Data Extraction and Annotation for Web Databases using Multiple Annotators Approach- A Review

Yogesh W. Wanjari
Dept. of CS&IT,
Dr. B. A. M. University,
Aurangabad-431004
India

Dipali B. Gaikwad
Dept. of CS&IT,
Dr. B. A. M. University,
Aurangabad-431004
India

Vivek D. Mohod
Dept. of IT
SGGSIE & T
Nanded-431606
India

Sachin N. Deshmukh
Dept. of CS&IT,
Dr. B. A. M. University,
Aurangabad-431004
India

ABSTRACT
Web contain huge amount of information on Web sites the user can retrieve this with help of the search input query to Web databases & fetch the relevant information. Perhaps Web databases return the multiple search output records dynamically on Web browser, these search record are containing the Deep Web pages in the form of HTML pages. It is time consuming &human efforts are involved. The traditional search engine does not index the hidden Web pages from Web databases, such as (Google, Yahoo etc.). Many existing proposed techniques have addressed the problem of how to extract efficient structure data from Deep Web. The deep web refers to the hidden database used by web sites. But the information extraction & annotation is key challenge in web mining. The information retrieval should be done automatically & arrange in a systematic way for further processing. Various methodologies like wrapper induction is been induced. The labeling is done to the extracted information as per the concept.Various types of annotators are used on the basis of the data to be annotated. In this paper survey the automatic annotation approach on the basis of different feature of text node and data units.

General Terms
Data extraction, Web data annotation, Deep web pages and Wrapper induction

Keywords
Data Extraction, Data annotation, Annotators, Text nodes, Data Units and Wrapper

1. INTRODUCTION
Now a day's web technology is getting an emergence importance in day to day life! Everyone is familiar with surfing the web, uploading personal or important data on the web, sharing data with friends or social communities like the Facebook. Even mobile technology focus on the various trends in web. There are various technologies & researches are focusing on the extraction of relevant information from large web data storage. But still there is requirement of availability of automatic annotation of this extracted information into a systematic way so to be processed later for various purposes Web information extraction and annotation has been active research area in web mining. A huge amount of the data is available on the web. The user enter the search input query in the search engine, and search engine return the dynamically search output records on Web browser. Many E-commerce sites are available to users, for example, when a user wants to check the details while buying a notebook such as configuration and price, but such type of information is only stored in the form of hidden back-end databases of the various notepad vendors, then the user has visit to each web site and collect regarding information from various web site and distinguish these all retrieved information manually so he can get the required product at reasonable price. This is a time consuming process & due to human effort it leads to inaccuracy up to particular extent. There is a need for technique which should help us to provide retrieved relevant data as per user requirements. The last decade focus on multiple methodologies in firing queries, information fetching & optimization. The concept of wrapper is introduced. The wrapper is a software concept which wraps the contents of a web page using its source code via HTTP protocols [8] but it does not change the original query mechanism of that web page. This scenario assumes that every web database is having a common schema design. Therefore, we use the terms extractors and wrappers interchangeably [2]. We know that Word Wide Web having huge amount of data available on it but there is no tools or technology to extract relevant information from Web databases. In deep web databases search engines is referred as Web databases (WDB). When we extract the pages, the resulted pages returned from a WDB have multiple Search Result Records (SRRs). Each SRRs contain multiple data units each of which describes one aspect of real-world entity & text units [1]. Consider a book comparison web; we can compare SRRs on a result page from a book WDB. Each SRRs represents one book with several data &text units .It consists text node outside the <HTML>, Tag node surrounded by HTML Tags & title, author ,price, publication& the values associated with it as data units. A data unit is a piece of text that semantically represents one concept of an entity. It corresponds to the value of record under an attribute. It different from the text node which is refers to the sequence of text surrounded by a pair of HTML tag.

The relationship between the data unit and text node is very important for the purpose of annotation because the text node are not always identical to data nodes. The WDBs has multiple sites to store in it. For this task, labeling to required data & storing the collected SRR into a data base is important. Early applications require tremendous human efforts to annotate data units manually, which severely limit their scalability. Later approaches focus on how to automatically assign labels to the data units within the SRRs returned from WDBs. So this well reduces human involvement &increase the accuracy. For example in a book comparison website we wish to find the price details from the different websites for

the same book so we can decide the choice to buy the book with the reasonable price & the reliable website. The ISBNs can be compared to achieve this. If ISBNs are not available, their titles and authors could be compared.

