Information Systems 56 (2016) 36–54

Contents lists available at ScienceDirect

Information Systems

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

A tool for producing structured interoperable data from product features on the web

Tuğba Özacar *

Department of Computer Engineering, Celal Bayar University, Muradiye, 45140 Manisa, Turkey

ARTICLE INFO

Article history:
Received 11 March 2015
Accepted 7 September 2015
Recommended by: F. Carino Jr.
Available online 25 September 2015

Keywords:
Information extraction
GoodRelations
Protégé
Web scraping
Ontology
Rich snippets

ABSTRACT

This paper introduces a tool that produces structured interoperable data from product features, i.e., attribute name–value pairs, on the web. The tool extracts the product features using a web site-specific template created by the user. The value of the extracted data is maximized by using GoodRelations, which is the standard vocabulary for modeling product types and their features. The final output of the tool is GoodRelations snippets, which contain product features encoded in RDFa or Microdata. These snippets can be embedded into existing static and dynamic web pages in a way accessible to major search engines like Google and Yahoo, mobile applications, and browser extensions. This increases the visibility of your products and services in the latest generation of search engines, recommender systems, and other novel applications.

1. Introduction

The web contains a huge number of online shops which provide excellent resources for product information. Besides, the data of e-commerce is growing at a rapid speed [1]. Information in e-commerce includes technical specifications and descriptions of products. If we present this information in a structured way, it will significantly improve the effectiveness of many applications [2].

The vast majority of web content consists of different kinds of textual documents, which are provided in a number of different formats and vary from plain text to semi-structured documents containing data records. This makes different methods of bringing structure and semantics to the web (including web information extraction) an active research field [3]. Although the web has a dynamic nature, Etzioni has argued for that "information on the web is sufficiently structured to facilitate effective web mining" [4]. Since a big portion of web content subject to web information extraction is created from data repositories, a web information extraction system rediscovers the structure that was encoded in a web page.

This paper introduces a tool[1] that produces structured interoperable data from product features, i.e., attribute name–value pairs, on the web. It extends the previous work of the author [5] in two ways. First it supports tree nodes that define text operations (e.g. concatenate, contains, fragment, lower, upper, replace, substring, and trim) on tree nodes. Second it presents a user-based evaluation accomplished using 15 different "real world" scenarios.

Designed as a plug-in for the open source ontology editor Protégé [6], the proposed tool exploits the advantages of the ontology as a formal model for the domain knowledge.

* Tel.: +90 236 2012103; fax: +90 236 2412143.
E-mail address: tugba.ozacar@cbu.edu.tr

[1] Download link: https://github.com/tugbaozacar/iris

http://dx.doi.org/10.1016/j.is.2015.09.002
0306-4379/

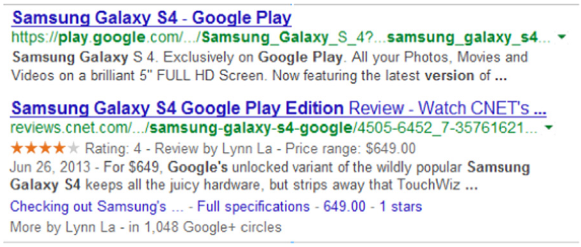

Fig. 1. A regular snippet and a rich snippet.

Another promising feature of the tool is support for building an ontology that is compatible with GoodRelations Vocabulary [7]. GoodRelations is the most powerful vocabulary for publishing all of the details of your products and services in a way friendly to search engines, mobile applications, and browser extensions. In [8], GoodRelations product ontology is defined as a "product atlas" describing specifications, marketing copy, catalog data, photos, videos, manuals, installation instructions, updates, responses to issues, prices and reviews of over 1 million products from various sources. It updates twice a week, so companies can enter all their descriptions and prices and the information will flow through the thousands of e-commerce systems within a few days. The goal is to have extremely deep information on millions of products, providing a resource that can be plugged into any e-commerce system without limitation.

If you have GoodRelations in your markup, Google, Bing, Yahoo, and Yandex will or plan to improve the rendering of your page directly in the search results. Rich snippets—the few lines of text that appear under every search result—are designed to give users a sense for what is on the page and why it is relevant to their query. Fig. 1 shows the difference between a regular and a rich snippet. The first search result is a regular snippet and the second one is a rich snippet. The proposed tool supports marking up your content with RDFa or Microformats for creating rich snippets of the extracted products.

In addition to seeing rich snippets in the search results, rich markup indicates the relevance of your page for a particular query. You provide information to the search engines so that they can rank up your page for queries to which your offer is a particularly relevant match. For many popular shop applications including Drupal Commerce, Magento, Prestashop, and Rakuten.de/Tradoria.de, Joomla/Virtuemart, there exist free extension modules that make adding GoodRelations RDFa for semantic SEO as simple as a few mouse-clicks.

The tool supports "Open Standards" including RDFa, Microdata and JSON. International Telecommunications Union (ITU-T) specifies that "Open Standards" facilitate interoperability and data exchange among different products or services and are intended for widespread adoption. In [9] the key benefits of interoperable data are listed as follows: enabling information sharing with trusted partners, enhancing system capabilities and longevity, lowering overall costs of information applications, improving the breadth and quality of information, increasing the speed and accuracy of decisions, improving transparency and speed of disclosure of information to valid constituents, preserving data for future uses.

The organization of the article is as follows: In Section 2, we review background information and related work. Section 3 includes an overview of the system's architecture, features and settings and a scenario based quick-start guide. Section 4 presents a user-based evaluation accomplished using 15 different "real world" scenarios. Finally, Section 5 concludes the article with a brief talk about possible future work.

2. Background knowledge and related work

The information extraction systems can be divided into following three categories [10]:

- *Procedural wrapper*: The approach is based on writing customized wrappers for accessing required data from a given set of information sources. In these systems the extraction rules are coded into the program.
- *Declarative wrapper*: These systems consist of a general execution engine and declarative extraction rules developed for specific data sources.
- *Automatic wrapper*: These systems use machine learning techniques to learn extraction rules by examples.

In [11] information extraction systems are divided with respect to the level of automation of wrapper creation into *manual*, *semi-automatic* (i.e., based on user interaction with user interface) and *automatic* (using supervised machine learning techniques). In this paper, we present a tool that has two main components: a wrapper and an ontology builder.

The main component of the proposed tool is a declarative and manual wrapper, which has a general rule engine that executes the extraction rules specified in a template file created manually by the user. The extraction rules are specified using our domain-specific language (Appendix D) based on XML Path Language (XPath) [12]. XPath is a query language that can be used for selecting arbitrary parts of HTML documents. In addition to specifying the XPath query, these rules also specify how to convert the extracted information to the internal object format. The template file also provides information on how the HTML documents should be acquired.