International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.1, Jan-Feb 2012 pp-528-533 ISSN: 2249-6645

Extraction of Deep Web Contents

Sasikala.D[1], Selva Kumar.G[2]

* Final M.E Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore.
** Assistant Professor, Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore.

Abstract :

The World Wide Web is the emerging field available to users to access the contents of the web. This field is parted into two. They are surface web and deep web. The surface web refers to the static and is linked with other pages whereas deep web refers to the web page that is not indexed by the general search engine. The extraction of contents from web pages arise the problem of web – page – programming - language independent. This problem is raised because of the underlying complex structure of the web pages. To overcome this problem, visual features are used and to extract the contents of the deep web visual features are taken as the primary concern. While considering the visual feature as the primary concern, the dependent problems in web pages are eliminated. The extraction of deep web contents from the deep web pages involves both the data record extraction and data item extraction. To evaluate the performance of the extraction process, the method revision is used.

Keywords – Deep web, Visual Block Tree, Visual Features, Web Data Extraction, Web Mining, Wrapper Generation.

I. INTRODUCTION

With the emergence of World Wide Web, lot of information is available on-the fly as a result of query submitted. The web pages resulted is said to be the surface web, which is indexed by crawlers for the ease of users. The other part of the web is said to be the deep web. This deep web is lying for beyond the databases and these web pages are not indexed by the normal crawlers. Accessing these web pages is done by submitting queries to web data bases. Extracting the contents of these web pages is a critical problem. The web pages are designed using the Html. There are some limitations that are illustrated by the proposed solutions which are based on analyzing the html code of the web pages. First, the web pages are web-page-programming – language dependent. This is because the earlier approaches are not adapted to the present evolving versions of HTML. Second, they are incapable of handling the ever-increasing complexity of HTML source code of web pages. In order to make the web pages good in presentation, more and more presentation techniques are embedded in to the web pages. In earlier, these techniques are not considered as much important, they designed the web pages simple. But today the above presentation techniques are implemented. This makes the structure of the web page more complex. In previous works, many approaches are considered to overcome the above limitations. Those approaches are failed because they are failed to meet certain requirements. These approaches are described in the later sections. The purpose of extracting the contents of deep web is to present the visual approach which is web – page – programming – language independent. This proposed method is done by considering the visual cues along with some non-visual information. By considering the visual cues, the dependent problems solved. The extraction process is done by combining both the data record extraction as well as the data item extraction. This approach aims at automatically adapting the information extraction knowledge previously learned from a source web site to a new unseen site, at the same time, discovering previously unseen attributes. The four step strategy is employed for the extraction. They are given as:

(1) Sample deep Web page from a Web database is taken, its visual representation is obtained; transform it into a Visual Block tree.
(2) From the visual block tree the data records are extracted.
(3) Then, the data item separation and align the data items of same semantic together.
(4) Visual wrappers are generated for the resulted web database of the sample invisible web pages.

Thus, the extraction process is carried out efficiently. The evaluation measure revision is used to evaluate the performance of web data extraction. It is the percentage of the web databases whose data records or data items that cannot be perfectly extracted.

II. RELATED WORK

There are number of approaches presented for the extraction of contents from web pages. Those approaches are manual approach, semi-automatic approach and automatic approach. The detailed survey of these approaches is presented in [5] and [6].

2.1 Manual Approach

This is the earliest approach, helps the programmer to generate wrappers to identify the data fields and extract the data fields. This manual approach utilizes various tools. Some of the tools are Minerva [10], web – OQL [1], TSIMMIS [9].

2.1.1 Minerva

This tool uses the grammar in EBNF style, for each document, a set of productions is defined. This tool attempts to combine advantage of a declarative grammar based approach with features typical for procedural programming language by incorporating an explicit exception – handling mechanism inside the grammar.

2.1.2 Web – OQL

This tool is a declarative query language capable of locating selected pieces of data in the HTML pages. This tool originally aims at performing queries like SQL over the web.

