Decision Support Systems 53 (2012) 425–437

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Faceted product search powered by the Semantic Web

Damir Vandic, Jan-Willem van Dam, Flavius Frasincar *

Erasmus University Rotterdam, PO Box 1738, NL-3000 DR Rotterdam, The Netherlands

ARTICLE INFO

Article history:
Received 17 June 2010
Received in revised form 17 October 2011
Accepted 15 February 2012
Available online 23 February 2012

Keywords:
Semantic Web
SPARQL and RDF
Product identification
Category mapping
Product search

ABSTRACT

This paper presents a platform for multifaceted product search using Semantic Web technology. Online shops can use a ping service to submit their RDFa annotated Web pages for processing. The platform is able to process these RDFa annotated (X)HTML pages and aggregate product information coming from different Web stores. We propose solutions for the identification of products and the mapping of the categories in this process. Furthermore, when a loose vocabulary such as the Google RDFa vocabulary is used, the platform deals with the issue of heterogeneous information (e.g., currencies, rating scales, etc.).

1. Introduction

Online product search, as a tool to help customers find their products of interest, has become more important than ever as consumers nowadays purchase more often on the Web [1]. This is due to the fact that there is an increase in product specificity and consumer preference variation. The most important reason for this is technical advancement, as this has led to a large increase of different product types. A second reason is that general wealth increase causes consumers to strengthen their preferences. The search space on the Web for products has also grown, which makes product search even more important.

There are several problems with the current state of product search on the Web. First, the search engines cannot deal properly with synonyms and homonyms. Second, there is no good support for multiple languages, and more importantly, the aggregation of Web-wide information is seldom done. This is clear when we analyze the way we search for products on the Web. We keep switching back and forth from search results to find, for example, the cheapest price of a certain product. It would be useful if the product information is aggregated and shown to the user in one unified view. Third, there is no parametric Web-wide search available. Users cannot use queries like ‘all solar panels which give 12A output and cost less than $2000’.

There are some localized, as opposed to Web-wide, product search Web sites where the user can perform this kind of parametric search. Usually these search engines only support basic product properties. Examples of these properties are the brand, the price, and the review rating of a product. Shopping.com, Google Products, and Shopzilla.com are three well-known parametric product search engines of such kind.

A user can search, for example, for a washing machine with a maximum price of £750 of the brand ‘Bosch’. Fig. 1 shows an example of this search. The user specifies the query constraints and the search engine queries the database, which contains all products, in order to display the washing machines that fulfill the requirements of the user. As a result of this, only stores that are indexed in the database of the search engine are shown.

The databases of these kinds of search engines are updated through application programming interfaces (APIs) of Web shops that sell products. Of course, not every Web shop has an API and/or data feed possibilities. Furthermore, every search engine has its own standards which have to be obeyed by the Web shops. For instance, the API of Shopping.com is different than the API of Shopzilla. This means that not every Web shop will have their data prepared for both Shopping.com and Shopzilla. As it is costly to adjust data to a standard, it is not likely that a single search engine will receive data from all Web shops. By annotating Web pages with information on the Semantic Web, the APIs of nowadays can be made obsolete. The annotated information is also publicly available, which enables a search engine to gather product information directly from Web pages.

There is one severe consequence of the current situation of product search. Because a user is not going to view all presented search results, there is a chance that (s)he cannot precisely find a product that matches his or her criteria. What happens is that users more quickly start to focus on the price and give less weight to the product features. The result is that a fierce price competition arises. This can be considered negative for both consumers and companies, as a user can prefer a product that meets all requirements but has a slightly higher price.

* Corresponding author.
E-mail addresses: vandic@ese.eur.nl (D. Vandic), info@qdentity.nl (J.W. van Dam), frasincar@ese.eur.nl (F. Frasincar).

