Sindh Univ. Res. Jour. (Sci. Ser.) Vol. 48 (1) 145-150 (2016)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Database Technology on the Web: Query Interface Determining Algorithm for Deep Web Based on HTML Features and Hierarchical Clustering

R. A. SHAIKH, I. MEMON*, J. A. MAHAR++**, H. SHAIKH**

School of Computer Science and Engineering, UESTC, Chengdu 611731, China

Received 4th January 2014 and Revised 11th June 2015

Abstract: According to the features of Hypertext Markup Language, the interactive elements exist in the terminal of Document Object Model tree and they are close to each other in local area, we proposed a method to find web query interface which combines models and rules. In this method, after establishing tree model of Hypertext Markup Language, we locate the parts of interfaces by interaction density and cluster interactive groups by their similarity in local structure hierarchically. Then some non-query interfaces are filtered out in the help of content-filter composed of rules. This method avoids the excessive dependence on tag "form" and presents a better performance than traditional methods in the property of accuracy and generality. And the accuracy of experiment results on common dataset TEL-8 and self-organized dataset reached respectively to 90.1% and 92%.

Keywords: Web Query Interface, HTML Features, Hierarchical Clustering, Document Object Model Tree

1. INTRODUCTION

Web Query Interfaces (WQIs) are the most important sources of Deep Web searching engine, which can provide information much more valuable and professional than traditional engine, such as Google, Baidu and others (MengXiaofeng, 2001) (Latiri, 2003). Generally, WQIs are presented in Hypertext Markup Language (HTML) page in the form of semi-structured. And the fact that they are in a great amount and are quite different from each other in structure, style and function makes it difficult to find correct WQIs automatically. And we made some research on it. According to the features that interactive elements exist in the terminal of Document Object Model (DOM) tree and they are close to each other in local area, we proposed a new method to find WQIs automatically, which overcomes the excessive dependence on the tag "form" and presents a better performance than traditional method in accuracy and generality.

This passage is organized as followed. Section 2 introduces some traditional WQI methods and lists some definitions of the HTML feature. Section 3 describes the way to parser HTML page and model features of elements. Section 4 states how to cluster the elements belonging to the same WQI together and filter the non-query interfaces. Section 5 presents the records and analysis on experiments, which proves the effectiveness and generality of this method. Section 6 summaries the whole passage with further plan.

2. DISCUSSED PROBLEMS

We want to deliver a method to find huge amounts of heterogeneous WQI automatically. And the traditional methods are basically classified into pre-query and post-query ones. Post-query ones utilize features on page to find WQI, such as specific tag "<form>", visible text and some patterns in structure. And post-query ones send possible query tuples to detect the WQI according to the amount and hit rates of return-back results. So the key issue is how to construct correct search tuples and recognize the searching results. The research on pre-query method starts from the beginning of 20th century (Liu, 2007) (Heidy, 2013) (Eduard, 2009). Zhang (Zhang, 2004) assumed that the WQIs of different sources follow certain grammar rules and proposed the 2P grammar indicating the pattern and priority of WQI. But they can't describe all WQIs in an abstract way well and the amount of production rules was too big. In 2005, (Barbosa, 2005) (Barbosa, 2007) proposed a hierarchical strategy to extract HTML form related to specific domain. But this method involved too much human interference. (Wang, 2001) proposed a framework to find WQI based on ontology, which consists of web page classifier (WPC), feature of structure classifier (FSC) and feature of content classifier (FCC). They filtered domain-unrelated pages, recognized the form area and removed non-query form, respectively and the result was satisfactory. But the FSC took effect with excessive dependency on tag "form", and the capability of WQI finding could be limited.

++Corresponding author: J. A. MAHAR Email: mahar.javed@gmail.com
*College of Computer Science, Zhejiang University, Hangzhou 310027, China
**Department of Computer Science, Shah Abdul Latif University, Khairpur Mir's, Pakistan

Compared to the pre-query methods, the accumulation on post-query ones are rather small. (Lin, 2009) detected simple WQIs by the guidance of the amount and hit rate of search result and the keyword in interface, such as “search”. The fatal weakness is that it’s difficult to construct search tuples for complex WQI with many attributes and the multiple interactions forward-and -back waste network resource.

Some basic definitions associated with HTML features are presented as follows:

Definition 1: Dependency ability.