2. LITERATURE SURVEY

The World Wide Web is having vital data in numerous formats the users have to deal with this data by using a search based form. The user will retrieve the information by firing the query. In traditional approach the search base form is design to fire the queries & required data is fetched. HTML form is containing the plain text. Querying, Integration, and Meditation etc. are used. But this techniques are not effective to produce accurate search result record from web databases, because of human involvement and poor quality of the data extraction output. Two main problem aeries during extracting the relevant information First: to categorized the unstructured view of data such as search engine. Second: categorized structure and semi-structure view of data. The web sites are also having heterogeneous nature due to language independent. The e commerce website or the information portals are updating their content on a regular basic. *Domain oriented approach* is used to automatically extract news; the domain oriented approach is based on tree edit-distance approach. This approach is not only capable for to extract relevant information text passages but also eliminates not-useful matters e.g. banners, menus and links. The tree edit distance algorithm was used for news extraction [4].The web data is now machined process able so, we require the relevant information extraction with the semantic grouping. The semantic grouping means the data with similar meaning can form group with same concept. XML/RDF has been widely used for representing semantic web that required annotation for recognition of semantic web. These techniques provide manual mapping of unlabeled document segment to ontological concepts. In bootstrapping semantic labeling is addressed in semantic web annotation. The presentation style & spatial locality in the HTML tag is focused [3].The sites like educational, news portal and e-commerce are dynamically update contents on a regular basis so called as content-rich web sites contents management software that creates HTML pages by populating templates from databases. The two things have to be focused. Spatial locality in HTML page and its corresponding DOM tree can also representing the content similarity. The structural analysis technique use to group together related elements in a HTML pages into unlabeled tree. The algorithm can use the hand-labeled concept instances from HTLM pages for identification of unlabeled concept instances in HTML pages and assigns semantic labels to them. The algorithm does not used hand-crafted ontology. For determining the consistency in presentation style we can use the feature extraction i.e. likelihood measures the closeness of data item to the concept at every node in the partition tree is used. So the data belong to same concept or set of concepts lie under similar group.

Table 1. Analysis of Approaches based on Techniques & tools used

Sr. No	Approaches	Techniques	Tools	Limitation
1.	Manual	Identify & Extract data items using wrapper	Minerva TSIMMIS Web-OQL	Low Efficiency & Poor scalability
2.	Semi-Automatic	Sequence based	WIEL Soft-Mealy Stalker	Manual efforts for labeling Web pages & time consuming
		Tree based	W4F XWrap	
3.	Automatic	Data Record Extraction	Omini	Vary as per the techniques, Only text node level annotation
		Data Record Extraction	Road-Runner IEPAD	
		HTML Tag Tree Structure	DEPTA DeLa	

For spatial locality we can use the likelihood estimation to assign the semantic labeled to nodes in partitioning tree. For improving ambiguity we can use the bipartite-graph based ambiguity resolution technique to provide the facility disambiguation to improve the precision of semantic label assignment. Three types of approach for data extraction techniques are analyzed on the basis of the various techniques and tools [7]. As per the analysis from Table 1, the limitation of manual approach had overcome by inducing sequence based and tree based techniques. In RoadRunner[11] comparison between HTML pages and generate wrapper based on their similarity and differences. The Labeller is used for the automatic wrapper generation [5] Due to problem of human efforts and low efficiency, the unsupervised approach is an active research area in data extraction. Automatic data extraction approach is mainly categorized into three techniques data records extraction, HTML tag tree structure, Tree and pattern matching. But this approaches not suitable for the dynamic Web databases. ViDE is the Visual data extraction system which is independently works without HTML tag tree structure. ViDE is focused on the Visual features of the Web pages. ViDE is primarily based on the visual features human users can capture on the deep Web pages while also utilizing some simple non visual information such as data types and frequent symbols to make the solution more robust. [7]