2.1.3 TSIMMIS

This tool includes wrappers that can be configured through specification files written by the user. Specification files are composed by a sequence of commands that define extraction steps. An extractor based on the specification file parses an html page to locate the interesting data and extract them.

2.2 Semi-Automatic Approach

This approach uses the HTML – aware tools. The semi-automatic technique is broadly classified into text-based and sequence based technique. It rely on inherent structural features of HTML documents for accomplishing data extraction and grouping. The documents are turned to parsing tree before processing. Some representing tools of this approach are W4F [11], XWrap [8].

2.2.1 World Wide Web Wrapper Factory

This is a toolkit for the construction of wrappers. It is the java toolkit for building wrappers. The wrapper development process consists of three independent layers. They are: Retrieval layer, Extraction layer, and Mapping layer. This tool kit classifies the wrapper development process in three phases: first, the user describes how to access the document, second, he describes what pieces of data to extract, and third, he declares what target structure to use for storing the data extracted.

2.2.2 Xwrap

XWRAP is another important HTML –aware tool for semi automatic construction of wrappers. The tool features a component library that provides basic building blocks for wrappers, and a user friendly interface to ease the task of wrapper development. This tool classifies the wrapper generation process into two phases: structure analysis and source -specific xml generation.

2.3 Automatic Approach

The automatic approaches are primarily on text-based and tag-structured based approach. This approach uses tools that each tool will perform their functions separately. They do not combine their process to give whole result. Each process is independent of their functions. Though this approach is automatic, it has some limitations. The tools used by this approach are Depta [12], Roadrunner [3], IEPAD [13]. Some methods in [7] perform data record extraction not the data item extraction.

2.3.1 Depta

Data extraction based on partial tree alignment is another technique which extracts only HTML based web pages. This is an un-supervised tool. It can be only applicable to web pages that contain more than two data records in a data region. It is limited to handle nested data records. It conducts the process of mining from the single web page. The extraction process is at the record level.

2.3.2 Roadrunner

It is a tool that explores the inherent features of HTML documents to automatically generate wrappers. By comparing HTML structure of web pages of same "page class", generating a result of schema for the data contained in the pages. The unique feature of this tool is that no user intervention is requested.

2.3.3 IEPAD

This tool generalizes the extraction pattern from the unlabelled web pages. If a web page contains multiple homogenous data records to be extracted, they are rendered using the same template which provides good visualization. The center star algorithm is applied for the alignment of multiple strings.

III. ALGORITHM IMPLEMENTATION

In this section, the summarization of VIPS algorithm [2] is introduced. The Vision based Page Segmentation algorithm focuses primarily on layout features and is proposed to extract the content structure of the web page. The layout features used to partition the page at the semantic level. The vision-based content structure is deduced by combining the DOM Structure as well as the visual cues that are obtained from the web browser. The layout features used in this algorithm are listed in the visual features of the deep web pages.

The web page layout includes the location and size of the web page. The Fig 1 depicts the layout model of the sample web page.

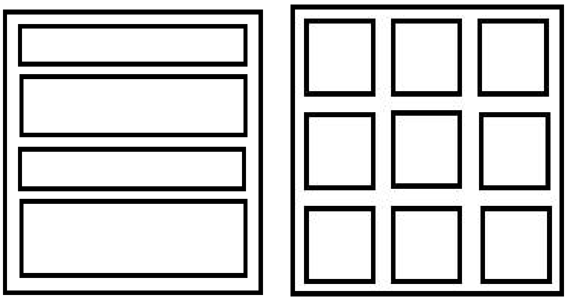


Fig 1 – layout model of data record of a sample deep web page

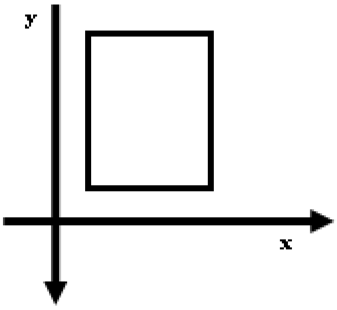


Fig 2 – web page layout

The Fig 2 shows how the web page is displayed on the coordinate system. The VIPS algorithm extracts the semantic structure of the web page. The semantic structure is considered as the hierarchical structure where each block represents as the node in visual block tree. The node of the visual block tree is assigned a degree of coherence value which represents how coherent content of that block. The degree of coherence has the following properties:

