Web Content Mining Based on Dom Intersection and Visual Features Concept

Shaikh Phiroj Chhaware[1], Dr.Mohammad Atique[2], Dr. Latesh. G. Malik[3]
[1] esearch Scholar, G.H. Raisoni College of Engineering, Nagpur 440019 (MS),
[2] Associate Professor, Dept. of Computer Science & Engineering,
S.G.B. Amravati University, Amravati (MS)
[3] Professor, Department of Computer Science & Engineering G.H. Raisoni College of Engineering,
Nagpur 440019 (MS)

ABSTRACT

Structured Data extraction from deep Web pages is a challenging task due to the underlying complex structures of such pages. Also website developer generally follows different web page design technique. Data extraction from webpage is highly useful to build our own database from number applications. A large number of techniques have been proposed to address this problem, but all of them have inherent limitations because they present different limitations and constraints for extracting data from such webpages. This paper presents two different approaches to get structured data extraction. The first approach is non-generic solution which is based on template detection using intersection of Document Object Model Tree of various webpages from the same website. This approach is giving better result in terms of efficiency and accurately locating the main data at the particular webpage. The second approach is based on partial tree alignment mechanism based on using important visual features such as length, size, and position of web table available on the webpages. This approach is a generic solution as it does not depend on one particular website and its webpage template. It is perfectly locating the multiple data regions, data records and data items within a given web page. We have compared our work's result with existing mechanism and found our result much better for number webpage.

Keywords: *Document Object Model, Web Data Extraction, Visual Features, Template Detection, Webpage Intersection, Data Regions, Data Records.*

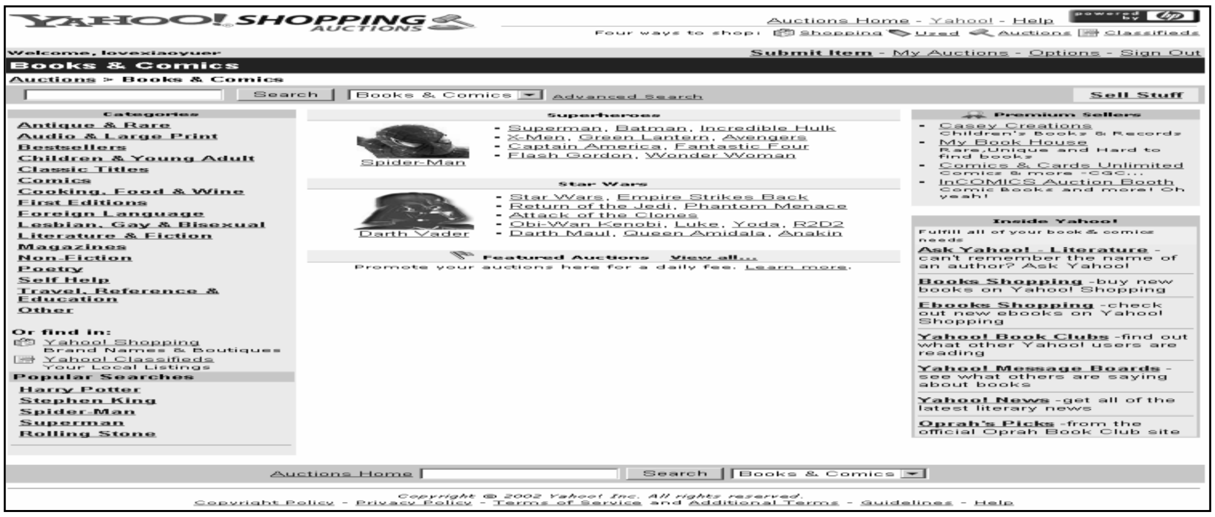
I. Introduction

The web contains the large amount of structured data and served as a good interface for databases over the Internet. A large amount of web content is generated from web databases in response to the user queries. Often the retrieved information which is a user query results are enwrapped in a dynamically generated web page in the form of data records. Such special web pages are called as deep web page and online databases are treated as deep web databases. Various web databases have reached 25 million according to a recent survey [1].

The data records and data items which are available on these deep web pages need to be convert in machine processable form. This machine processable data is required in many post data processing applications such as opinion mining, deep web crawling, meta-searching. And hence the structured data needs to be extracted efficiently considering the enormous amount data available on internet which is very rapidly growing day by day. Templates of the deep web pages are generally similar and spread over the other web pages of the same website [2].

Many works can be found in the literature for deep web structured data extraction and mostly they are relying on the web page segmentation either visual clues segmentation or DOM tree based web page segmentation [3]. But in all such techniques it is observed that they require much machine time to process the single web page as it is web browser dependent for gathering necessary visual information for the various objects displayed on the web page [4]. World Wide Web has millions searchable information sources. The retrieved information 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.

