Clustering Visually Similar Web Page Elements for Structured Web Data Extraction

Tomas Grigalis[1], Lukas Radvilavičius[1], Antanas Čenys[1], and Juozas Gordevičius[2]

[1] Vilnius Gediminas Technical University, Lithuania
{tomas.grigalis,lukas.radvilavicius,antanas.cenys}@vgtu.lt
[2] Vilnius University Institute of Mathematics and Informatics, Lithuania
juozas.gordevicius@mii.vu.lt

Abstract. We propose a novel approach for extraction of structured web data called ClustVX. It clusters visually similar web page elements by exploiting their visual formatting and structural features. Clusters are then used to derive extraction rules. The experimental evaluation results of ClustVX system on three publicly available benchmark data sets outperform state-of-the-art structured data extraction systems.

1 Introduction

Automatic extraction of structured data from web pages is one of the key challenges for the Web search engines to advance into a more expressive semantic level. However, current algorithmic approaches often fail to achieve satisfactory performance in real-world application scenarios due to abundant structurally complicated and dynamic WEB 2.0 pages.

Information extraction systems can be broadly divided into supervised and unsupervised categories. Supervised learning approaches, such as Lixto [1], require some manual human effort to derive the extraction rules, while automated information extraction systems [2,3,4,5] work automatically and need no manual intervention. In this work we focus on the latter as we believe that only fully automatic systems can be applied for web-scale data extraction.

Thus we present a novel stuctured web data extraction system, ClustVX, which is fully automatic, scalable, and domain independent. ClustVX is based on two fundamental observations. First, vast amount of information on the Web is presented using fixed templates and filled with data from underlying databases. For example, Fig. 1(a) shows three Data Records (DRs) representing information about three digital cameras in an online store. The three DRs are listed according to some unknown to us style template and the information comes from a database. This also means, that each DR has almost the same Xpath (tag path from root node in HTML tree to particular web page element), where only a few node numbers differs. Second, although the templates and underlying data differ from site to site, humans understand it easily by analyzing repeating visual patterns on a given Web page [6]. We hypothesize, that the data which

M. Brambilla, T. Tokuda, and R. Tolksdorf (Eds.): ICWE 2012, LNCS 7387, pp. 435–438, 2012.

has the same semantic meaning is visualized using the same style. For example in Fig. 1(a) prices are brown red and bold, title is green and bold, text "Online Price" is grey.

ClustVX exploits both of these two observations by representing each web page element with a combination of its Xpath and visual features such as font, color and etc. For each visible web page element we encode this combination into the string called Xstring. Clustering Xstrings allows us to identify visually similar elements, which are located in the same region of a web page and in turn have same semantic meaning. See Fig. 1(b) where price elements are clustered together according to their Xstring. Subsequent data extraction leads to a machine readable structured data that is shown in Fig. 1(c).

Samsung ES80 **$84.95** Online Price | **Fujifilm FinePix T300** **$174.95** Online Price | **Vivitar ViviCam F529** **$84.95** Online Price

(a) An example of three digital cameras (Data Records) in a web page

Xstring:	htmlbodydivdiva-Verdana,FF6600;400
$84.95	/html/body/div[1]/div[3]/a
$174.95	/html/body/div[2]/div[3]/a
$84.95	/html/body/div[3]/div[3]/a

(b) A cluster with visually similar price elements

Image 1	Samsung ES80	$84.95	Online Price
Image 2	Fujifilm FinePix T300	$174.95	Online Price
Image 3	Vivitar ViviCam F529	$84.95	Online Price

(c) Desired extraction result

Fig. 1. An example of structured web data extraction using ClustVX

2 The Proposed Approach

The ClustVX processes a given Web page in the following steps:

1. A web page is retrieved and rendered in a contemporary web browser. This is very importand step, since web browser handles all WEB 2.0 features, such as client-side scripting, AJAX requets and etc. All visual styling information from HTML source code and CSS files is also processed by the browser.
2. All HTML text formatting tags, such as <b>, , are removed from a web page. This is done to enhance clustering accuracy.

