Tree Structured Web Template Matching for Deep Web Extraction

R.Vijay[1], Dr. K. Prasadh [2]

[1]*Research Scholar, M.S University*
Tamil Nadu, India

[2]*VK college of Engineering and Technology*
Parippally, Kerala

***Abstract*— Usually, Deep Web contains more accurate and valuable information than based on users' requests. But, making use of such combined structure of extracted information needs significant efforts because the obtained web pages are utilized for revelation not for exchanging the data. So, extracting the relative data from Web pages from the absolute Web sites has been a primary step for Web information combination. A significant feature of extraction of web pages is that the absolute web pages share the similar template because they are engendered with a predefined outline by stopping data values. Even though many approaches have been planned for deep Web data extraction very few works discuss this problem at a page-level. Mohammed Kayed et. Al., presented approach for web page extraction method by implementing the new approach called FiVaTech, to automatically notice the extracted data of a Web site. But the drawback is that it can be applied only to 2 or 3 set of web pages alone. To increase the number of web pages for web page extraction, in this work, proposed a novel representation of page generation with Tree Based Template Matches (TBTM). TBTM deduce the schema and templates for each individual Deep Web site, which contains either singleton or multiple data records in one Web page. Experimental evaluation is done to estimate the performance of the TBTM scheme with a large set of database for deep web data extraction against existing FivaTech. The performance of the proposed scheme is measured in terms of relativity, number of schemas, execution time and comparison results showed that the proposed TBTM attains 10-12% high in obtaining the relative information over the set of web pages.**

***Keywords*— Deep Web Page, Tree structure, Template Matching, Schema extraction, Data records.**

I. INTRODUCTION

Given the quick development and achievement of open information sources on the World Wide Web, it is gradually more gorgeous to extract data from these sources and build it for further dispensation by end users and request programs. Data hauled out from Web sites can serve up as the catalyst for a assortment of tasks, counting information reclamation (e.g. business intelligence), event supervising (news and stock marketplace), and electronic business (shopping assessment).

Data extraction is where data is examined and lagged behind through to regain appropriate information from data sources (like a database) in a precise pattern. Further data exemption is completed, which engages adding metadata and other data incorporation; another procedure in the data workflow. The best part of data pulling out comes from shapeless data sources and diverse data formats. Data extraction expansion can be demanding task in processing information. Every organization has significant data hidden in bends of the company, now and then increase diagonally over the world.

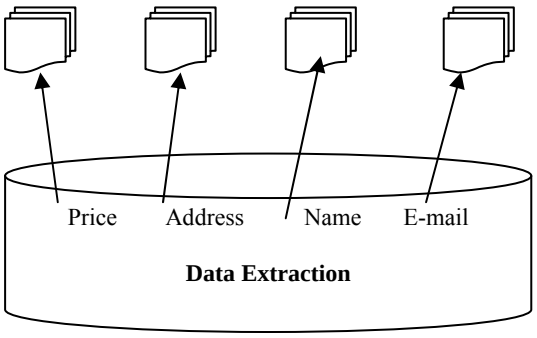

Fig. 1 Data Extraction

Once the rare data is clustered, the genuine work begins. If the relationship needs to employ this information to create reports, create creation decisions, or construct intelligent business choices, must haul out the germane data from source documents, business news, web sites, and a lot of of other sources. The above figure describes how the data can be hauled out from the data sources and it demonstrates where the data can be recovered from diverse online databases. The strategy routinely haul out the query outcome records (QRRs) from HTML pages (to a user query) animatedly engendered by a deep web site. A perfect record extractor should attain the following:

1) All data records in the data region are dogged up from the database and

2) For each extracted data record, no data item is neglected and no erroneous data item is included.

Extracting prepared data from the web pages is obviously very helpful, as it allows us to pretense multifaceted queries over the data. Extracting prearranged data has also been documented as a significant sub-problem in information incorporation systems, which incorporate the data present in diverse web-sites. Consequently, there has been a lot of current study in the database and AI communities on the crisis of mining data from web pages.