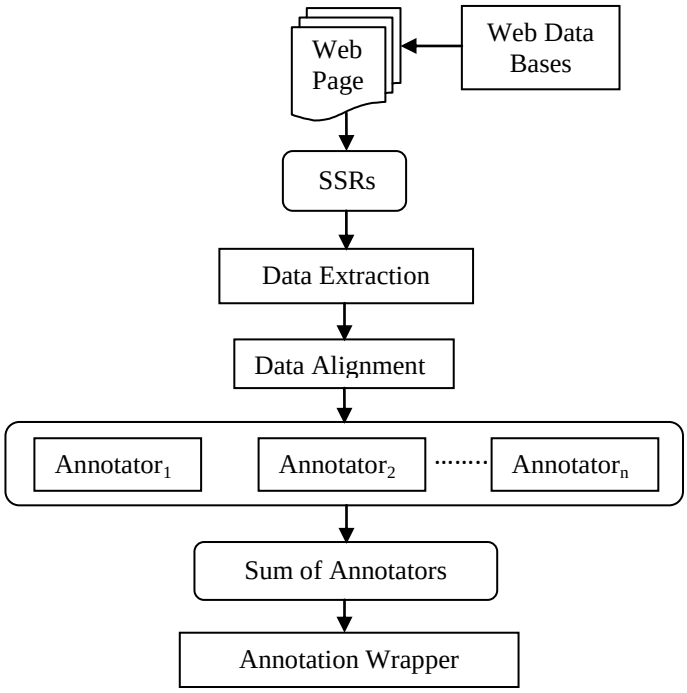

Fig 1: Data Extraction and Annotation

3. DATA EXTRACTION & ANNOTATION

According to user query search engine provide the information from the back-end deep web databases or we can say hidden database. The data extraction is performed by the wrapper induction many approaches focused on the effective grammar or regular expression for wrapper induction. But wrapper induction is used for data extraction not for automatic annotation [1] or labeling the data records. The Data extraction and Annotation system as shown in Fig. 1 Consists of four major components: from deep web crawler [10], a wrapper generator, a data aligner and a label assigner (Annotators).

Web Crawler: Web Crawler are a tool that solving the resource discovery problem in the World Wide Web. Find search result record from the hidden web, two main function of the Web crawler is first: To building an indexes of the various search result records and second: Navigation the web automatically on the basis of user demands.

Wrapper: Wrapper is a program or set of rules are to define for the HTML tags for Web data extraction. Wrapper generates automatic regular expression for HTML web pages, and performs heuristic-based automatic data extraction and annotation for web databases.

Data Aligner: Given the induced wrapper and the web pages, the data aligner first extracts data objects from the pages by matching the wrapper with the token sequence of each page. It then filters out the HTML tags and rearranges the data instances into a table similar to the table defined in a relational DBMS, where rows represent data instances and columns represent attributes.

Annotation/Label Assigner: The main roll of label assigner is assigning labels to the data units by matching the form labels obtained by the form crawler to the columns of the table. The basic idea is that the query word submitted through the form elements will probably reappear in the corresponding fields of the data objects, since the web sites usually try their best to provide the most relevant data back to the users.

3.1 Data Extraction

Given a regular expression pattern and a token sequence representing the web page, a nondeterministic, finite-state automaton can be constructed and employed to match its occurrences from the string sequences representing web pages. each occurrence of the regular expression represents one data object from the web page so we can found the occurrence from regular expression & from data tree.

A data-tree is defined recursively as follows: [2]

- If the regular expression is atomic, then the data-tree is a single node and the occurrence of the expression is the node label.
- If the regular expression is $E_1E_2...E_n$, then the data-tree is a node with n children and the i^{th} $(1<i<n)$ child is a data-tree that records the occurrence of E_i.
- If the regular expression is $(E_1|E_2)$, then the data-tree is a node with one child that records the occurrence of either E_1 or E_2.
- If the regular expression is $(E)^*$ and there are m occurrences of E, then the data-tree is a node with m children and the i^{th} $(1<i<m)$ child is a data-tree that records the m^{th} occurrence of E.

The following methods are used for building DOM tree

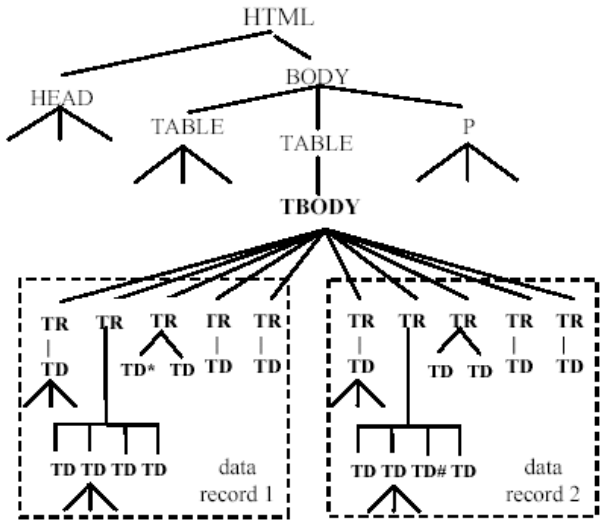

Fig. 3 Building DOM Tree

Fig. 4 Sample HTML page

3.1.1 Edit distance

It is defined as the no of point mutations to change, insert or delete a letter. The matrix can be used to hold the edit distance.