Figure 1 shows such deep web pages from www.yahoo.com. In this example figure we can see that the actual interesting information contained within the page occupies only 1/3 of the entire web page while other information which are generally treated as unwanted information or simple "noise" information occupies rest of the web page i.e about 2/3 of web page space. This noise information generally possesses advertisement of product, navigation link, hyperlink, contact information, web page menu bar and items, links to other web pages of the web sites etc. These types of noisy information are very harmful for the data extraction process. They generally hamper the efficiency of web data extraction algorithm employed because


YAHOO! SHOPPING AUCTIONS
Auctions Home - Yahoo! - Help
Four ways to shop: Shopping Used Auctions Classifieds
Welcome, lovexiaoyuer
Submit Item - My Auctions - Options - Sign Out
Books & Comics
Auctions > Books & Comics
Search | Books & Comics | Advanced Search | Sell Stuff

Categories
Antique & Rare
Audio & Large Print
Bestsellers
Children & Young Adult
Classic Titles
Comics
Cooking, Food & Wine
First Editions
Foreign Language
Lesbian, Gay & Bisexual
Literature & Fiction
Magazines
Non-Fiction
Poetry
Self Help
Travel, Reference & Education
Other

Or find in:
Yahoo! Shopping
Brand Names & Boutiques
Yahoo! Classifieds
Your Local Listings

Popular Searches
Harry Potter
Stephen King
Spider-Man
Superman
Rolling Stone

Superheroes
Spider-Man
- Superman, Batman, Incredible Hulk
- X-Men, Green Lantern, Avengers
- Captain America, Fantastic Four
- Flash Gordon, Wonder Woman

Star Wars
Darth Vader
- Star Wars, Empire Strikes Back
- Return of the Jedi, Phantom Menace
- Attack of the Clones
- Obi-Wan Kenobi, Luke, Yoda, R2D2
- Darth Maul, Queen Amidala, Anakin

Featured Auctions View all...
Promote your auctions here for a daily fee. Learn more.

Premium Sellers
- Casey Creations
Children's Books & Records
- My Book House
Rare,Unique and Hard to find books
- Comics & Cards Unlimited
Comics & more -CGC...
- InCOMICS Auction Booth
Comic Books and more! Oh yeah!

Inside Yahoo!
Fulfill all of your book & comics needs
Ask Yahoo! - Literature - can't remember the name of an author? Ask Yahoo!
Books Shopping -buy new books on Yahoo! Shopping
Ebooks Shopping -check out new ebooks on Yahoo! Shopping
Yahoo! Book Clubs -find out what other Yahoo! users are reading
Yahoo! Message Boards - see what others are saying about books
Yahoo! News -get all of the latest literary news
Oprah's Picks -from the official Oprah Book Club site

Auctions Home | Search | Books & Comics
Copyright © 2002 Yahoo! Inc. All rights reserved.
Copyright Policy - Privacy Policy - Terms of Service and Additional Terms - Guidelines - Help

Figure 1. An example deep Web page from Yahoo.com

algorithm needs to be processed this information also. So actual efficiency of algorithm cannot be achieved and if we are trying to process very large numbers of web pages for data extraction, noisy information consume too much of the processing time.

This webpage contains information regarding books which are presented in the form of data records and each data record has some data items such as author, title etc. To make the data records and data items in them machine processable, which is needed in many applications such as deep Web crawling and meta searching, the structured data need to be extracted from the deep Web pages. Extraction of web data is a critical task this data can be very useful in various scientific tools and in a wide range of application domains.

II. Literature Review

Many works has been carried out for the data extraction from the deep web pages. The basis for categorizing them are generally based how much amount of human efforts and interference required for data extraction process. These categories are as follows:

This paper present the actual work carried out to target the problem of structured data extraction from the deep web pages. This paper is organized as follows. Part II contains the literature review of existing system and methodologies used to solve this problem. Part III focuses first approach that we have used that is based on webpage template detection mechanism based on DOM intersection method. Part IV gives the details of second approach which is generic mechanism to carry out the task of structured data extraction based on DOM tree parsing and using visual features to locate the main data on the page. The part V gives the experimental setup for the methods, result obtained and comparisons of result with existing system. Part VI presents conclusions and future works.

A. Supervised Approach

The idea present in paper [5] is one such approach where a manual intervention is required during data extraction from deep web pages. Other system is MINERVA which is again utilizes the same mechanism for this problem

B. Semi-Supervised Approach

Paper [6] and paper [7] has built up the system for data extraction and they are employing the semi-supervised approach for it. At some of data extraction process the user intervention is required specially during actual data population to the local database while rest of the part extraction process is automatic.

C. Unsupervised Approach

This is a novel approach for web data extraction. Now a day all the research focus in this area is diverted on this kind of mechanism. [1] gives such a kind of idea which is based on visual features. But it also has severe limitation as it does not mine the data from multiple data regions. Also the process given for the data extraction is very much complex as multiple visual are needs to be examined.

III. Data Extraction Based On Dom Intersection Method

This section presents a novel approach for website template detection based on intersection of document object model (DOM) tree of a test page with the another web page from the same website. The result of this mechanism is stored in a table which has common nodes in the DOM tree. Then for the next web page processing, the node information from this table is applied on the incoming web page's DOM tree instead of DOM tree processing for that web page. The table information then can be very much useful for forming the rules for automatic data extraction I,e for automatic wrapper generation method.

A. The Proposed Framework

Our website template detection method is based on the webpage DOM tree intersection concept. Here we consider the most common intersection node information of the two web pages from the same website.