(1) The greater the DoC value, the more consistent the content within the block

(2) In the hierarchy tree, the DoC of the child is not smaller than of its parent.

The VIPS algorithm, take the sample web page as input and this input page is segmented into blocks along with visual cues. Once it is segmented, it extracts the blocks from the constructed html DOM tree, then the algorithm ties to find out the separators that are in the web page. The separators are the horizontal and vertical lines in the web page that distinguishes the contents and images clearly. When the separators are identified the semantic structure for the given page is constructed. The VIPS algorithm is very effective since the top-down approach is employed.

IV. PROPOSED APPROACH

In this section we briefly describe about the proposed methodology. The proposed methodology is based on visual perception for extracting the contents of the deep web pages. As the web page is displayed regularly in a two-dimensional media, it made users to browse the contents of the web page. A promising research direction is opened where the visual features are utilized to extract deep web data automatically. It also utilizes some non visual information. The non visual feature includes the same type of font, frequently occurring symbols and data types are also used. Since the web pages displayed consist most of text and images, web page layout and font are considered as visual information. The fonts are determined by its size, face, color, frame, etc., These visual features are important for identifying special information in the pages. To perform this, the features used are position, layout, appearance and content. The position features describes the location of the data region on deep web page. The layout features describes how the data records in the data region is typically arranged. The appearance features which captures the visual features with in data records. The content features indicate the regularity of the contents in the data records. The proposed system extracts both the structured as well as the unstructured pages. The major difference found between existing systems and proposed is that the current system is capable of extracting any web page programmed in any language which the existing system fails to do. The flow of vision based page segmentation algorithm is given in Fig 3.

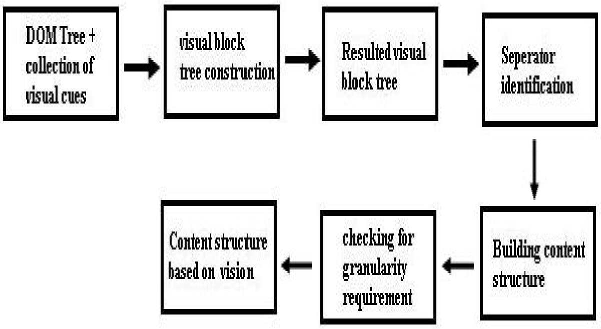

DOM Tree + collection of visual cues
visual block tree construction
Resulted visual block tree
Seperator identification
Building content structure
checking for granularity requirement
Content structure based on vision

Fig 3 Vision based Page Segmentation Algorithm

The main visual features that are to be considered before implanting the segmentation process is give n below:

(1) Position features – this feature describe the location of the data region in the web page. It has the following properties in order to locate the data region in the web page. The data regions are always placed in the centrally in horizontal position. The size of data region is always large when compared to the size of the whole web page.
(2) Layout features – this feature describes about the arrangement of data records in the data region. It also specifies some of the properties. They are: the data records are placed at the flush left of the data region. The data records are adjoined. The space between adjoins is the same and they will not overlap.
(3) Appearance features – these features specify the visual features that contained in the data record. Its properties are the data records are appear in similar. The data items having same semantic have similar presentations. The neighboring text data items use distinguishable fonts often.
(4) Content features – this feature intimate the uniformity of the contents that contained in the data record. It includes the following properties: the first data item in each record is must. The presentation of data items follows certain order. Some fixed static texts available in the web page are not generated by the web data bases.