The rest of the paper is organized as follows. Section 2 provides the related literature works regarding the data extraction problem. Section 3 provides a brief overview of the proposed approach for solving the web data extraction problem. Section 4 describes the experimental evaluation and section 5 describes the results and compared with the existing approach. Section 6 ends with conclusion.

II. LITERATURE REVIEW

Web data extraction has been a significant part for numerous examinations of Web data analysis requests. In [1], the author designed the data extraction crisis as the decoding procedure of formation of page supported with the structured data and tree templates. The author proposed an unsupervised, page-level data extraction strategy to deduct the schema and templates for every Website, which has either singleton or numerous data records in one Webpage.

Survey reveals that the approaches utilized for web data extraction require being enhanced to attain the effectiveness and accuracy of separate wrappers. In addition to this, investigations signified that the integration of a lightweight ontological strategy utilizing WordNet is capable to determine the likeliness of data records and notice the exact data region with precision employing the semantic properties of these data records [2].

In [3], the author proposed a construction technique for new web page repetitive and URL feature based technique for Deep Web data extraction. It utilizes contiguous repetitive tag region and similar URL to separate the page into blocks, positioned the data region and dig out the particular URL template, which is processed to quickly recognize the data region and the limit of data records in the same set of pages. The subsequent changes in the Web, termed as Semantic Web, have to enhance the Web with semantic page annotations to facilitate knowledge-level querying and investigation. On the other hand, manual production of these ontologies is a time consuming and complex task. In [4], the author explained an automatic pulling out scheme that study domain ontologies for semantic web from deep web.

To progress the precision of extracting attributes, and strategy is presented to haul out the attributes heuristically. During the mining, attributes are augmented utilizing ontology by which deep semantic knowledge is obtained with the query interfaces [5]. Object matching is a vital footstep to incorporation of Deep Web sources [6]. Existing methods presume that record pulling out and attribute segmentation are of elevated accuracy. But since of restriction of extraction methods, information increased during the incomplete processes of methodology. If objects are matched supported on noisy and unfinished information, the author can not attain acceptable performance.

Ever more, numerous data sources emerge as online databases, concealed following query forms, thus shaping the deep Web. The recognition of this novel medium for data distribution is important to novel problems in data integration. In [7], the author proposed two strategies, which are the case form approach and the combination model approach, correspondingly, to professionally achieve a roughly absolute position of output plan attributes from a deep Web data source.

For the trendy DIV page design in Web Pages, the author presented a technique supported with the situation of DIV to haul out main text from the body of Web pages by renovating, lingering atomic DIV and examining DIV position [8]. The visualization information of Web page is practical for information extraction, which shuns utilizing the sophisticate natural language processing knowledge. The paper [9] shared the natural language processing expertise with vision disposition of HTML page in the request of information drawing out for Web page, processed out applicable research.

In association with the crisis of present composite implementation, high error rate and low taking out speed of Web information extraction expertise, the paper [10] proposed a novel technique of Web extraction supported with the uniqueness of construction of Web page. To progress the effectiveness of information extraction, it needs us to further investigate the automatic technique of Web information extraction [11]. By supporting the DOM extraction technique, page clustering is utilized to discover the high comparison clusters, and enhanced the accurateness of clustering results by utilizing the similarity measure [12]. But the accuracy of the outcomes does not met with the data extraction processes.

III. PROPOSED METHODOLOGY

In this paper, the process of deep web data extraction is done effectively by adapting the TBTM which considers multiple input pages for taking over the processes. The proposed TBTM approach for deep web data extraction is processed under four different processes. The first process describes the process of identifying the similar set of templates from the set of web pages. The second process describes the process of retrieving the data records from the web pages. The third process depicts the presence of repetition of web pages from the absolute web pages. The fourth process describes the grouping of relative information from the web pages. The architecture diagram of the proposed TBTM approach is shown in fig 2.

