RESEARCH ARTICLE OPEN ACCESS

A Survey on Data Extraction using Visual features

R.Anand Kumar[1], P.Jamuna[2], I.Sudha[3],
Department of CSE, Pondicherry University, Puducherry-India

ABSTRACT
The process of retrieving the exact data that is needed by the user is known as Data Extraction. Extracting the exact data from web pages is a complex problem because the data's in the database present in a complex structure. Many techniques have been proposed but all of them have some limitations because they are web page programming language dependent. In this paper a vision based approach that is web page programming language independent is proposed. Here the visual features like web page layout, font attributes and image size are used to retrieve the data in an effective manner.

Keywords-Data extraction, Visual features

I. INTRODUCTION

The World Wide Web has more and more online Web databases which can be searched through their Web query interfaces. The number of Web databases has reached 25 million according to a recent survey. All the Web databases make up the deep Web (hidden Web or invisible Web). Often the retrieved information (query results) is enwrapped in Web pages in the form of data records. These special Web pages are generated dynamically and are hard to index by traditional crawler based search engines, such as Google and Yahoo. In this paper, we call this kind of special Web pages deep Web pages. Collection of data is known as data item and collection of data item is known as data records In this paper, we study the problem of automatically extracting the structured data, including data records and data items, from the deep Web pages. The problem of Web data extraction has received a lot of attention in recent years and most of the proposed solutions are based on analysing the HTML source code web-page-programming-language- dependent, or more precisely, HTML-dependent.

In this paper a Vision based approach Data Extractor (ViDE) that is web page programming language independent is proposed. Here we use the visual features like web page layout and fonts. ViDE consists of two main components, Vision based Data Record extractor (ViDRE) and Vision-based Data Item extractor (ViDIE).

Our approach employs a four-step strategy. First, given a sample deep Web page from a Web database, obtain its visual representation and transform it into a Visual Block tree; second, extract data records from the Visual Block tree; third, partition extracted data records into data items and align the data items of the same semantic together; and fourth, generate visual wrappers (a set of visual extraction rules) Our implementation uses the VIPS algorithm [4] to obtain a deep Web page's Visual Block tree and VIPS needs to analyse the HTML source code of the page, our solution is independent of any specific method used to obtain the Visual Block tree in the sense that any tool that can segment the Web pages into a tree structure based on the visual information, not HTML source code, can be used to replace VIPS in the implementation of ViDE.

Section 2 presents the survey of well-known evaluation of data extraction and section 3 concludes the work.

II. SURVEY ON AVAILABLE TECHNIQUES

In this section, we provide a survey of Data extraction using visual features.

G.O. Arocena and A.O. Mendelzon [1] given that the widespread use of internet has created many data management problems such as extracting data from Web pages and making databases accessible from Web Browsers. These problems increased the problem of querying graphs; semi structured data and structured documents. Several systems and languages have been proposed to solve the web Data Management problem but, none could able to solve the problem. These problems restructure the original data in the web, so in this paper we are using WebOQL system whose goal is to structure the restructure data in the web.

D. Buttler, L. Liu, and C. Pu [2] deals with fully automated object extraction system – Omini. The unique feature of Omini is the use of set of algorithms and information extraction rules for extracting objects from dynamic web pages. Omini parses web pages into tree structures and performs object extraction in two stages. First it uses a set of sub tree extraction algorithm to locate the smallest sub tree that contains all the objects. Second, it employs object extracting algorithm to find the correct object. Both these stages are fully automated.

D. Cai, X. He, J.-R. Wen, and W.-Y. Ma [3] use Link Analysis to improve the performance of web-search. Page Rank and HITS are the two popular algorithms used for searching. These link analysis algorithms treat a web page as a single node in the web graph, but a web page contains multiple nodes. Here the web pages are partitioned into blocks using vision based page segmentation algorithm. By extracting page-to-block and block-to-page relationships from link structure and page layout analysis, we construct a semantic graph over the WWW such that each node exactly represents a single node.