4.1 Visual Block Tree

To transform the deep web page into a visual block tree, VIPS algorithm [2] is used. Visual block tree is resulted by the segmentation process. It is a segmentation of the web page. This tree contains the whole page as root block and the rectangular region represents each block of tree in the page. Leaf blocks cannot be segmented further which represents the semantic units.
The visual block tree has the following properties:

(1) Block *a* contains block *b* if *a* is ancestor of *b.*
(2) *a* and *b* do not overlap if they do not satisfy the above stated property.
(3) The blocks with the same parent are arranged in the tree according to the order of the corresponding nodes appearing on the page.

These properties are shown in Fig 4. In Fig 4 (a), b1, b2, b3 are the leaf blocks that contain their child blocks. The following figure represents the visual block tree for the given web page.

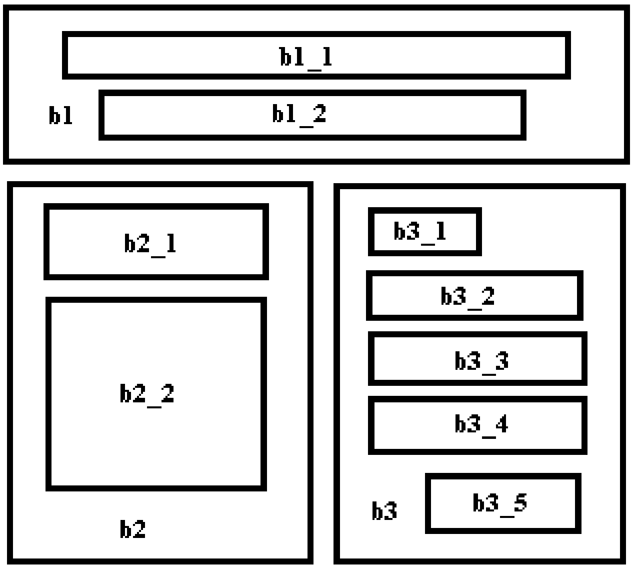


Fig 4 (a) – The presentation structure of the deep web page

By applying vision based page segmentation algorithm to a web page, the visual block tree is generated for that web page. The following Fig 5 shows the generated visual block tree for the given web page.

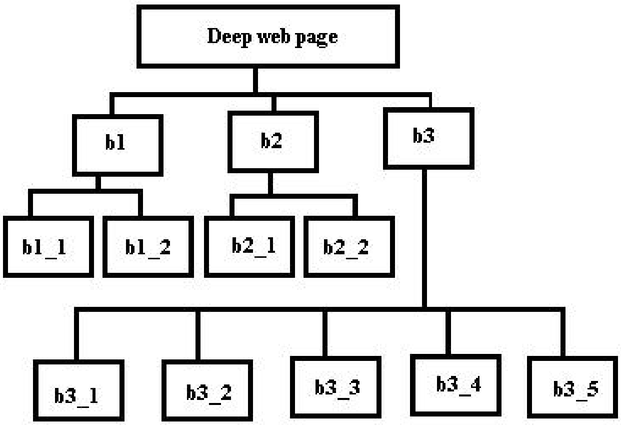
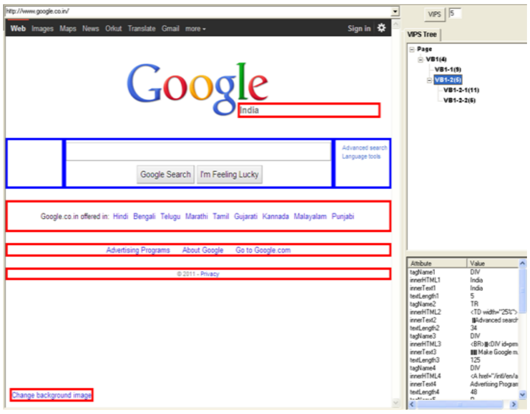


Fig 4 (b) – The visual block tree representation of the web page

Thus the figure show the web page gets segmented into blocks along with its visual block tree is shown aside. The blocks are highlighted for the selected leaf block in the visual block tree.