Whenever any new web page is available, it passes though the four steps:

1) DOM Tree building
2) Template Detection
3) Template Intersection
4) Wrapper Table Generation.

For the next web page from the same website only steps 1 and 3 are required. Wrapper table will get updated automatically whenever there is a change in DOM tree of the both web pages. Here the changes in the DOM tree of first web pages is learn. The step 4 is essential for wrapper generation.

B. Document Object Model Tree Building

This is the first step towards actual data extraction. Here we build a DOM tree or tag tree of a HTML web page.

- Most HTML tags works in pairs. The nesting of the tag-pair can be possible to have the parent child relationship among other nodes of the tree.
- The DOM tree is build up from the HTML code of a web page.

The DOM tree node presents the pair of tags and each pair of tags can be nested within which children node may be available.

C. Template Detection

The web page templates are just fragments of the HTML code included in the collection of HTML documents. Automatic generation of templates is generally done and they are replicated to other web page development. This ensures uniformity and speedy development of the web site. For this purpose, we are using simple tree matching algorithm which is given below:

Algorithm: SimpleTreeMatching(A, B)

```
   Input: DOM Tree A and B
1.  If the roots of the two trees A and B contain distinct symbols then return 0.
2.  Else m = the number of first-level subtrees of A;
         n = the number of first-level subtrees of B;
    Initialization          M[i,0] = 0 for i = 0, . . ., m;
                            M[0,j] = 0 for j = 0, . . ., n;
3.  For i = 1 to m do
4.      For j = 1 to n do
5.          M[i,j] = max(M(i,j-1),M(i-1, j),M(i-1,j-1)+W(i,j)
                  where W(i,j) = SimpleTreeMatching($A_i$,$B_j$)
6.  Return (M(m,n)+1)
```

The working of the Simple Tree matching algorithm is illustrated by means the figure 2. It takes two labels ordered rooted tree and map the nodes in each tree. At a glance, a generalized mapping can be possible between same nodes among the two different trees. Replacements of nodes are also allowed, e.g., node C in tree A and node G in tree B. Simple Tree Matching algorithm is top-down algorithm and it evaluated the similarity between two trees by producing the maximum matching.

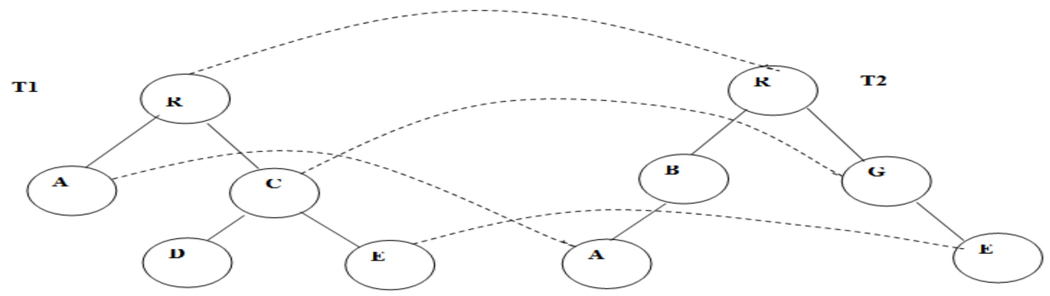


Figure 2. Matching between two labels ordered rooted trees

D. Template Intersection

The template intersection phase is applied when there is new DOM tree appears for the processing. This phase simply eliminates non-matching nodes of first tree with another one and hence here we get the appropriate template of that deep web page. This allows the extraction of main content from the deep web page at later stage.

E. Wrapper Table Generation

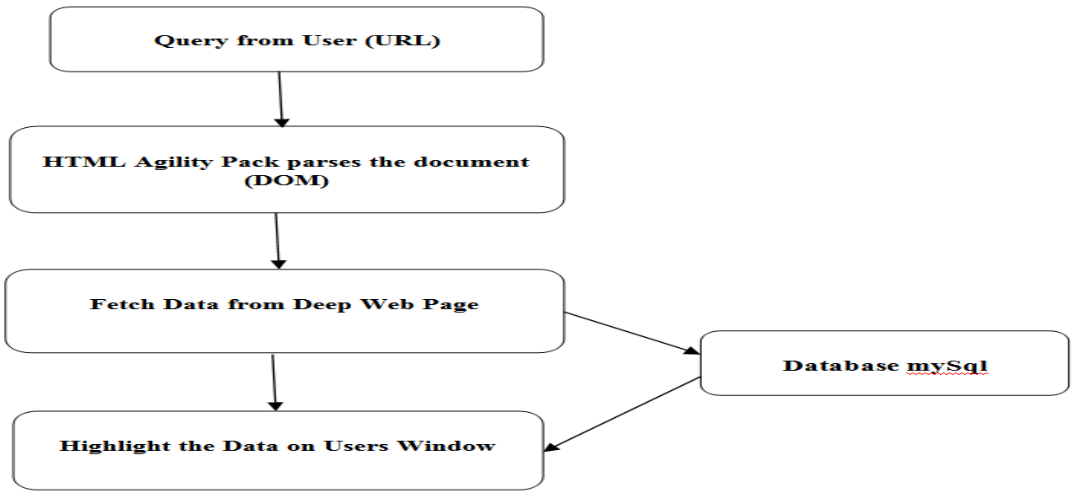
The wrapper table gives the information about which nodes are common in both trees. The first column presents the tree, e.g., T1 and T2 and further columns presents the node tag. As shown in figure 1, tree T1 has nodes R, A, C, D, and E. So in the wrapper table numeric number 1 will be marked in T1 row which presents that these nodes are the part of tree T1. Such kind of processing is done for all appearing tree. The wrapper table is very for automatic data extraction from other pages of the website. We can form wrapper rules from this table.