Given input as a training set of web pages from a Web site. The presence of scheme is identified based on utilizing the DOM trees. Likewise, use DOM trees for all set of extracted web pages and combined all set of DOM trees into a single set of tree. From this combined set of tree, leaf nodes are recognized to identify the repetitive nodes. The resultant tree is then utilized to distinguish the template and the schema of the Web site.

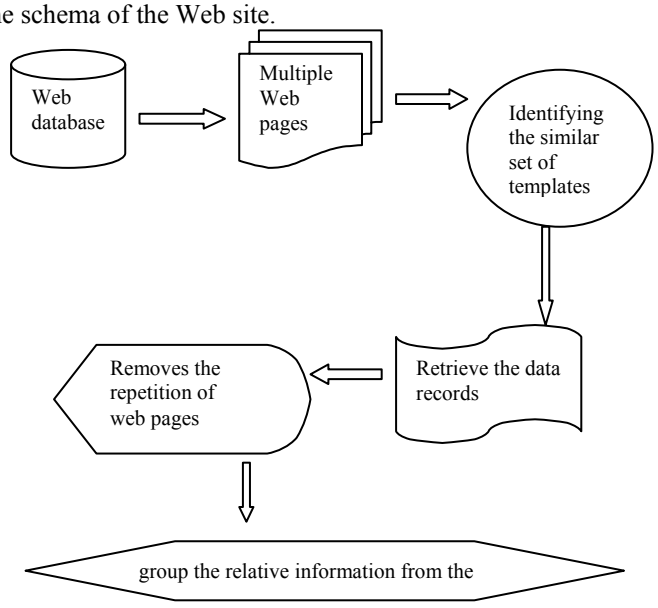

Fig. 2 Architecture diagram of the proposed Tree Based Template Matching Approach e

Given input as a training set of web pages from a Web site. The presence of scheme is identified based on utilizing the DOM trees. Likewise, use DOM trees for all set of extracted web pages and combined all set of DOM trees into a single set of tree. From this combined set of tree, leaf nodes are recognized to identify the repetitive nodes. The resultant tree is then utilized to distinguish the template and the schema of the Web site.

Consider the combined DOM trees have the same root node and tries to develop the tree in a depth-first fashion. Form a matrix representation for the constructed tree by specifying the child nodes for every tree. The system then enters four important steps

- Tree construction with templates,
- Matrix representation based on tree,
- Pattern mining, and
- Optional nodes merging:

A. Tree Construction with Templates

For the identification of the presence of similar set of templates for a website, the construction of tree is done. The trees are built for the comparison of whether two sub-trees (with the same root tags) are similar or not. Determine the sub-tree matching performance to identify the similarity. By matching the leaf nodes with dynamic programming, the replacement of node is done if it does not match. Once the leaf nodes in the sub-trees are matched, instead of changing the entire root of the sub-tree, simply changing the child leaf nodes based on node replacement strategy. In the meantime, the matching is regularized from 0 to 1 supported with the contemplation of set type data. The figure below describes the sample representation.