If the sub-tag has an attached relationship with its parent tag in structure or context, this tag is regarded as with dependency ability. For example, the tag “<option>” is embedded in its parent tag “<select>” in structure and context, so tag “<option>” has dependency ability.

Definition 2: Interaction element is a leaf node of DOM tree.

If we convert html to a DOM tree, the element with interaction ability is generally a leaf in DOM tree.

Definition 3: Interaction ability.

If certain tag can react to the user’s text input, click and other operation, this tag is regarded as with interaction ability. As related definition in HTML, the set of elements with interaction ability contains tag “<input>”, “<button>”, “<select>”, “<form>” and so on (HTML Forms).

Definition 4: Interaction density.

It means percentage of interaction elements in the area surrounded in certain tag. And the density of local area trends to fall with upwards tracking DOM tree.

3. ELEMENT MODELING

The core work in element modeling is to extract as much useful information as possible. According to the need to calculate interaction density and record DOM path of node, we applied the post-order traversal and pre-order traversal of the DOM tree to model the interface elements. In the process of the traversal mentioned above, we merged the node with dependency ability to its direct parent, which upgrades the integrity and accuracy of the description of elements in WQIs.

Model of interface element

Model of interface element are defined as followed: Note = {Tag, Id, Name, text, type{R-i} Res ability, {info}, visual text, Path Rec and “Tag” describes the html tag of this node; “Id” the value of the id attribute; “Name” the value of the name attribute; “Text” the outer html text. “Type” equals to 1 presenting the node with interaction ability and 2 the node without. “{Ri}” indicates interaction ability, in which the value of Rimeans the strength of interaction ability contributed by the nodes in “Res Ability” presents interaction density of local area surrounded by this node and it is defined as

$$\text{Re}\,s\ Ability\ (P) = \frac{\sum_{i=1}^{num}\sum_{j=1}^{1} i_{rj} + \sum_{j=1}^{1} p_{rj}}{\sum_{i=1}^{num}\sum_{j=1}^{2} i_{rj} + \sum_{j=1}^{2} p_{rj}}$$

In which p is the root node of the local area in the form of tree, num is the number of its direct descendant nodes, indicates the index of the sub-node, j indicates the element type. “{Info}” is a flat string composed of nested key-value pairs in the attributes in tag, such as style information and event registered information. Similar with JSON, “ Visible Text” presents the visible text only belonging to current node and it's a substring of “Text”, “Path Rec ” presents the path from tag “body” to the current node, each node in the path is described in the form of tag+”_”+index+”_”+amount, in which “tag” presents the name of html tag, “amount” the total amount of its sibling (including this node), and “index” the index of current node among them from left to right. And every string of node is connected with “/”. The set “child-j“is the set of child nodes of current node and j is used to distinguish from others.

HTML features Extraction

According to definitions mentioned in section 2, interaction density of local area should be calculated hierarchically upwards. So we establish the model “Tree” of the page in tree structure by post-order traversal. In the process above, a filter is used to merge the node with dependency ability to its direct parent node. Then a pre-order traversal is made, in which path information is recorded, and ”Visible Text” is got by subtraction between ”Text” of current node and the one of its direct descendant nodes.

Obviously, process of features extraction has a close relationship with the model we choose. What’s more, the need to filter nodes with dependency ability is also necessary. And we will convince you the necessity of the merging process with an example illustrated in **(Fig.1)**. With the increasing amount of Tag “option”, a form element with interaction ability, the interaction density around will dramatically rise. As presented in Table 1, though the interaction density “Res Ability” of tag “select” is unchanged, the interaction density “Res Ability” of the ancestor of tag “select” will rise, which may arouse error. As **(Fig. 2)** shows, the correct area in inner-square was recognized as the one in outer-square. What’s more, information of the option make up the

range of value in tag "select". For the reasons above, it's necessary to merge the tag with dependency ability.