IV. Data Extraction Based On Dom Tree Parsing & Visual Features

This is our second approach which is a generic solution for extracting main structured data from deep webpages. This uses DOM Tree parsing process and some visual features of the table such area, size and location.

This approach uses HTML Agility Pack parser to process the DOM tree. This parser parses the document in the form of DOM tree. DOM tree is used to traverse the web document by using XPath method this method is traverse the web document in two ways i.e. from root node to leaf node and vice versa. The following figure 3 shows the proposed system. This shows how to extract the main data from deep web pages and how we take query from user and what is the use of HTML Agility Pack for parsing the document. It shows the extracted data is stored in database and it displays the extracted data on the user’s window.

As we know the web page is consists of group of blocks means each block has specific height and specific width. To get the main data which is available in the page we have find out area of each block which web page has and the find the largest block from the page that has the main data which we required to extract. Here we used algorithm to calculate the largest block. This algorithm is applied on the DOM tree and it searches for the largest table on the web page by comparing all the tables of the same web page. This is iterative procedure and once the largest table is found, it is send to the local database.


Query from User (URL)
HTML Agility Pack parses the document (DOM)
Fetch Data from Deep Web Page
Database mySql
Highlight the Data on Users Window
Figure 3. Proposed System based on second approach

Algorithm: Find the largest block of web page

```
1. Get the width of each block
2. Get the height of the each block
3. Calculate the Area of each block
4. MAXArea = 0
5. Area = Height * Width
6. Repeat step 6 compare  MAXArea< Area
7. then MAxArea= Area
8. goto step 5
9. Displays the contains of MAXArea
   End.
```

The Deep web is normally defined as the content on the Web not accessible through a search on general search engines. This content is sometimes also considered to as the hidden or invisible web to extract. The Web is a complex that contains information from a variety of sources and includes an evolving mix of different file types and media files. It is much more than static, self-contained Web pages. Above algorithm we calculate the area for each execution and find the largest block from the web document. Suppose we execute this query for static web page sometimes it does not work because static page might have or not block wise allocation of the web page but it works for dynamic web page properly means for deep web page.

V. Experimental Setup and Result Comparisons

For the experimentation we have gathered the input web pages from various commercial websites such as www.dell.co.in, www.amozon.in, www.shopclues.com, www.snapdeal.com, www.yahoo.com, www.rediff.com etc.

Our first approach based on the template detection of DOM intersection of various web pages from same website, for this we have used multiple webpages of the same website and try to extract the data from it.

The second approach is having better result as it present the generic solution. The detail experimental setup of this approach is given as below.

A. System Requirement

The experimental results of our proposed method for vision-based deep web data extraction for deep web document are presented in this section. The proposed approach has been implemented in VB.NET. It usually ships in two types, either by itself or as part of Microsoft Visual Studio .NET. To use the lessons on this site, you must have installed either Microsoft Visual Basic .NET 2003 or Microsoft Visual Studio .NET 2003. All instructions on this site will be based on an installation of Microsoft Visual Studio .NET. From now on, unless specified otherwise, we will use the expressions "Microsoft Visual Basic" or "Visual Basic" to refer to Microsoft Visual Basic .NET 2003. Memory requirement is minimum 1GB. If we want to refer to another version, we will state it. WAMP Server for the execution of MY-SQL, HTML AGILITY PACK. It is an open source technology so it easily available and free of cost. It is used for parsing web document in DOM. The Html Agility Pack is a free, open-source library that parses an HTML document and constructs a Document Object Model (DOM) that can be traversed manually or by using XPath expressions. (To use the Html Agility Pack you must be using ASP.NET version 3.5 or later.) In a nutshell, the Html Agility Pack makes it easy to examine an HTML document for particular content, and to extract or modify that markup. Wamp Server is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. These pages are used in proposed method for removing the different noise of deep web page. The removal of noise blocks and extracting of useful content chunks are explained in this section.

B. User Window

In this system use plays very important because user request for data extraction and the data is available anywhere in the centralized database. Then user enters URL and then presses the button extract clues, and then the data is extracted as per requirement of user. For process execution following procedure is followed. Initially user opens browser for data extraction. Figure 4 shows the browser for data extraction. It has two parts first part contains input means normal web page which is available anywhere in World Wide Web and second part is output window which contains the extracted data on the basis of largest area of web page block and hide the remaining part of the web page.