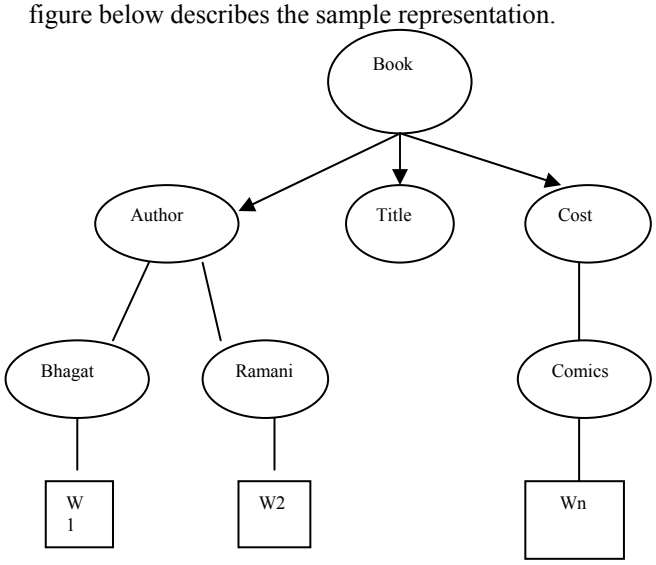

Fig. 3 Sample representation of tree

B. Matrix Representation Based on Tree

After representation of tree, the matrix is formed by specifying the nodes in the matrix M. Consequently, two nodes with the related tag are specified by the matching sign if their matching score is larger than a threshold T. For the sub-nodes in the tree representation, two text nodes acquire the similar symbol when they contain the similar text values (or else, they obtain diverse set of symbols). The alignment algorithm tries to make parallel with the set of nodes in the matrix M, row by row, to exchange M into an associated matrix, where every row contain either the similar symbol for each column or leaf nodes of diverse set of symbols, which will be tagged as basic-typed. From the associated matrix M, the user lists the set of nodes where each node specified in a defined row.

At each row in the formed matrix, the task alignedRow ensures if the row is associated or not. If it is associated with the similar symbol or when the child leaf nodes of diverse symbols appears only in its existing column; these nodes are recognized as deviation. If not, the algorithm repetedly doing this procedure tries to make parallel this row. In every iteration, a column (a node) shiftColumn is chosen from the present row and all of the nodes in this column are moved downward a distance shiftLength in the matrix M and scrap the blank spaces with an illogical node.

The node selection nrc is identified based on the position of nodes in the matrix representation in the column c from row r to be transferred depends on two values:

- Span (nrc) and
- checkSpanr(nrc)

The first value is determined as the highest number of diverse set of nodes based on their occurrences in the input web pages nrc in every column of the matrix. Besides, this value denotes the greatest promising cycle duration of the node. If the presence of nrc is identified in each column c, then regard as it as a free node and its life span over the network will be 0. In the meantime, the value checkSpanr of a node nrc at row r depends on whether there exists a node with the matching symbol at row rup and column c, such that,

$$checkspan_r(n_{rc}) = \begin{cases} 1 \\ 0 \quad \text{if } (r\text{-}r_{up}) = \text{span } (n_{rc}) \\ -1 \end{cases}$$

The value 1 is met when the value for r-rup is greater than the value obtained through span (nrc) and -1 obtained when the value for r-rup is less than the value obtained through span (nrc). There are set of rules to be followed over in order to met with the following conditions,

Rule 1: If checkspan (nrc) = 1, then select the associated set of nodes (input web pages) with the specified row and column of the representation of the matrix.

Rule 2: If checkspan (nrc) = 1 with the presence of same set of symbols in the subsequent row or column in the matrix, then the associated set of column values has to be chosen.

Rule 3: If both rule R1 and R2 fail, align the present set of rows independently by splitting it into 2 parts based on the alignment of nodes.

With these set of rules, the nodes in the matrices are aligned and processed.

C. Repetitive Pattern Mining

If numerous set of web pages are given as input, then it is simple to hold set-typed data. With a naïve approach, it is easy to determine the repetitiveness of web pages which is given as input. Nevertheless, there can be numerous repetitive patterns exposed and a pattern can be entrenched in one more pattern, which creates the assumption of the

template tricky. The effect of missing possible data is gripped in the previous step. As a result, it is necessary to focus on how recurring patterns are combined to presume the data structure. Detection of every successive repetitive pattern (tandem repeat) and combine them (by erasing all incidences except for the first one) from minute length to outsized length. This is since the prearranged data defined here are nested and every instance of a set-type should happen instantly to each other.