Table 1. The details of three public benchmark data sets used for ClustVX evaluation

Data Set	TBDW [7]	ViNTs-2 [8]	Alvarez [2]
Sites	51	102	200
Pages per site	5	11	1
AVG records	21	24	18
Total records	1052	2489	3557

3. An Xstring representation is generated for each visible web page text element. As we see in Fig. 1(b) Xstring consists of a) tag names from Xpath b) visual features of that element (font style, color, weight, etc.). Structural features (string of tag names) identifies position in HTML document. Visual features, which are obtained from web browsers API, enhance understanding of semantic similarity between web page elements.
4. All visible web page elements are clustered according to their Xstring. Resulting clusters contain only semantically similar web page elements. In Fig. 1(b) we see a cluster of price elements that all have the exaclty same Xstring and therefore belong to the same cluster.
5. Extraction of structured data. This process is based on two observations about DR representation in a web page. First, a group of DRs are usually rendered in a contiguous region of a web page [5] and are visually similar. Second, a group of DRs are formed by some child subtrees and at some level have same parent node [5]. Thus, by calculating longest common prefix of Xpaths from each cluster , we can find the exact locations of DRs groups (Data Regions) in a page. For a simple example, consider the Fig. 1(b), where Xpaths of clustered price elements are located. First, we find the longest common prefix (/html/body) of these clustered Xpaths. The prefix leads us to the particular region of a web page, where DRs are located. Then, the longest common suffix (/div[3]/a) is items' path in the DR. The Xpath substring between prefix and suffix (/div[*]) is used to segment Data Region into DRs. All clusters that have the same longest common Xpath prefix represent one particular Data Region. There may exist many Data Regions in one page and ClustVX locates them all.

3 Experimental Evaluation

We evaluate ClustVX using the three publicly available benchmark data sets containing in total of 7098 DRs from 353 different template web sites. See Tab. 1 for details. These data sets contain web pages retrieved from different web sites. Each web page contains DRs, which should be extracted. To evaluate ClustVX we take only one web page per site because all pages in one site use the same template.

We compare the evaluation results of ClustVX system to these state-of-the-art automatic structured data extraction systems: M. Alvarez et. al. [2], G-STM

Table 2. Experimental evaluation results of ClustVX system compared to other state-of-the-art methods

Data Set	TBDW				VINTS-2			Alvarez	
System	ClustVX	G-STM	DEPTA	FiVaTech	ClustVX	G-STM	DEPTA	ClustVX	Alvarez
Precision	**99.81**%	99.80%	99.50%	97.00%	**98.57**%	98.50%	95.10%	**98.20**%	97.90%
Recall	**99.52**%	96.60%	85.30%	97.40%	**98.51**%	96.70%	83.90%	**99.69**%	98.30%

[3], FiVaTech [4] and DEPTA [5]. Since none of these systems are available to download, we use the evaluation results reported in corresponding publications. As shown in Tab. 2, where best results are marked in bold, ClustVX consistently outperforms other approaches.

4 Conclusions and Research Directions

We have introduced a novel approach, ClustVX, to extraction of structured data from web pages. It uses structural as well as visual features of web page elements to discover the structure of underlying data. Evaluation on three publicly available benchmark data sets demonstrated, that the method consistently achieves very high quality in terms of precision and recall and outperforms other approaches.

Our future work will focus on evaluation of ClustVX on contemporary real-world web pages that are full of Java Scripts and are dynamic. The existing benchmark data sets lack features introduced by Web 2.0, such as, AJAX. Although we stipulate that ClustVX is invariant to these advanced features, a proper dataset is necessary to prove its applicability in real-world settings.

References

1. Baumgartner, R., Flesca, S., Gottlob, G.: Visual web information extraction with lixto. In: Proc. VLDB, pp. 119–128 (2001)
2. Álvarez, M., Pan, A., Raposo, E.A.: Extracting lists of data records from semi-structured web pages. Data & Know. Engineering 64(2), 491–509 (2008)
3. Jindal, N., Liu, B.: A generalized tree matching algorithm considering nested lists for web data extraction. In: The SIAM Int. Conf. on Data Mining, pp. 930–941 (2010)
4. Kayed, M., Chang, C.: Fivatech: Page-level web data extraction from template pages. IEEE Trans. on Know. & Data Engineering 22(2), 249–263 (2010)
5. Zhai, Y., Liu, B.: Web data extraction based on partial tree alignment. In: Proc. WWW, pp. 76–85. ACM (2005)
6. Miao, G., Tatemura, J., Hsiung, W., Sawires, A., Moser, L.: Extracting data records from the web using tag path clustering. In: Proc. WWW, pp. 981–990. ACM (2009)
7. Yamada, Y., Craswell, N., Nakatoh, T., Hirokawa, S.: Testbed for information extraction from deep web. In: Proc. WWW, pp. 346–347. ACM (2004)
8. Zhao, H., Meng, W., Wu, Z., Raghavan, V., Yu, C.: Fully automatic wrapper generation for search engines. In: Proc. WWW, pp. 66–75. ACM (2005)