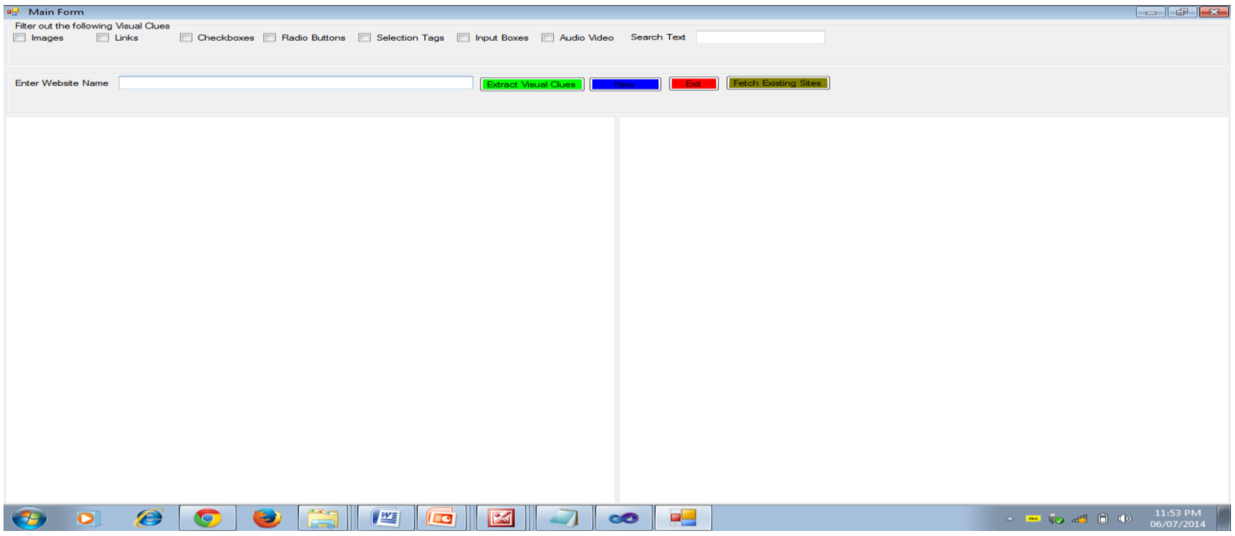


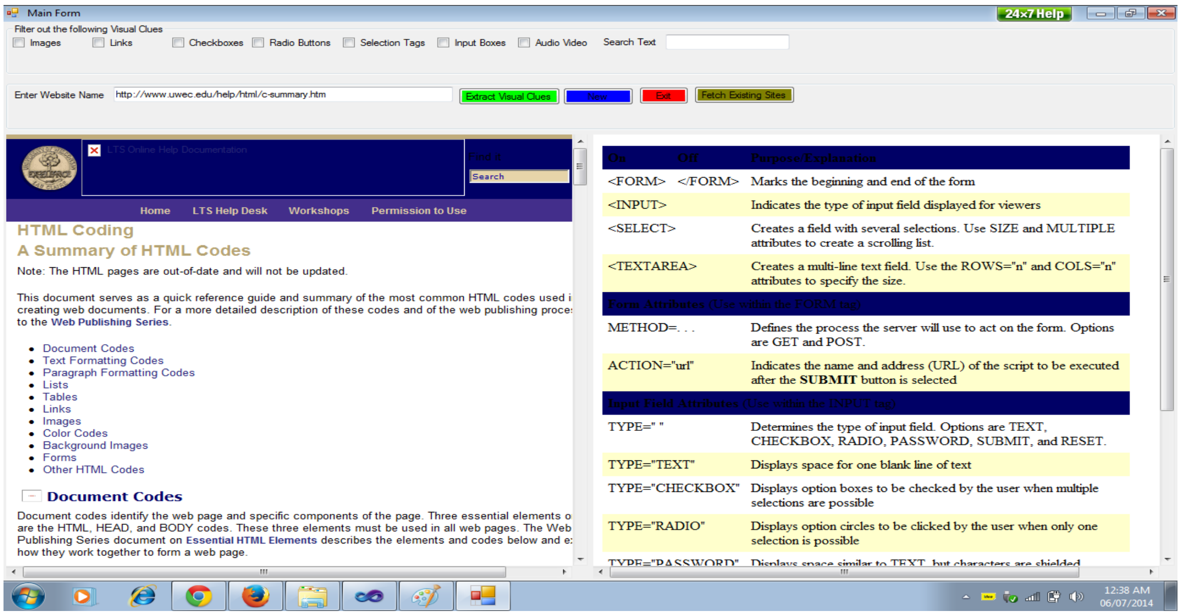
Figure 4. User Window

C. Extracted Data Window

The figure 5 shows the execution of the process in this image user enters the URL and asks for data extraction.

D. Data Fetch from Database

Sometimes user wants to visit that data which is already extracted by someone else that data is available in database (MySql). For every execution the extracted data is stored in database in tabular format. Here we use the technique to extract the largest block (area) of the web page means to calculating the area of every block of the web page. That extracted block has specific height and width with help of this parameter we calculate the area of every block. Following window shows that how user access the data from database. Here user simply clicks on the Fetch Data button and then the list will be opened user needs to select the required site and then click on the OK button. Then the required dataa is displayed on the output window. Figure 6 shows the interface for the data from database.