D. Cai, S. Yu, J. Wen, and W. Ma [4] discussed that the content in the web are structured based on the visual representation. Many web applications such as information retrieval, information extraction and automatic page adaptation can benefit from this structure. It presents an automatic top-down independent approach to detect web content structure. This technique is independent to documentation representation such as HTML and works well even when the HTML structure is different from layout structure.

C.-H. Chang, M. Kayed, M.R. Girgis, and K.F. Shaalan [5] surveys the major web extraction approaches and compares them in three dimensions: the task domain, the automation degree, and the techniques used. The first dimension explains why an IE system fails to handle some web sites of particular structures. The second dimension classifies IE systems based on the techniques used. The third dimension measure the degree of automation for IE system.

C.-H. Chang, C.-N. Hsu, and S.-C. Lui [6] proposes a pattern discovery approach that extract structured data from semi-structured web documents. In this paper, we introduce IEPAD (Information Extraction based on Pattern Discovery), a system that describes extraction patterns from web pages without user-labelled examples. IEPAD applies several pattern discovery techniques, including PAT-trees, multiple string alignments and pattern matching algorithms. Extractors generated by IEPAD can correctly extract structured data records with attribute values.

V. Crescenzi, G. Mecca, and P. Merialdo [7] investigates techniques for extracting data from HTML sites through the use of automatically generated wrappers. To automate the wrapper generation and the data extraction process, the paper develops a novel technique to compare HTML pages and generate a wrapper based on their similarities and differences. Experimental results on real-life data-intensive Web sites confirm the feasibility of the approach.

D.W. Embley, Y.S. Jiang, and Y.-K. Ng [8] tells extraction of information from unstructured or semi structured Web documents requires a recognition and boundaries of records. Without dividing the large documents, extraction of record information will not succeed. In this paper we describe a approach to discover record boundaries in web documents. In this approach the documents are structures as a tree of nested HTML tags, locate the sub tree containing the records of interest identify candidate separator tags within the sub tree using five independent heuristics, and select a consensus separator tag based on a combined heuristic.

W. Gatterbauer, P. Bohunsky, M.Herzog, B. Krpl, and B. Pollak [9] have given information extraction from web tables is based on the use of <table> tags. A multitude of different HTML implementations of web tables make these approaches difficult to scale. In this paper, we approach the problem of domain independent information extraction from web tables by shifting our attention from the tree-based representation of web pages to a variation of the two-dimensional visual box model used by web browsers to display the information on the screen. The thereby obtained topological and style information allows us to fill the gap created by missing domain-specific knowledge about content and table templates.

J. Hammer, J. McHugh, and H. Garcia-Molina [10] described the management of semi-structured data, i.e., data that has irregular or dynamically changing structure. They describe components of the Sandford Tsimmis project that help to extract semi-structured data from web pages. It allows storage and querying of semi-structured data and that allowing means browsing through the World Wide Web.

C.-N. Hsu and M.-T. Dung [11] integrate a large number of web information sources may increase the use of the World Wide Web. A solution to the integration is through the use of a Web Information mediator that provides smooth, transparent access for the clients. Information mediators need wrappers to access a web source as a structured database, but building wrappers by hand is impractical. This paper presents Soft Mealy, a novel wrapper representation. This representation is based on a finite-state transducer (FST) and contextual rules. This approach can wrap a wide range of semi structured web pages.

N. Kushmerick [12] advocate wrapper induction, a technique for automatically constructing wrappers. In this article, we describe six wrapper classes, and use a combination of empirical and analytical techniques to evaluate the computational tradeoffs among them. We first consider *expressiveness*: how well the classes can handle actual Internet resources, and the extent to which wrappers in one class can mimic those in another. We then turn to *efficiency*: we measure the number of examples and time required to learn wrappers in each class, and we compare these results to PAC models of our task and asymptotic complexity analyses of our algorithms.