D. Analyzing the web pages as input

After the mining step, it is necessary to detect the occurrence of choices given to the users based on their queries. For occurrence, the vector of the distinct nodes is utilized. The occurrence vector for given input web pages wb is specified as,

the vector (wb1, wb2, wbu),

Where wbi is 1 if wb occurs in the ith occurrence, or 0 otherwise

E. Data Schema Detection

With the representation of tree, the schema of the input web pages should be simply inferred by eradicating the nodes containing single child and devoid of types and protecting all essential leaf nodes. The order of web pages is sorted in the form of tree with the site presences are specified in leaf node k. If an inner node is not a essential node and not tagged as set type, then it is specified as a tuple. The algorithm below describes the process of the entire improvised FiVaTech approach as follows,

// Algorithm

Input: Web database WD, Data records DR, Data items DI, Users U

With the obtained set of input web pages,
Design a tree T
Based on representation of T
Form a matrix M
Based on the presence of similarity over the contents in web pages

Shift the set of nodes from left to right or vice-versa
Form a set of rules R
Form a vector v (WD)
Identify the presence of schema
Separate the exact representation from the leaf child nodes
End

End

The above algorithm describes the entire process of identification of schema to enhance the extraction of deep web pages.

IV. Experimental Evalaution

An experimental evaluation is done to estimate the performance of the proposed tree based template matching approach. The proposed TBTTM approach experiments are done on a Pentium 4 1.9 GH, 512 MB PC. A large set of web database is utilized which consists of more than 10,000 entries of web database. These web databases are categorized into several domains. Several set of users submit their queries to extract the relevant information from the web database. For each Web database, we present three set of user queries and collect five deep Web pages holding three data records at any rate. With the set of web database, the enhanced co-citation algorithm is implemented to the database to extract the web pages directly from the web database. With the set of web pages, the data records and its corresponding data items are retrieved based on user query related information.

V. Results And Discussion

In this work, TBTM approach is efficiently extracted the web pages directly from the web database presents in multi-data region. For a diverse set of web pages, the users' required data records are extarcted and processed with the data items. After extracting the data records, the users' query related information is processed. The below table and graph describes the performance of the TBTM approach.

TABLE I
NO. OF SCHEMAS VS. RELATIVITY

No. of schemas	Relativity (%)	
	Tree based Template Matching (TBTM)	FiVaTech
10	32	17
20	40	22
30	45	28
40	53	34
50	59	42
60	64	48
70	72	56
80	79	67

The relativity of the obtained information extracted from the deep web pages are analyzed and processed based on the number of schemas present in the retrieved templates. The value of the TBTM is compared with the existing simple FiVaTech approach.

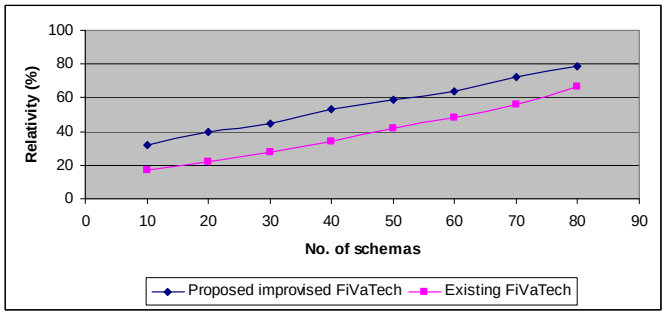

Fig. 4 No. of schemas vs. relativity

Fig 4 describes the information relativity obtained with the set of schemas in the websites. Compared to the existing FiVaTech, the proposed improvised FiVaTech has high relativity in obtaining the exact data from deep web

pages. Because, in the proposed work, the trees are constructed based on the presence of the obtained results for the given set of input pages. Based on the tree formation, the representation of matrix is done with the specified schemas and templates. So, only the required set of information alone extracted from the websites and given to the user. The variance in the relativity is 15-17% high in the proposed improvised FiVaTech.