Enter Website Name http://www.uwec.edu/help/html/c-summary.htm

HTML Coding

A Summary of HTML Codes

Note: The HTML pages are out-of-date and will not be updated.

This document serves as a quick reference guide and summary of the most common HTML codes used i
creating web documents. For a more detailed description of these codes and of the web publishing proce
to the **Web Publishing Series**.

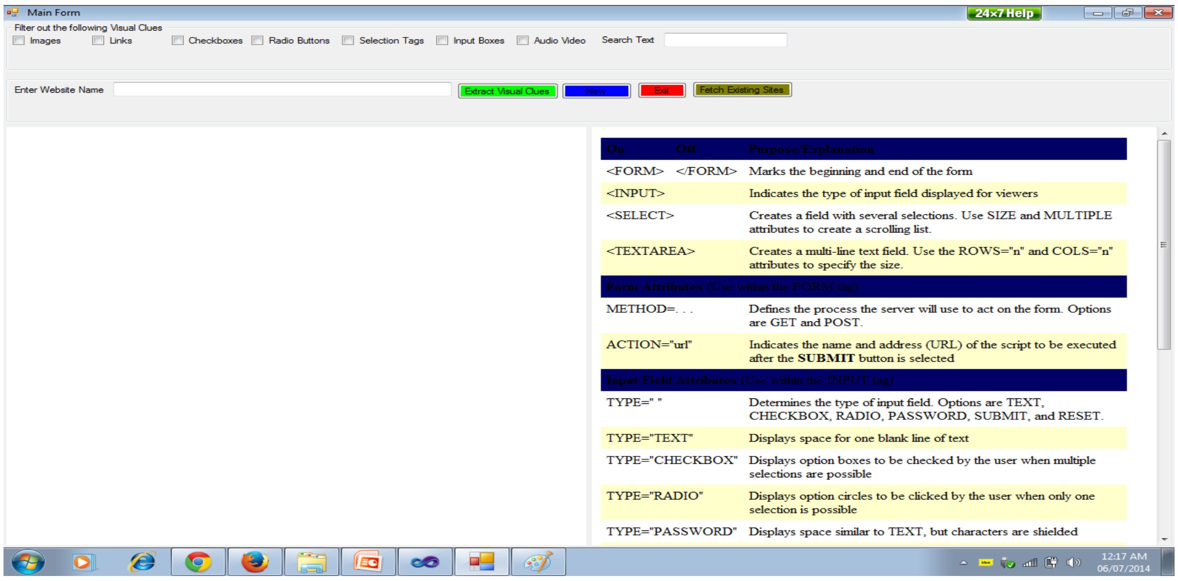
- Document Codes
- Text Formatting Codes
- Paragraph Formatting Codes
- Lists
- Tables
- Links
- Images
- Color Codes
- Background Images
- Forms
- Other HTML Codes

Document Codes

Document codes identify the web page and specific components of the page. Three essential elements o
are the HTML, HEAD, and BODY codes. These three elements must be used in all web pages. The Web
Publishing Series document on **Essential HTML Elements** describes the elements and codes below and e
how they work together to form a web page.

On	Off	Purpose/Explanation
<FORM>	</FORM>	Marks the beginning and end of the form
<INPUT>		Indicates the type of input field displayed for viewers
<SELECT>		Creates a field with several selections. Use SIZE and MULTIPLE attributes to create a scrolling list.
<TEXTAREA>		Creates a multi-line text field. Use the ROWS="n" and COLS="n" attributes to specify the size.
Form Attributes (Use within the FORM tag)		
METHOD=. . .		Defines the process the server will use to act on the form. Options are GET and POST.
ACTION="url"		Indicates the name and address (URL) of the script to be executed after the **SUBMIT** button is selected
Input Field Attributes (Use within the INPUT tag)		
TYPE=" "		Determines the type of input field. Options are TEXT, CHECKBOX, RADIO, PASSWORD, SUBMIT, and RESET.
TYPE="TEXT"		Displays space for one blank line of text
TYPE="CHECKBOX"		Displays option boxes to be checked by the user when multiple selections are possible
TYPE="RADIO"		Displays option circles to be clicked by the user when only one selection is possible
TYPE="PASSWORD"		Displays space similar to TEXT, but characters are shielded

Figure 5. Extracted Data Window



On	Off	Purpose/Explanation
<FORM>	</FORM>	Marks the beginning and end of the form
<INPUT>		Indicates the type of input field displayed for viewers
<SELECT>		Creates a field with several selections. Use SIZE and MULTIPLE attributes to create a scrolling list.
<TEXTAREA>		Creates a multi-line text field. Use the ROWS="n" and COLS="n" attributes to specify the size.
Form Attributes (Use within the FORM tag)		
METHOD=. . .		Defines the process the server will use to act on the form. Options are GET and POST.
ACTION="url"		Indicates the name and address (URL) of the script to be executed after the **SUBMIT** button is selected
Input Field Attributes (Use within the INPUT tag)		
TYPE=" "		Determines the type of input field. Options are TEXT, CHECKBOX, RADIO, PASSWORD, SUBMIT, and RESET.
TYPE="TEXT"		Displays space for one blank line of text
TYPE="CHECKBOX"		Displays option boxes to be checked by the user when multiple selections are possible
TYPE="RADIO"		Displays option circles to be clicked by the user when only one selection is possible
TYPE="PASSWORD"		Displays space similar to TEXT, but characters are shielded