A. Laender, B. Ribeiro-Neto, A. da Silva, and J. Teixeira [13] proposed in the literature to address the problem of web data extraction use techniques borrowed from areas such as natural language processing, languages and grammars, machine learning, information retrieval, databases, and ontologies. As they provide distinct features and capabilities, direct comparison is difficult. In this paper, they proposed taxonomy for characterizing web data extraction tools, briefly survey major web data extraction tools described in the literature, and provide a qualitative analysis of them.

B. Liu, R.L. Grossman, and Y. Zhai [14] propose a more effective technique to perform the task. The technique is based on two observations about data records on the Web and a string matching algorithm. The proposed technique is able to mine both contiguous and non-contiguous data records.

W. Liu, X. Meng, and W. Meng [15] proposed a language dependent technique to solve the data extraction problem. This proposed solution performs the extraction based on the visual information of the response pages. They also proposed a new measure revision to evaluate the extraction performance. This method can achieve very high extraction accuracy.

Y. Lu, H. He, H. Zhao, W. Meng, and C.T. Yu [16] present a multi-annotator approach that first aligns the data units into different groups such that the data in the same group have the same semantics. Then for each group, we annotate it from different aspects and aggregate the different annotations to predict a final annotation label. An annotation wrapper for the search site is automatically constructed and can be used to annotate new result pages from the same site. Our experiments indicate that the proposed approach is highly effective.

J. Madhavan, S.R. Jeffery, S. Cohen, X.L. Dong, D. Ko, C. Yu, and A. Halevy [17] highlight these challenges in two scenarios – the Deep Web and Google Base. We contend that traditional data integration techniques are no longer valid in the face of such heterogeneity and scale. We propose a new data integration architecture, PAYGO, which is inspired by the concept of data spaces and emphasizes pay-as-you-go data management as means for achieving web-scale data integration.

I. Muslea, S. Minton, and C.A. Knoblock [18] introduce an inductive algorithm, stalker that generates high accuracy extraction rules based on user-labeled training examples. Labeling the training data represents the major bottleneck in using wrapper induction techniques, and our experimental results show that stalker requires up to two orders of magnitude fewer examples than other algorithms. Furthermore, stalker can wrap information sources that could not be wrapped by existing inductive techniques.

Z. Nie, J.-R. Wen, and W.-Y. Ma [19] explore a new paradigm to enable web search at the object level in this paper, extracting and integrating web information for objects relevant to a specific application domain. We then rank these objects in terms of their relevance and popularity in answering user queries. In this paper, we introduce the overview and core technologies of object-level vertical search engines that have been implemented in two working systems: Libra Academic Search (http://libra.msra.cn) and Windows Live Product Search (http://products.live.com).

A. Sahuguet and F. Azavant [20] present the World Wide Web Wrapper Factory (W4F), a toolkit for the generation of wrappers for Web sources, that o_ers: (1) an expressive language to specify the extraction of complex structures from HTML pages; (2) a declarative mapping to various data formats like XML; (3) some visual tools to make the engineering of wrappers faster and easier.

III. CONCLUSION

In this paper, we focused on the structured Web data extraction problem, including data record extraction and data item extraction. First, we surveyed previous works on Web data extraction and investigated their inherent limitations. We found that the visual information of Web pages can help us implement Web data extraction. Based on our observations of a large number of deep Web pages, we identified a set of interesting common visual features that are useful for deep Web data extraction. Based on these visual features, we proposed a novel vision-based approach to extract structured data from deep Web pages

IV. REFERENCES

[1] G.O. Arocena and A.O. Mendelzon, "WebOQL: Restructuring Documents, Databases, and Webs," Proc. Int'l Conf. Data Eng. (ICDE), pp. 24-33, 1998.

[2] D. Buttler, L. Liu, and C. Pu, "A Fully Automated Object Extraction System for the World Wide Web," Proc. Int'l Conf. Distributed Computing Systems (ICDCS), pp. 361-370, 2001.