Attribute	Value
tagName1	DIV
innerHTML1	India
innerText1	India
textLength1	5
tagName2	TR
innerHTML2	<TD width="25%">
innerText2	Advanced search
textLength2	34
tagName3	DIV
innerHTML3	<BR><DIV id=prm
innerText3	Make Google m
textLength3	125
tagName4	DIV
innerHTML4	<A href="/intl/en/a
innerText4	Advertising Progran
textLength4	48

Fig 5- Visual block tree generation for a sample web page

4.2 Extraction of data records

The data record extraction aims to identify the boundary of the data records and extract them from the deep web pages. To extract data records from the visual block tree, location of data region if found and then data records are extracted from the region. The blocks in the visual block tree is the data region. The extraction of data records indicates that the position features are the primary content in the invisible web page. The location of the data record is identified by the block that satisfies the position features. To extract data records from data region accurately the facts must consider are: there may be blocks that do not belong to any data record and annotation about data records, and one data record may correspond to one or more blocks in the visual block tree, and the total number of blocks in which one data record contains is not fixed. The data records are regarded as the description of the corresponding object that consists of group of data items and some static template texts. The rectangular dashed lines in the Fig 6 represent the data record for a given deep web page. The data extraction process is carried out in three phases. They are: removal of noise blocks, clustering of blocks and regrouping of blocks.

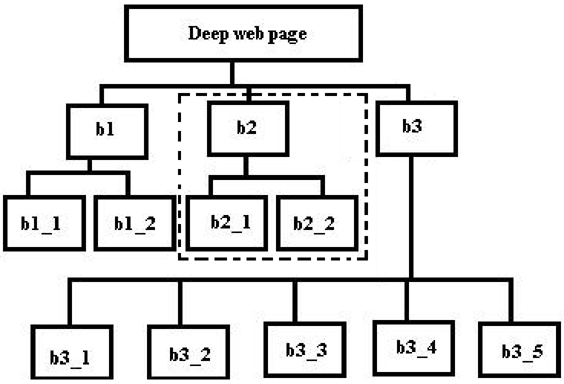


Fig 6 – Sample representation of data record

4.2.1 Removal of noise blocks

The noise blocks are blocks which do not contain any data records and annotation about data records. Usually these blocks are aligned at the top or bottom of the web page. This phase does not guarantee about the removal of all noise blocks.

4.2.2 Clustering of blocks

When the noise blocks are partially removed, the remaining blocks are grouped based on their appearance similarity. The weight of one type of content is proportional to their total size relative to the total size of the two blocks. This appearance similarity is calculated based on the following aspects: for images, the size is considered and for plain text and link text, the shared fonts are considered. Based on these aspects clustering process is done.

4.2.3 Regrouping of blocks

The blocks are regrouped such that the blocks of the same data record form a group. The first data item in each data record is mandatory. The regrouping process, involves the following three steps: it first rearranges the block in each cluster based on their appearance and the arrangement in the web page. Select the cluster with n blocks and these selected blocks used as seeds to form data records. Finally it determines which group they belong to.

4.3 Extraction of data items

The extraction of data item process focuses on the leaf nodes of the visual block tree. The three types of data items in the data record are: mandatory, optional and static data items. The mandatory data items are always appear in all data records. The optional data items may be missed in some data records. The static data items are the annotations to data. Fixed static texts refer the text appear in every data record. The position of data items in respective to their data record is classified as: absolute position and relative position. The absolute position says that the positions of the data item of certain semantics are fixed in the line they belonged. The relative position says that the position of the data item relative to the data record ahead of it.The extraction of data item process is carried out in two phases: segmentation of data record and aligning data item.

4.3.1 Segmentation of data record

The data record segmentation is carried out by collecting the leaf nodes in the data record of the visual block tree in left to right order. Leaf node also correspond each composite data item.