3.1.2 Tree edit distance

Tree edit distance between two trees A and B (labeled ordered rooted trees) is the cost associated with the minimum set of operations needed to transform A into B. The set of operations used to define tree edit distance includes three operations like node removal, node insertion& node replacement. A cost is assigned to each of the operations

3.1.3 Multiple alignments

Pair wise alignment is not sufficient because a web page usually contains more than one data records. We need multiple alignments. Two techniques are utilized for this: Center Star method & Partial tree alignment. This is a classic technique, and quite simple. It is commonly used for multiple string alignments, but can be adapted for trees.

3.1.4 Building DOM trees

The usual first step is to build a DOM tree (tag tree) of a HTML page. Most HTML tags work in pairs. Within each corresponding tag-pair, there can be other pairs of tags, resulting in a nested structure. Building a DOM tree from a page using its HTML code is thus natural. As per Fig.1, in the tree, each pair of tags is a node, and the nested it are the children of the node.

4. TYPES OF ANNOTATORS

The returned result page contains multiple SRRs. the data units corresponding to the same concept (attribute) often share special common features in certain patterns. Based on this, in this paper we used the six basic annotators have been defined to label data units, with each of them considering a special type of patterns/features. Each annotator are play unique role in labeling the name to the data units are extracted by the wrapper. Four of these annotators (i.e., table annotator, query-based annotator, in text prefix/suffix annotator, and common knowledge annotator) are similar to the annotation heuristics used by DeLa but there different implementations for three of them (i.e., table annotator, query-based annotator, and common knowledge annotator) [1] [6] [2].

4.1 Table Annotator

The resulted page fetch from multiple website consist of different SRR. Each information can be stored in the form of table .A table consist of different column header &rows. The cell of this table indicates the data unit. We can store the multiple data units. The table annotator used in Dela [2] Approach mainly focus on the <TD> tag elements. The information stored in <TD>elements is stored in the annotator table. But few websites contain the <TD> tag elements. So the table annotator is modified .The row is considered as SRR & the column is considered as attribute. The data unit having same features can be aligned under header & the column header. By considering the special feature we can annotate the SRR. Firstly we have to identify all the values of column then as per SRR we have to fill the data. In such way the limitation of Dela [2] is improved.

4.2 Query-Based Annotator

The SRR is always returned from WDB on the basis of fired query. When the user submits the data in the text box or select field from the list box on the search form, the query is fired on the WDB. Then the SRR is identified & the data is stored under the column header. The no of occurrences of matching the column header will decide the group & we can label it. The Dela uses only the local labels in the query. However, DeLa uses only local schema element names, not element names in the IIS [2].so, the new approach is use to utilize the global schema.

4.3 Schema Value Annotator

Many attributes on a search interface have predefined values on the interface. For example, the attribute vendor may have a set of predefined values in its selection list. More attributes in the IIS tend to have predefined values and these attributes are likely to have more such values than those in LIS. When values from different LIS are integrated then we have to modify the schema values to perform annotation.

4.4 Frequency Based Annotator

The adjacent units have different occurrence frequencies. The data units are always associated with the higher frequency & lower frequency. The higher frequencies are the attribute

names, as part of the template program for generating records, while the data units with the lower frequency most probably come from databases as embedded values. Suppose there is a group of lower frequency then we can easily find its preceding values shared by all data units in the group .We can analysis the data unit until it is different & map its preceding. Then we can combine the preceding to form the label.

4.5 In-Text Prefix/Suffix Annotator

In some cases, the data unit is aligned with its label. The data unit consists of the comma separated vales & the labels associated with it. Theses lie in a particular sequence separated from each other in all multiple SRR. After alignment it will form a group. The in text prefix/suffix will check for data unit. If the same prefix is there &not a deliiminator then it is removed from all data units but if the number of data nodes match with the same suffix to the data node within next group then the suffix is used for the annotation. Any group whose data unit texts are completely identical is not considered by this annotator.

4.6 Common Knowledge Annotator

Some data units on the result page are self-explanatory because of the common knowledge shared by human beings For example, "in stock" and "out of stock" occurs in many SRRs from e-commerce sites. Human users understand that it is about the availability of the product because this is common knowledge. Each common concept contains a label and a set of patterns or values. As another example, the e-mail address (assume all lower cases) so the common knowledge annotator work on the data units which exploit the interpretation of common knowledge data.

