ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 4, April 2013

Extracting and Aligning the Data Using Tag Path Clustering and CTVS Method

J.KOWSALYA[#1], K.DEEPA [*2]

[#1]*PG Scholar ME-Software Engineering Final yea, Sri Ramakrishna Engineering College*
Coimbatore, Tamil Nadu, India

[*2]*Associate Professor*
Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India.

Abstract – **Web database generate query result page based on user's query. The information extracted automatically from query result page is used in many web applications. We present a novel method called Tag path clustering for record extraction from multiple attributes. It focuses on how a distinct tag path appears repeatedly in the DOM tree of the web document. It compares a pair of tag path occurrence patterns (called visual signal) to estimate how likely these two tag path represents the same list of objects. This paper introduces the similarity measure that captures how closely the signals appear and interleave. We propose a new record alignment that aligns the attribute in the record, first pairwise and then holistically using CTVS method (combining tag and value similarity).We introduce a new technique to handle the case when the non contiguous QRR, which may be due to the presence of auxiliary information such as, comments, recommendations or advertisement. The nested structure is handled by the nested structure processing method.**

Key words

Data extraction, data record alignment, multiple attribute.

I. INTRODUCTION

Information Extraction is the process of extracting information from the online databases. Online databases, called web databases, comprise the deep web. The pages in the deep web are dynamically generated in response to a user query submitted through the query interface of a deep web. A web database returns the relevant information based on user's query in either structured or semi structured, encoded in HTML pages.

This paper focuses on the problem of automatically extracting the data records in the query result page generated by web database and also handle the case where multiple data values from more than one attribute are clustered inside one leaf node of the tag tree. Our method focuses on how a distinct path appears repeatedly in the document. Instead of comparing a pair of individual sub trees in the data, we compare a pair of tag path occurrence patterns (called visual signal) to estimate how likely these two tag paths represent the same list of objects. We introduce a similarity measure that captures how closely the tag paths appear and how they interleave. We apply a tag path clustering based on similarity measure, and extract sets of tag paths that form the structure of the data records. A novel method is proposed to align the identified QRRs. First, pairwise, it can be aligned the data value belonging to the same attribute in the same column.

After all the pair of record aligned, a holistic alignment is performed to finding the connected components from the graph. A new nested structure processing algorithm is used to identify the nested structure in the QRRs.

II. QRR EXRACTION

A data region is a part of a web page that contains multiple data records of the same kind, which can be consecutive or non-consecutive. Instead of viewing the web page as a DOM tree, we consider it as a string of HTML tags. A data region maps to one or more segments of the string with repeating texture composed of HTML tags, which results in the visually repeating rendered on the web page. We aim to find the HTML tags that are element of the data region.

A. Visual Signal Extraction

The visual information rendered on a web page, such as fonts and layout, is conveyed by HTML tags. For each tag occurrence, there is an HTML tag path, containing an ordered sequence of ancestor nodes in the DOM tree. A web page can be viewed as a string of HTML tags, where only the opening position of each HTML tag is considered. Each HTML tag maps to an HTML tag path. Each tag path defines a visual pattern. An inverted index characterizing the mappings from HTML tags paths to their locations in the HTML document can be built for each web page shown in Table1. Each indexed term in the inverted index, i.e, one of the unique tag paths, is defined to be the visual signal.

Formally, a visual signal s_i is a triple $<p_i, S_i, O_i>$, where pi is a tag path, Si is a visual signal vector that represents occurrence position of pi in the document, and O_i represents individual occurrences. Si is a binary vector where $S_i(j)=1$ if p_i occurs in the HTML document at osition j and $S_i(j)=0$ otherwise. O_i is an ordinary list of occurrence.

Examples of visual signal vector are shown in the third column of Table 2. All of the visual signal vectors extracted from a web page have the same length, which is the total number of HTML tag occurrence in the web page. The vector representation $\{S_i\}$ of a web page is much simpler than the DOM tree representation. The visual signal vector represents how each atomic level visual pattern repeats in the web page. The visually repeating pattern in a web page involves multiple visual signals. These visual signals together form a certain repeating texture. Each texture corresponds to a data region that contains multiple data records of the same kind.

HTML code	Pos	Tag path
<html>	1	Html
<body>	2	html/body
<h1>Webpage</h1>	3	html/body/h1
<table>	4	html/body/table
<tr>	5	html/body/table/tr
<td>Cell#1</td>	6	html/body/table/tr /td
</tr>	NA	NA
<tr>	7	html/body/table/tr
<td>Cell #2</td>	8	html/body/table/tr /td
</tr></table></body>	NA	NA

Table 1: Finding tags path for HTML tags.

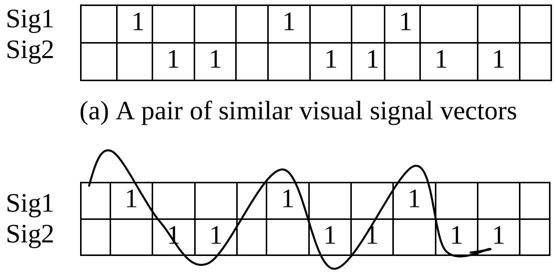
Detecting the visually repeating information is equivalent to identifying the set of visual signals with similar patterns that are element of the same data region. In other words, detecting visually repeating information is a clustering problem. To solve this problem we use the spectral clustering algorithm.