4.3.2 Aligning data item

Data item aligning focuses on how the data items of same semantic together are aligned and it should maintain the order of data items in the data record. This process is carried out by the following steps: first, all data items are not aligned. Second, data items are orderly aligned in data records. Third, optional data items which do not appear in some data records are encountered and those vacant spaces are filled with predefined blank item. This process involves the visual matching of data items. The absolute position mentioned here is the distance between the left side of the data item and the left side of the data region. This matching process considers both the absolute position as well as the relative position. If two data items do not have any absolute position, then they can be matched using their relative position. While matching process is done in relative position then the data item immediately before the two input data item is matched. For this matching process, the content feature’s and the appearance feature‘s properties are implemented.

4.4 Generation of visual wrappers

The visual wrappers are the set of extraction rules that are generated by using the extracted data record and the data item. These are programs which performs the data record and data item extraction with the set of parameters obtained from the sample web pages. The visual information is used to generate the visual wrappers.

V. Conclusion

In this paper, a novel vision based deep web data extraction method is introduced that consists of several distinct novel algorithms, which try to overcome inherent deficiencies, burdens and limitations. The users have a great opportunity to benefit from the flourish of the deep web. Normally the desired information in the deep web pages is embedded in the data records which are returned by the web databases as a response of user query. As our approach employs the extraction of structured data using visual features, this provides more efficiency. The primary steps in this approach: building visual block tree, extraction of data records and data items and the construction of visual wrappers are done by implementing the VIPS algorithm which primarily uses the visual features. The vision-based approach is intended to solve the HTML – dependent problem. In the earlier approach, the visual features are obtained by calling the Application Programming Interfaces of the Internet Explorer, this leads more time consuming. The new set of Application Programming Interfaces is developed to obtain visual features directly from the web pages. Thus, this methodology improves the optimization of search efficiently and more precise.

References

[1] G.O Arocena and A.O Mendelzon (1998), “webOQL: Restructuring Documents, Databases, and webs,” Proc. Int’l Conf. Data Eng.(ICDE), pp. 24-33.

[2] D.Cai, S. Yu, J. Wen, and Ma (2003), W. VIPS: A vision based page segmentation algorithm. Microsoft Technical Report MSR-TR-2003-79.

[3] 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.

[4] D. Cai, S. Yu, J. wen, and W. Ma (2003), “Extracting Content Structure for web Pages Based on Visual Representation,” Proc. Asia Pacific web Conf. (APweb), pp. 406-417.

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

[6] 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.

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

[8] L. Liu, C. Pu, and W. Han, "XWRAP: An XML-Enabled Wrapper Construction System for Web Information Sources," Proc. Int'l Conf. Data Eng. (ICDE), pp. 611-621, 2000.

[9] 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.

[10] V. Crescenzi and G. Mecca, "Grammars Have Exceptions," Information Systems, vol. 23, no. 8, pp. 539-565, 1998.

[11] A. Sahuguet and F. Azavant, "Building Intelligent Web Applications Using Lightweight Wrappers," Data and Knowledge Eng., vol. 36, no. 3, pp. 283-316, 2001.

[12] Y. Zhai and B. Liu, "Web Data Extraction Based on Partial Tree Alignment," Proc. Int'l World Wide Web Conf. (WWW), pp. 76-85, 2005.

[13] 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.

Authors

Miss. Sasikala.D received the B.Tech degree in IT from Anna University, Chennai and currently pursuing M.E degree in Computer Science and Engineering in Sri Shakthi Institute of Engineering and Technology, Coimbatore. Her research interest includes Computer Networks, Data Mining and Web Mining.

Mr. Selvakumar.G, Assistant professor in the Department of CSE in Sri Shakthi Institute of Engineering and Technology, Coimbatore, has completed M.E in Anna university. He has 5 years of experience in software development, consultancy and software project management especially in the area of capital markets. He has 3 years of teaching and research experience. His area of interest is web technologies.