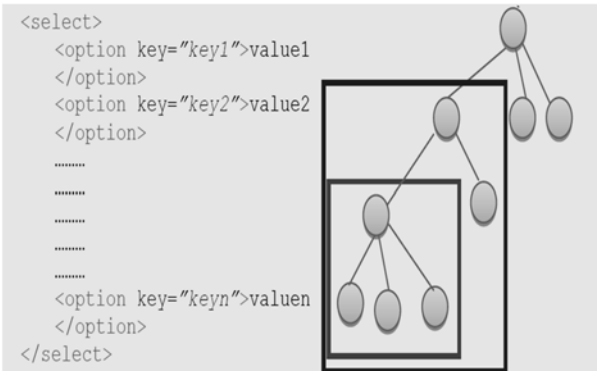

Fig.1 HTML Segmentation Fig.2 Example showing interact area are enlarged

Table-1 The Relationship between Dependent Node Merged and not Merged

Area (Surrounded by tag)	Res Ability (tag)		Relationship between merged and not
	Not merged	Merged	
Tag "select"	$\frac{n}{n}$	$\frac{1}{1}$	$\frac{n}{n}=\frac{1}{1}$
The ancestor tag of "select"	$\frac{n+x}{n+y}$	$\frac{1+x}{1+y}$	$\frac{n+x}{n+y}\geq\frac{1+x}{1+y}$
PS. Only if x=y, the equal is correct			

4. WQI METHOD BASED ON HIERARCHICAL CLUSTER AND CONTENT FILTER

As the experience in human-computer interaction, man can realize the interaction form from Web page at the first glance through obvious interaction elements (Phil, 2009). So interaction groups, whose interaction density is over certain threshold, are the breakthrough in the recognition of the WQI. Each interaction group got from pre-order traversal are regarded as a single class before clustering. After hierarchical clustering based on the distance in structure, the WQIs can be established with the help of content filter. And our method achieves satisfactory result in recognition accuracy and generality.

Interaction group distance in structure

Interaction group distance in structure is considered in two aspects: the relative distance and the close extent of two groups, so the definition of the distance is defined as $dist(n_1,n_2)$ in formula 1 presents the relative distance between group 1 and group 2, and their root are n1 and n2, respectively. "Close weight" presents the close extent of two groups and it is measured by the depth of nearest common parent of the two groups in Model "Tree". The relative distance is calculated as formula 2-5.

$$distLocal(n1,n2)=\frac{dist(n1,n2)}{closeWeight} \quad (1)$$

$$dist(n1,n2)=dist(dif_{n1},0)+dist(dif_{n2},1)+distSib(dif_{n1},dif_{n2}) \quad (2)$$

$$dist(dif_{n1,0})=\sum_{i=1}^{num-1}(numC_{dif_{n1}}+1-(index_{dif_{n1i}}+1)) \quad (3)$$

$$dist(dif_{n2},1)=\sum_{i=1}^{num-1}(index_{dif\_n2\_i}+1) \quad (4)$$

$$distSib(dif_{n1},dif_{n2})=index_{dif\ n20}-index_{dif\ n10}+1 \quad (5)$$

Formula 3 is composed of three parts, in which n1 and n2 are the roots of separate sub-trees, the nearest common parent of n1 and n2 is named as "com", is the path from "com" difn1to n1 anddifn2 is the path from "com" to n2. Relative distance between n1 and "com" is presented in formula 4. It's a sum of the relative distance of each node in the path difn1.The relative distance of node here is the distance between current node and its last sibling node. The relative distance between node n2 and node "com" is presented in formula 5. It's a sum of the relative distance of each node in the path difn2, and the relative distance here is the distance between current node and its first sibling node. And the third part of "dist(n1,n2)" is the distance between the first node in difn1 and the first node in difn2. For the example showed in (Fig.3), the distance between tag 1111 and tag 121 is calculated as dist (tag1111, tag121) = ((1+1-(0+1))+2+1-(0+1))+(0+1)+ (1-0+1) = 6.

Our Algorithm

Our algorithm is presented in this section. Firstly, interaction groups, whose interaction density is over threshold of 0.9, are detected through the pre-order traversal and they are regarded as the initial set to be clustered. Once meeting a node whose interaction density is over threshold in the pre-order traversal process, we should stop further traversal of its descendant nodes and put the sub-tree into the initial set. Otherwise, process mentioned above won't be stopped until the last leaf node of the sub-tree. After finishing pre-order traversal, we get an initial set composed of complete WQIs or maybe some segmented parts of WQIs. And the next step, we should restructure them into the way they originally are. For the parts of the WQI are linear close to each other in structure. So we decide to process them with hierarchical clustering. We calculate the distance between the adjacent local interfaces as formula 1 mentioned in section 4.1 and give the higher priority to the two ones similar in structure to be clustered. The distance between the new group and adjacent one keep still. The clustering process will be repeated until no distance between two

adjacent groups is small than the threshold of distance. In our method, the threshold of distance is 1.6, which is settled in experiment 5.1. Later, content filter is used to filter the non-query interfaces or elements, such as login interface, remark area, by the features of node in the group. Interaction groups filtered out by content filter will be processed in two strategies. To the ones containing interaction elements less than 3 are discarded as a whole. And to the others, the non-query elements in them will be trimmed out from groups and the groups processed are retained. The left set is the result of our method and each group in the set is a WQI found.