Figure 6. Fetching Data from Database

E. Database

For every execution the extracted data is stored in the database in the form of tables for future use. Suppose user wants to fetch the data from database which is available in the database then user just click on the fetch existing site button then data automatically fetched from the database as per requirement. The figure 7 shows the sample of database where our data is stored. In our approach we have extracted some deep web page data, extracted web page URL is shown inn following table. Table contains ID and URL here ID is allotted for every extracted web document.

F. Result Evaluation

Our main approach is to extract the data from deep web pages and remove all unwanted noise from the web page then find out the result in the form of extracted data and it is to be calculated by using following expressions.

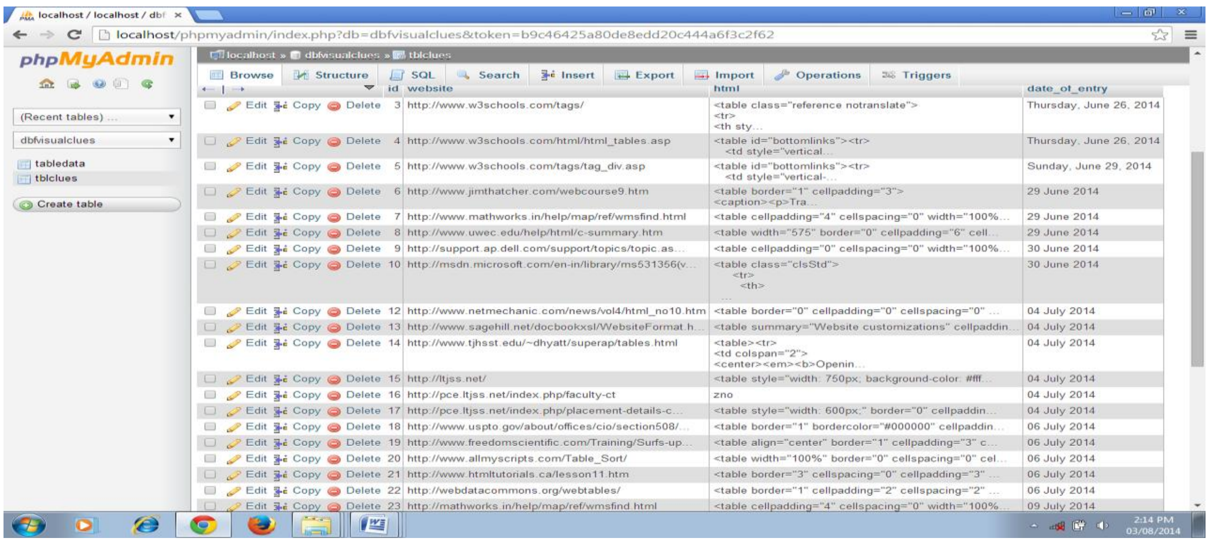
Precision is the percentage of the relevant data records identified from the web page.

Precision = DRc / DRe

Recall defines the correctness of the data records identified.

Recall = DRc / DRr

Where, DRc is the total number of successfully extracted data records sample web page is subjected to the proposed approach to identify the relevant web data region. Data region having description of some products is extracted by our data extraction method after removing the noises. The filtered data region, DRe is the total number of data records on the page. DRr is the total number of data records extracted.


localhost / localhost / dbf
localhost/phpmyadmin/index.php?db=dbfvisualclues&token=b9c46425a80de8edd20c444a6f3c2f62
phpMyAdmin
(Recent tables) ...
dbfvisualclues
tabledata
tblclues
Create table
localhost » dbfvisualclues » tblclues
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Triggers
2:14 PM
03/08/2014