TABLE III
NO. OF ENTRIES IN DATABASE VS. PRECISION, RECALL

No. of entries	TBTM		FiVaTech	
	Precision (%)	Recall (%)	Precision (%)	Recall (%)
100	96	96.4	94.3	95.3
200	96.3	96.9	95	95.5
300	96.5	97.1	95.3	95.8
400	96.9	97.5	95.6	96.1
500	97.2	97.8	95.9	96.5
600	97.5	98.2	96	96.8
700	98.3	98.5	96.5	97.3
800	98.5	98.7	96.7	97.5

The determination of precision and recall is made out in table II based on the number of entries in the web database. The precision and recall of the proposed improvised FiVaTech approach is compared with the existing FiVaTech approach

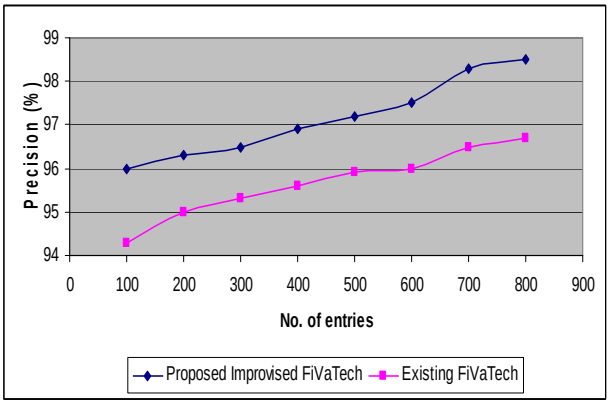

Fig. 5 No. of entries in database vs. Precision

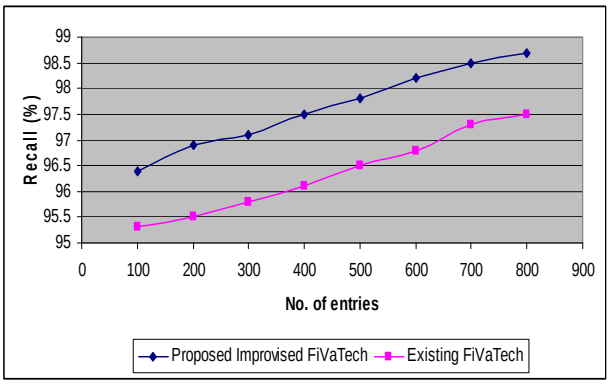

Fig. 6 No. of entries in database vs. recall

Fig 5 & 6 describes the determination of precision and recall value of the extracted web pages based on the number of entries in the web database. The precision value is measured based on the rate at which the retrieved web pages are relevant to the users' queries. The recall value is measured based on the rate at which the number of related web pages retrieved to the users' queries. Compared to the existing FiVaTech approach, the TBTM approach provides a better precision and recall rate outcome since TBTM extracts the web pages directly from the web database using tree construction and matrix representation by filtering the web pages. The variance is 2-3% high in precision and 1-2% high in recall rate in the proposed FiVaTech approach

TABLE IIIII
NO. OF ENTRIES IN DATABASE VS. TIME CONSUMPTION

No. of entries in database	Time consumption (sec)	
	TBTM	FiVaTech
100	15	20
200	19	26
300	26	31
400	30	36
500	34	43
600	40	50

The consumption of time required to extract the relevant deep web pages based on the entries of the web database illustrated in table III. The consumed time of the proposed TBTM approach is compared with the existing FiVaTech approach

