson. How can we support multifocal exploration of semantic data? www.imash.leeds.ac.uk/event/keynote.html.

[15] G. Kobilarov and I. Dickinson. Humboldt: Exploring linked data. In *LDOW'08*.

[16] E. Oren, R. Delbru, and S. Decker. Extending faceted navigation for rdf data. In *ISWC*, 2006.

[17] Kenneth A. Ross and Angel Janevski. Querying faceted databases. In *SWDB*, pages 199–218, 2004.

[18] M. C. Schraefel and et al. The evolving mSpace platform: leveraging the Semantic Web on the trail of the memex. In *Hypertext*, 2005.

[19] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A Core of Semantic Knowledge. In *WWW 2007*.