5. DATA UNITS SIMILARITIES

The data alignment is to put the data units of the same concept into one group so that they can be annotated holistically. Whether two data units belong to the same concept is determined by how similar they are based on the features likes *Data content (DC), Presentation Style (PS), Data Types (DT), Tag path (TP)* and *Adjacency (AD)* the similarity between two data units (or two text nodes) d1 and d2 is a weighted sum of the similarities of the five features between text nodes and data units as in [1] Yiyao Lu et al.

$$Sim(d_1,d_2) = w_1*SimC(d_1,d_2) + w_2*SimP(d_1,d_2) + w_3*SimD(d_1,d_2) + w_4*SimT(d_1,d_2) + w_5*SimA(d_1,d_2) \quad (1)$$

5.1 Data content similarity (Sim C)

It is the Cosine similarity between the term frequency vectors of d_1 and d_2: Where, V_d is the frequency vector of the terms inside data unit d, $\|V_{d1}\|$ is the length of V_d, and the numerator is the inner product of two vectors [1].

$$SimC(d_1,d_2) = V_{d1}+V_{d2} / \|V_{d1}\| * \|V_{d2}\| \quad (2)$$

5.2 Presentation style similarity (Sim P)

It is the average of the style feature scores (FS) over all six presentation style features (F) between d_1 and d_2 [1]

$$SimP(d_1,d_2) = \sum_{i=0}^{6} FSi/6 \quad (3)$$

5.3 Data type similarity (Sim D)

It is determined by the common sequence of the component data types between two data units. The longest common sequence (LCS) cannot be longer than the number of component data types in these two data units. Thus, let t_1 and t_2 be the sequences of the data types of d_1 and d_2, respectively, and TLen(t) represent the number of component types of data type t, the data type similarity between data units d_1 and d_2 is [1]

$$SimD(d_1,d_2) = LCS(t_1,t_2) / Max(Tlen(t_1), Tlen(t_2)) \quad (4)$$

5.4 Tag path similarity (Sim T)

This is the edit distance (EDT) between the tag paths of two data units. The edit distance here refers to the number of insertions and deletions of tags needed to transform one tag path into the other. It can be seen that the maximum number of possible operations needed is the total number of tags in the two tag paths. Let p_1 and p_2 be the tag paths of d_1 and d_2, respectively, and PLen(p) denote the number of tags in tag path p, the tag path similarity between d_1 and d_2 is [1]

$$SimT(d_1,d_2) = 1- EDT(p_1,p_2) / PLen(p_1) + PLen(p_2) \quad (5)$$

5.5 Adjacency similarity (Sim A)

The adjacency similarity between two data units d_1 and d_2 is the average if the similarity between d_1^p and d_2^p and the similarity between d_1^p and d_2^p, that is [1]

$$SimA(d_1,d_2) = (Sim'(d_1^p, d_2^p) + Sim'(d_1^p, d_2^p)) /2 \quad (6)$$

6. PHASES OF ANNOTATOR

From the SRR, first identify all data units and then organize them into different groups with each group corresponding to a different concept. The data unit with same concept can fall under the same column header like table annotator. E.g.: All names of the vendors for notepad are in group together. Grouping data units of the same semantic can help identify the common patterns and features among these data units [1] so it will help for better accuracy in semantic annotation.

6.1 Alignment Phase

This phase identify all data units in the SRRs and then organize them into different groups with each group corresponding to a different concept

6.2 Annotation Phase

In this phase, single or combined multiple annotators are used as per the requirement for annotation. This work on the probability based.

6.3 Wrapper generation Phase

The wrapper set the rules for extracting the information from same WDB. The annotator wrapper can be used for further analysis. We can write the wrapper after combining the multiple annotators. For mapping the information between text node &data node we have to first find the relationship between them. Relationship between the data unit and text node are as bellow:

- **One-to-One**

In some cases the text nodes are equivalent to data nodes so can be used for annotation in a easy way. For example the <a>… in HTML itself indicate the data value & attribute .But this is not the general case always to be considered in fig 4. Show that *title* attribute each search result considers as a one-to-one relationship [2][1].

- **One-to-Many**

This relationship contained many data nodes can be associated with one text node. For example by observing one particular text node we can multiple information (data units) are present in single text node like publication details. As shown in fig 4. each SRR (e.g., "Springer-Verlag/1999/0387984135/0.06667" in the first record) is a single text node. It consists of four semantic data units: Publisher, Publication Date, ISBN, and Relevance Score [1].