0167-9236/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2012.02.010

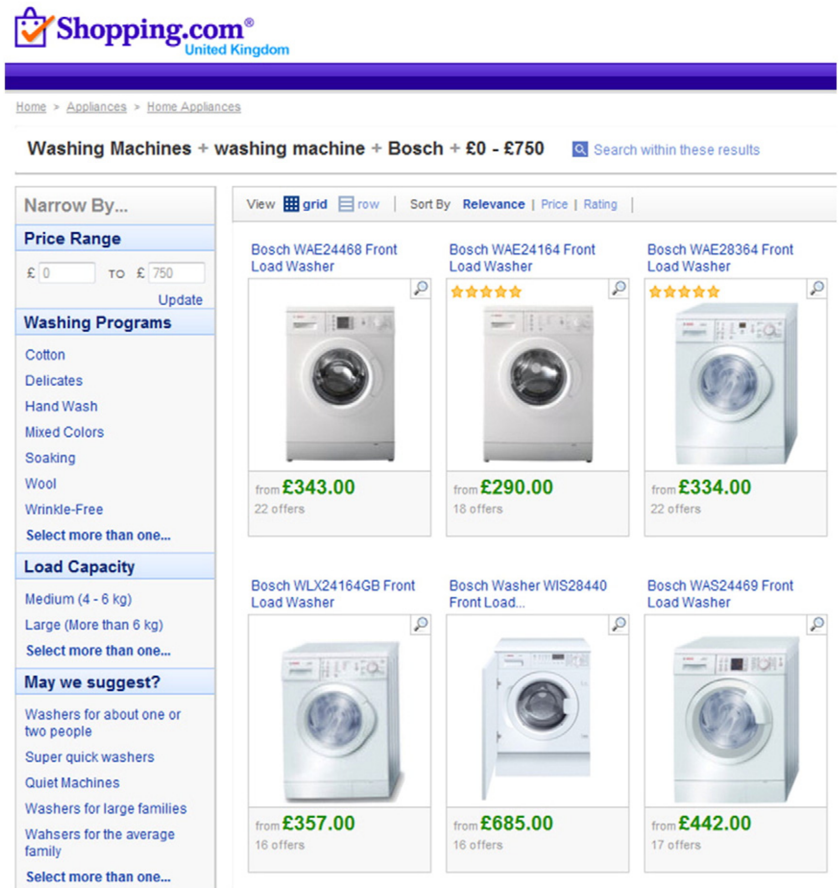

Fig. 1. Shopping for a 'Bosch' washing machine on Shopping.com.

With semantic search, the companies can develop new business models, because consumers can find the very specific products they are searching for, easier than before.

We have seen how Semantic Web can enhance online product search. In order to make this work, automatic aggregation of product information has to be made possible. The main goal of this research is to show how we can effectively aggregate product information from different sources. In this paper, we focus particularly focus on aligning product names (product identification) and categories (category mapping) from different Web shops. Product identification is the first step when an annotated Web page is processed. The actual product that is described on that Web page has to be identified in order to be able to perform aggregation of product properties (e.g., prices, reviews, etc.). Category mapping is necessary in order to make it possible for users to use the category facet in the user interface. Each product category that occurs in the product description needs to be mapped to a category from the internal product category hierarchy. As part of this research, we provide an implementation of this platform, the XploreProducts.com Web application, which uses a multi-faceted user interface and is available online at http://www.xploreproducts.com. We do not cover product search engine algorithms, as our focus lies on the aggregation of product information. Further, we propose a solution that solves several issues that current product search engines face. These issues are based on the fact that (product) information on pages is often not well-structured. Product information is usually annotated using a loose vocabulary because there is no widely accepted standard for product names, categories, currencies, scales, units, etc. [2].

Web shops can provide their product offerings on XploreProducts.com simply by supplying the URLs of their product Web pages. This is called the *ping service* of XploreProducts.com. The provided URLs should point to Web pages that have been annotated using the data-vocabulary.org RDFa vocabulary. We choose to use this vocabulary because it is easy to use and already supported by Google. This vocabulary is a good example of a simple and non-restrictive (loose) vocabulary. A loose vocabulary is characterized by the fact that the *range* of properties is not well-defined. For example, for the range of the *price* property in the data-vocabulary.org RDFa vocabulary, it is not specified if it should be a string, float data type, or some other data type. Some users can specify '$199!' as the price while others might use '199.99'. This makes it easy to use by users, but poses issues for automatic aggregation of information.

Although Google is supporting the data-vocabulary.org vocabulary, it is not using it to improve the search engine by aggregating the product information that is annotated on Web pages. The search queries are still based on keywords, like in every traditional search engine. Currently, Google is only using these annotations for the presentation of their search results. Fig. 2 shows a screen shot of one of the results of a search query on Google.com. As a result of annotating the Web page of the 'Art of Pizza' restaurant on yelp.com, Google can show information about the price, reviews, etc. We expect that Google's support for these, so-called, rich snippets [3] will lead to a fast adoption of RDFa and/or microformats, to improve search, presentation, and ranking of information.

Fig. 2. A result of searching for 'Art of Pizza restaurant' on Google.com.