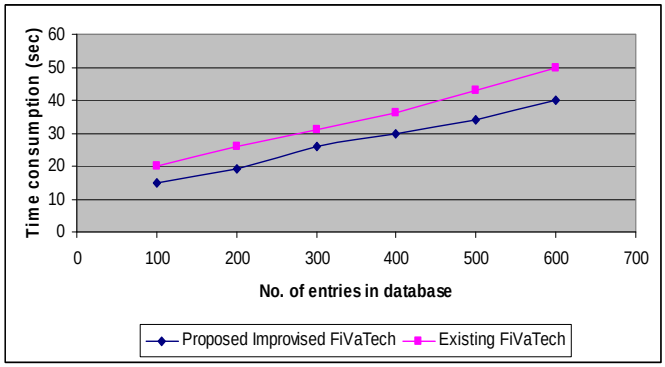

Fig. 7 No. of entries in database vs. time consumption

Fig 7 describes the consumption of time required to extract the relevant deep web pages based on the entries of the web database. Time consumption is measured in terms of seconds. Compared to the existing FiVaTech approach, TBTM approach consumes less time to extract the web pages. The variance in the time consumption is 5-10% less in the proposed improvised FiVaTech approach for web page data extraction.

Finally, it is being depicted that the TBTM efficiently extract the user relevant information from the web database based on their query by adapting the improvised FiVaTech approach.

VI. CONCLUSIONS

In this paper, presented tree based template matching for deep web data extraction approach, called TBTM to merge multiple DOM trees simultaneously. A new algorithm is designed for multiple string alignment which takes optional and set-type data into consideration. With the constructed fixed/variant pattern tree, we can easily deduce the schema and template for the input Web site. Compared to the existing FiVaTech approach, TBTM approach consumes less time to extract the web pages. Because the existing FiVaTech approach used IE as APIs for extracting the web pages from the web database with less number as input images. But in TBTM approach, the deep web pages are extracted directly from the database by accepting more number of web pages as input. TBTM approach has 11% better information relativity over the obtained web pages on deep web page extraction compared to existing deep web data extraction methods.

REFERENCES

[1] Kayed, Mohammed ; , "FiVaTech: Page-Level Web Data Extraction from Template Pages", Knowledge and Data Engineering, IEEE Transactions on (Volume:22 , Issue: 2), 2010

[2] Jer Lang Hong, "Data Extraction for Deep Web Using WordNet", Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on (Volume:41 , Issue: 6), 2011

[3] Xingyi Li , "Web page repetitive structure and URL feature based Deep Web data extraction", Communication Systems, Networks and Applications (ICCSNA), 2010 Second International Conference on (Volume:1), 2010

[4] Zhiming Cui et. Al., "From Wrapping to Knowledge: Domain Ontology Learning from Deep Web", International Symposiums on Information Processing (ISIP), 2008

[5] Ren Fei et. Al., "Parsing Query Interfaces of Deep Web from Specialization to Generalization", Intelligent Information Technology Application, 2009.

[6] Pengpeng Zhao ; Lin, C. ; Wei Fang ; Zhiming Cui, "A Hybrid Object Matching Method for Deep Web Information Integration", International Conference on Convergence Information Technology, 2007.

[7] Fan Wang ; Agrawal, G., "Extracting Output Metadata from Scientific Deep Web Data Sources", Ninth IEEE International Conference on Data Mining, 2009. ICDM '09.

[8] Palekar V.R., Ali M.S. and Meghe R., "DEEP WEB DATA EXTRACTION USING WEB-PROGRAMMING-LANGUAGE-INDEPENDENT APPROACH", Journal of Data Mining and Knowledge Discovery, ISSN: 2229–6662 & ISSN: 2229–6670, Volume 3, Issue 2, 2012

[9] Xunhua Liu et. Al., "On Web Page extraction based on position of DIV", The 2nd International Conference on (Volume:4) Computer and Automation Engineering (ICCAE), 2010

[10] Qingshui Li et. Al., "Study of Web Page Information topic extraction technology based on vision", 3rd IEEE International Conference on (Volume:9) Computer Science and Information Technology (ICCSIT), 2010

[11] Hu Mingsheng et.al., "An approach for text extraction from web news page", IEEE Symposium on Robotics and Applications (ISRA), 2012

[12] Yunfei Gong et. Al., "Automatic web page segmentation and information extraction using conditional random fields", IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2012.