- **Many-to-One**

In this case, multiple text nodes together form a data unit. For example the vendor name can be embedded inside the <a>.. tag .Another example can be considered that the price can be entitled within … tag [1].

- **One-to-Nothing**

In this case the text node is not part of any data unit. For Example vender name does not contain data unit but instead describe the meaning data unit. It is also known as *Template text node* [1].

7. DATA & TEXT NODE ALIGNMENT

Data alignment algorithm is based on the assumption that attributes appear in the same order across all SRRs on the same result page, although the SRRs may contain different sets of attributes (due to missing values) [1]. SRRs from the same WDB are generated by the same schema. Thus, we can consider the SRRs on a result page in a table format where each row represents one SRR and each cell holds a data unit (or empty if the data unit is not available). The goal of alignment is to move the data units in the table so that every alignment group (column) contain similar data unit, preserving the order within every SRR is preserved. The alignment algorithm is based on following steps:

- **Merge Text Nodes**

This mainly focuses on removing the decorative or presentation style tags so that all text nodes can be merged.

- **Align Text Nodes**

This will align the nodes with the same concept or set of concepts under one group for atomic node as well as for composite nodes.

- **Split (Composite) Text Node**

The split node again have to be focused on the annotation work .we have to split the "values" in composite text nodes into individual data units. This step is carried out based on the text nodes in the same group.

- **Align Data Units**

This step is to separate each composite group into multiple aligned groups with each containing the data units of the same concept.

8. CONCLUSION

In this paper we reviewed that various data extraction techniques as well as automatic annotation approach using multiple annotators from different Web data bases. We also surveyed that how the data extraction from the various web pages but the traditional approach is having many drawbacks like human interference, the inaccuracy in result and poor scalability. Some approach are used the different feature extraction techniques such as sequence based Tree edit distance, DOM tree, pattern matching and HTML tag structure. In visual data extraction approach is the language independent. This approach mainly focus on the presentation style of and extract the visually information from the template. But still there is need to identify the best technique for data annotation problems.

9. ACKNOWLEDGMENT

I would like to thank the University Authorities to provide basic facilities for carrying out the research work. I would like to thank my guide Dr. Sachin. N. Deshmukh and my friend Miss. Dipali Gaikwad for most support and encouragement, valuable advices on grammar and theme of the paper.

10. REFERENCES

[1] Y. Lu, H. He, H. Zhao, W. Meng, C.Yu "Annotating Search Results from Web Databases", IEEE Knowledge and Data Engg"., vol. 25, March-2013.

[2] J. Wang and F.H. Lochovsky, "Data Extraction and Label Assignment for Web Databases," Proc. 12th Int'l Conf. World Wide Web (WWW), 2003.

[3] S. Mukherjee, I . V. Ramakrishnan and A. Singh, "Bootstrapping Semantic Annotation for Content-Rich HTML Documents", Proc. IEEE Int'l Conf. Data Eng. (ICDE)", 2005.

[4] Davi de Casto Reis, Paulo B. Golgher and Altigran S. da Silva, "Automatic Web News Extraction Using Tree Edit Distance", Proc. ACM World Wide Web (WWW), 2004.

[5] L. Arlotta, V. Crescenzi, G. Mecca, and P. Merialdo, "Automatic Annotation of Data Extracted from Large Web Sites," Proc. Sixth Int'l Workshop the Web and Databases (WebDB), 2003.

[6] Y. Lu, H. He, H. Zhao, W. Meng, and C. Yu, "Annotating Structured Data of the Deep Web," Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE), 2007.

[7] W. Liu, X Meng and W. Meng, "ViDE: A Vision-Based Approach for Deep Web Data Extraction," *IEEE Trans. Knowledge and Data Engg.*, vol. 22, no. 3, pp. 447-460, March 2010.

[8] H. He, W. Meng, C. Yu and Z. Wu, "Automatic Integration of Web Interface with WISE-Intigrator," VLDB J., vol. 13, no. 3 pp.256-273, Sept 2004.

[9] Chia-Hui Chang, Mohammed Kayed, Moheb Ramzy Girgis and Khaled Shaalan "A Survey of Web Information Extraction Systems" IEEE, TKDE-0475-1104.R3.

[10] J. Madhavan, D. Ko, L. Lot, V. Ganapathy, A. Rasmussen, and A.Y. Halevy, "Google's Deep Web Crawl," Proc. VLDB Endowment, vol. 1, no. 2, pp.

[11] 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.