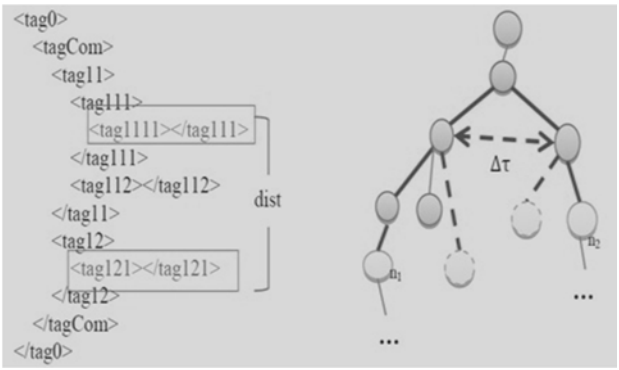

Fig.3 HTML Segmentation and its DOM Tree

5. EXPERIMENTS

This section shows the result of 2 experiments. In the first one, threshold for clustering is settled and the second one proves the effectiveness of our method. And our dataset is (Dataset TEL-8) provided by UIUC University.

Experiment 1

Experiment 1 aims at deciding the threshold of hierarchical cluster. Our experiment was conducted on the dataset of 21 pages with WQI(s) picked up randomly from TEL-8. And the result is illustrated in (Fig. 4), the X-axes presents the domain of clustering threshold which varies in the range of 1 and 3 with a step of 0.1 and the Y-axes, named as "coverage", presents the percentage of areas in WQI remarked to that in WQI found by our method (if WQI was not found in the page, then the coverage of this page is remarked as zero), and each line in different color shows the finding results of a page. The coverage of some pages remains zero when the threshold is below 1.5, for the reason that the originally-complete QWI is separated into multiple parts. When the threshold varies between 1.5 and 1.7, the coverage of different pages reach up to the top, approximate to 1, which means the areas actual, found is almost the same as the ones remarked. And coverage trends to fall with the increasing threshold over 1.7, for the reason that some other interaction elements are mistaken as the parts of this WQI. So the threshold in clustering is set as 1.6.

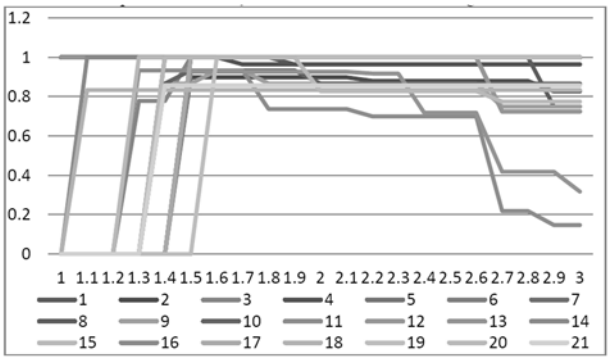

Fig.4 Relationship between Clustering Threshold and Capability of WQI Finding

Experiment 2

Two datasets are involved in experiment 2. The first one is TEL-8 which containing 477 WQIs from 8 different domains. We calculate the recognition rate of the WQI as the formula 6, in which INF presents the amount of WQI found by this method, and IN presents the total amount of WQI. Another dataset consists of 3 types of data. The first 50 pages containing WQI with tag "form" are picked up randomly from TEL-8 dataset, named as STEL8. Another 50 pages are crawled from web and they contain WQI without tag "form", named as SNTF. And last 50 pages are also crawled from web and they contain the non-query interface, such as login in, remarks, e-mail and so on, named as SNF. And the accuracy of result is calculated as formula 7, in which CIF presents the amount of WQI recognized correctly, CNIN presents the amount of non-query interface recognized correctly, and all pages presents the amount of pages in second dataset.