	id	website	html	date_of_entry
Edit Copy Delete	3	http://www.w3schools.com/tags/	<table class="reference notranslate"> <tr> <th sty...	Thursday, June 26, 2014
Edit Copy Delete	4	http://www.w3schools.com/html/html_tables.asp	<table id="bottomlinks"><tr> <td style="vertical...	Thursday, June 26, 2014
Edit Copy Delete	5	http://www.w3schools.com/tags/tag_div.asp	<table id="bottomlinks"><tr> <td style="vertical-...	Sunday, June 29, 2014
Edit Copy Delete	6	http://www.jimthatcher.com/webcourse9.htm	<table border="1" cellpadding="3"> <caption><p>Tra...	29 June 2014
Edit Copy Delete	7	http://www.mathworks.in/help/map/ref/wmsfind.html	<table cellpadding="4" cellspacing="0" width="100%...	29 June 2014
Edit Copy Delete	8	http://www.uwec.edu/help/html/c-summary.htm	<table width="575" border="0" cellpadding="6" cell...	29 June 2014
Edit Copy Delete	9	http://support.ap.dell.com/support/topics/topic.as...	<table cellpadding="0" cellspacing="0" width="100%...	30 June 2014
Edit Copy Delete	10	http://msdn.microsoft.com/en-in/library/ms531356(v...	<table class="clsStd"> <tr> <th> ...	30 June 2014
Edit Copy Delete	12	http://www.netmechanic.com/news/vol4/html_no10.htm	<table border="0" cellpadding="0" cellspacing="0" ...	04 July 2014
Edit Copy Delete	13	http://www.sagehill.net/docbookxsl/WebsiteFormat.h...	<table summary="Website customizations" cellpaddin...	04 July 2014
Edit Copy Delete	14	http://www.tjhsst.edu/~dhyatt/superap/tables.html	<table><tr> <td colspan="2"> <center><em><b>Openin...	04 July 2014
Edit Copy Delete	15	http://ltjss.net/	<table style="width: 750px; background-color: #fff...	04 July 2014
Edit Copy Delete	16	http://pce.ltjss.net/index.php/faculty-ct	zno	04 July 2014
Edit Copy Delete	17	http://pce.ltjss.net/index.php/placement-details-c...	<table style="width: 600px;" border="0" cellpaddin...	04 July 2014
Edit Copy Delete	18	http://www.uspto.gov/about/offices/cio/section508/...	<table border="1" bordercolor="#000000" cellpaddin...	06 July 2014
Edit Copy Delete	19	http://www.freedomscientific.com/Training/Surfs-up...	<table align="center" border="1" cellpadding="3" c...	06 July 2014
Edit Copy Delete	20	http://www.allmyscripts.com/Table_Sort/	<table width="100%" border="0" cellspacing="0" cel...	06 July 2014
Edit Copy Delete	21	http://www.htmltutorials.ca/lesson11.htm	<table border="3" cellspacing="0" cellpadding="3" ...	06 July 2014
Edit Copy Delete	22	http://webdatacommons.org/webtables/	<table border="1" cellpadding="2" cellspacing="2" ...	06 July 2014
Edit Copy Delete	23	http://mathworks.in/help/map/ref/wmsfind.html	<table cellpadding="4" cellspacing="0" width="100%...	09 July 2014

Figure 7. Database structure

Table I presents the overall result of the above two system presented in this paper and their performances are compared against the existing systems like ViDe and MDR against the parameters like recall and precision.

Table I

Result Evaluation and Comparisons

Approach	Total No. webpages	Recall	Precision
First Approach	78	96.7%	98.8%
Second Approach	121	92.8%	91.5%
ViDE	133	95.6%	93.4%
MDR	118	82.3%	78.6%

VI. Conclusion and Future Works

The desired information is embedded in the deep Web pages in the form of data records returned by Web databases when they respond to users' queries. In this paper, we have given two different approaches for the problem main data extraction from the deep web pages in structured form. The first approach is presenting a non-generic solution and is totally based on the specific website's webpage template. The second approach is generic and uses minimal number of visual features and hence the efficiency of the system enhances. Also this system is able to mine the data from the multiple data regions because it is mining the data in structured format.

There are numbers of ways available where we can improve the performance of the systems and eliminate the constraints that we have considered. There are several issues which yet to be address like in perfect data mapping with local database attributes and web data table attributes. This work eventually may lead to the use of natural language processing and text mining approach at advance level.

References

[1] Wei Liu, Xiaofeng Meng, Weiyi Meng,"ViDE: A Vision-Based Approach for Deep Web Data Extraction", *IEEE Transactions on Knowledge and Data Engineering*, vol.22, no.3, pp.447-460, 2010.

[2] Yossef, Z. B. and Rajgopalan, S., "Template Detection via Data Mining and its Applications", *Proceedings of the 11th international conference on World Wide Web*, pp. 580-591, 2002

[3] Ma, L., Goharian, N., Chowdhury, A., and Chung M., "Extracting Unstructured Data from Template Generated Web Document", *Proceedings of the 12th international conference on Information and /knowledge Management*, pp. 512-515, 2003.

[4] Chakrabarti, D., Kumar, R., and Punera, K., "Page Level Template Detection via Isotonic Smoothing", *Proceedings of the 16th international conference on World Wide Web*, pp. 61-70, 2007.

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

[6] 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.

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

[8] Yi, L., Liu, B., and Li, X., "Eliminating Noisy Information in Web Pages for Data Mining.", *Proceedings of the 9th ACM SIGKDDInternational Conference on Knowledge Discovery and Data Mining*, pp. 296-305, 2003

[9] 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.

[10] Ashraf, F.; Ozyer, T.; Alhajj, R., "Employing Clustering Techniques for Automatic Information Extraction from HTML Documents", *IEEE* Transactions on Systems, Man, and Cybernetics, Part C: Applications *and reviews*, vol.38, no.5, pp.660-673, 2008.

[11] Sandip Debnath,Prasenjit Mitra,C. Lee Giles, "Automatic Extraction of Informative Blocks from WebPages", *In Proceedings of the ACM* symposium on Applied computing, Santa Fe, New Mexico, pp. 1722 – 1726,2005.

[12] 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 ou Go,", Proc. Conf. Innovative Data Systems Research (CIDR), pp. 342-350, 2007.