[3] D. Cai, X. He, J.-R. Wen, and W.-Y. Ma, "Block-Level Link Analysis," Proc. SIGIR, pp. 440-447, 2004.

[4] D. Cai, S. Yu, J. Wen, and W. Ma, "Extracting Content Structure for Web Pages Based on Visual Representation," Proc. Asia Pacific Web Conf. (APWeb), pp. 406-417, 2003.

[5] C.-H. Chang, M. Kayed, M.R. Girgis, and K.F. Shaalan, "A Survey of Web Information Extraction Systems," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 10, pp. 1411-1428, Oct. 2006.

[6] C.-H. Chang, C.-N. Hsu, and S.-C. Lui, "Automatic Information Extraction from Semi-Structured Web Pages by Pattern Discovery," Decision Support Systems, vol. 35, no. 1, pp. 129-147, 2003.

[7] V. Crescenzi, G. Mecca, and P. Merialdo, "RoadRunner: Towards Automatic Data Extraction from Large Web Sites," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 109-118, 2001.

[8] D.W. Embley, Y.S. Jiang, and Y.-K. Ng, "Record-Boundary Discovery in Web Documents," Proc. ACM SIGMOD, pp. 467- 478, 1999.

[9] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Krpl, and B. Pollak, "Towards Domain Independent Information Extraction from Web Tables," Proc. Int'l World Wide Web Conf. (WWW), pp. 71-80, 2007.

[10] J. Hammer, J. McHugh, and H. Garcia-Molina, "Semi structured Data: The TSIMMIS Experience," Proc. East-European Workshop Advances in Databases and Information Systems (ADBIS), pp. 1-8, 1997.

[11] C.-N. Hsu and M.-T. Dung, "Generating Finite-State Transducers for Semi-Structured Data Extraction from the Web," Information Systems, vol. 23, no. 8, pp. 521-538, 1998.

[12] N. Kushmerick, "Wrapper Induction: Efficiency and Expressiveness," Artificial Intelligence, vol. 118, nos. 1/2, pp. 15-68, 2000.

[13] A. Laender, B. Ribeiro-Neto, A. da Silva, and J. Teixeira, "A Brief Survey of Web Data Extraction Tools," SIGMOD Record, vol. 31, no. 2, pp. 84-93, 2002.

[14] B. Liu, R.L. Grossman, and Y. Zhai, "Mining Data Records in Web Pages," Proc. Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 601-606, 2003.

[15] W. Liu, X. Meng, and W. Meng, "Vision-Based Web Data Records Extraction," Proc. Int'l Workshop Web and Databases (WebDB '06), pp. 20-25, June 2006.

[16] Y. Lu, H. He, H. Zhao, W. Meng, and C.T. Yu, "Annotating Structured Data of the Deep Web," Proc. Int'l Conf. Data Eng. (ICDE), pp. 376-385, 2007.

[17] J. Madhavan, S.R. Jeffery, S. Cohen, X.L. Dong, D. Ko, C. Yu, and A. Halevy, "Web-Scale Data Integration: You Can Only Afford to Pay As You Go," Proc. Conf. Innovative Data Systems Research (CIDR), pp. 342-350, 2007.

[18] I. Muslea, S. Minton, and C.A. Knoblock, "Hierarchical Wrapper Induction for Semi-Structured Information Sources," Autonomous Agents and Multi-Agent Systems, vol. 4, nos. 1/2, pp. 93-114, 2001.

[19] Z. Nie, J.-R. Wen, and W.-Y. Ma, "Object-Level Vertical Search,".

[20] A. Sahuguet and F. Azavant, "Building Intelligent Web Applications Using Lightweight Wrappers," Data and Knowledge Eng., vol. 36, no. 3, pp. 283-316, 2001. Proc. Conf. Innovative Data Systems Research (CIDR), pp. 235-246, 2007.