$$recog = \frac{INF}{IN} \times 100\% \quad (6)$$

$$accuracy = \frac{CIF + CNIF}{Allpage} \times 100\% \quad (7)$$

Referring to results in **(Table 2)** and **(Table 3)**, you can see the average accuracy of this method reach up to 90.7%, which has a better performance than early researches. Although our accuracy is lower 3.3% than Wang's, our method has an advantage in the generality over Wang's. As the results on the dataset organized by ourselves showed in **(Table 4)**, our method presents the same good recognition capability to the WQI without tag "form". Because our method make full use of the HTML features and avoids excessive dependency on tag "form". To some extent, our research makes a promotion in the research of WQI finding. We also checked the test cases that we fail to recognize. The reason is that there are more than one WQIs existing in the page and they are so close to each other in structure that even the threshold in clustering is unable to take effect normally.

Table- 2 Records of WQI Finding on the Dataset TEL-8

Type	No. page	No. WQI found	WQI Reg. rate %	Type	No. page	No. WQI found	WQI Reg. rate
Airfare	45	45	91.84	Hotel	39	35	89.74
Automobile	97	88	90.72	Job	52	48	88.46
Book	67	61	91.04	Movie	78	71	91.03
Car Rental	25	22	88.00	Music Record	70	63	90.00

Table-3 Accuracy Comparison

Author (year)	Characteristic	Recognition rate
Zhang etc (2004)	2P grammar	85%
Barbosa, Freire (2005)	Domain related and much human interface	90%
Wang etc(2011)	Tag form and text visible	94%
Method in this passage	Html features and hierarchal clustering	90.7%

Table -4 Records of WQI Finding on the Dataset Organized by Ourselves

	No. Page	**STEL8**	**SNTF**	**SNF**	**CTEL8**	**CSNTF**	**CSNF**	**Accuracy**
Dataset Organized	**150**	**50**	**50**	**50**	**45**	**46**	**47**	**92%**

6. CONCLUSIONS

According to the features of HTML, interactive elements exist in the terminal of DOM tree and they are close to each other in local area, we propose a method to find WQI through clustering interactive elements by the similarity in local structure, with the help of content-filter. Our method has a better performance on both TEL-8 and the organized dataset crawled from Web than traditional methods. What's more, it avoids the excessive dependence on tag "form" and performs a good generality than other methods. Next, we should consider more features about html, such as the division on visible page, to help WQI understanding and integrity.

REFERENCES:

Barbosa L., and J. Freire, (2005). Searching for Hidden-Web Databases. Proc. of the 8th ACM SIGMOD international workshop on web and databases. Baltimore, 1–6

Barbosa L., and J. Freire (2007) Combining Classifiers to Identify Online Databases Proc. of the 16th international conference on World Wide Web. New York, 107–118.

Dataset TEL-8, available on http://metaquerier.cs.uiuc.edu/repository/

Eduard C. D., K. Thomas, and Y. Clement, (2009). A Hierarchical Approach to Model Web Query Interfaces for Web Source Integration. VLDB, 2(1), 325–336.

Heidy, M., C. Marin, J. Victor, and S. Sosa, (2013). Automatic Discovery of Web Query Interfaces using Machine Learning Techniques. Journal of Intelligent Information Systems, 40(1), 85-108.

HTML Forms, available on http://www.w3chtml.com /html/form.html

Latiri, C. and S. B. Yahia, (2003). Query Expansion using Fuzzy Association Rules between Terms. In JIM Conference Journees Informatique Messine, Metz, France.

Lin, L., and L. Zhou, (2010). Web Database Schema Identification through Simple Query Interface. Resource Discovery Lecture Notes in Com. Sci., 6162(2): 18-34.

Liu, W., Meng, X. O, and M. Weiyi, (2007). A Survey on Deep Web Integration. Journal of Computer, 30(9), 1475-1489

Meng, X. O., (2001). A Survey on Web Data Management. Journal of Computer Research and Development, 38(4), 385-395.

Phil, T., and T. Susan, (2009). Practical Interaction Design. Proc. of the international conference, UK: BISL, 1-5. UK

Wang, Y., H. Li, and W. Zuo, (2011). Research on Discovering Deep Web Entries. Computer Science and Information Systems, 8(3), 779-799.

Zhang, Z., H. Bin, and C. C. Kevin, (2004). Understanding Web Query Interfaces: Best-Effort Parsing with Hidden Syntax. Proc. of the ACM SIGMOD International Conference on Management of Data. Paris, 107–118.