Unique tag path	Pos	Visual signal vector
Html	1	[1,0,0,0,0,0,0,0]
html/body	2	[0,1,0,0,0,0,0,0]
html/body/h1	3	[0,0,1,0,0,0,0,0]
html/body/ table	4	[0,0,0,1,0,0,0,0]
html/body/table/tr	5,7	[0,0,0,0,1,01,0]
html/body/table/tr/td	6,8	[0,0,0,0,0,1,0,1]

Table 2: Extracting visual signals from web page.

B. Similarity Measurement

The spectral clustering algorithm produces the results based on the pair wise similarity matrix calculated from the visual signals. In this paper, the similarity function captures how likely two visual signals belong to the same data region. Fig 1(a) shows a pair of visual signals that are highly likely to belong to the same data region. their positions are close to each other, and they interleave with each other. Every occurrence of visual signal 1 is followed by two occurrence of visual signal 2.



(a) A pair of similar visual signal vectors

(b) Segmented visual signal pair

Fig 2: Example pair of visual signals that appear regularly.

The distance between the center of gravity of two visual signals characterizes how close they appear. We call this measure the offset ω and calculate Equation (1)

$$\omega(S_i,S_j)=\left|\frac{\sum_{S_i(k)=1}k}{\sum S_i(k)}-\frac{\sum_{S_j(k)=1}k}{\sum S_j(k)}\right| \quad \text{….. (1)}$$

In Equation (1), S_i and S_j are two visual signals vectors and k€ {1,2,…,i} where l is the length of the visual signal vectors, and $S_i(k)$ is the kth element of $S_{i.}$

To capture the interleaving characteristic, we estimate how evenly one signal is divided by the other. We define a segment of Si is divided by S_j follows: a segment is a set of visual signal s_i between any pair of which there is no occurrence of visual signal s_j. Fig 2(b) illustrates how two signals divided each other. Let $D_{Si/Sj}$ be the occurrences counts in the segments of Si divided by Sj. We define interleaving measure in terms of the variances of counts in $D_{Si/Sj}$ and $D_{Sj/Si}$ in Equation (2).

$$\iota(S_i,S_j)=\max\left\{Var\left(D_{S_i/S_j}\right),Var\left(D_{S_j/S_i}\right)\right\} \quad \text{… (2)}$$

Both the offset measure and the interleaving measure yield non-negative real numbers. A smallest value of either measure indicates a high probability that the two visual signals come from the same data region. the similarity measure $\sigma(s_i,s_j)$ between two visual signals is inversely proportional to the product of these two measures and is defined by Equation (3).

$$\sigma(s_i,s_j)=\frac{\varepsilon}{\omega(s_i,s_j)\times\iota(s_i,s_j)+\varepsilon} \quad \text{…. (3)}$$

In Equation (3), ε is a non negative term that avoids dividing by 0 and that normalizes the similarity value so that it falls in to the range (0,1). Here ε =10. By Equation (3), we can calculate the similarity value of any pair of visual signals. The similarity measure captures the likelihood of two visual signals being from one data region.

C. Visual Signal Clustering

The pair wise similarity matrix can be fed in to a spectral clustering algorithm directly. We employ the normalized cut spectral clustering algorithm to produce the group of visual signals with similar pattern. A cluster containing n visual signals indicates that those n visual signals are from the same data region with high probability.

A data region contains multiple data records that use the same HTML code template, and a template typically has multiple Html tags that differentiate the data attributes. The size of a template is defined to be the number of unique HTML tag paths involved in the template. Thus, a template with size greater than n should corresponds to a cluster containing more than n visual signals. Given the fact that most HTML code templates contain more than three HTML tags that differentiate different data attributes, we assume that the smallest size of a template is three. Thus, we need to examine only the clusters containing three or more visual signals. We call these clusters called essential cluster of the document. Each cluster corresponds to one data region and contains a set of homogeneous data records.

D. Query Result Section Identification

After performing clustering, there may still be multiple data region in a query result page. However, we assume at most one QRR contains actual QRRs. Three heuristics are used to identify this data region, called query result section.

1. *The query result section usually occupies a large space in the query result page.* For each data region d, we calculated the area weight by d's area weight divided by largest area of identified data region.

2. *The query result section is usually located at the center of the query result page.* For this data region d, we calculated the center distance weight by smallest center distance among all identified region divided by d's center distance.

3. *Each QRR usually contains the raw data strings than the raw data strings in other section.* For each data region d, we calculate the value weigh by average number of raw data srings in data records of d divided by largest number of data values in all identified regions.

III. QRR ALIGNMENTS

QRR alignment is performed by a three novel step data alignment methods are pair wise alignment, holistic alignment and nested structure processing method that combines tag and value similarity.

A. Pairwise QRR alignment

The pair wise QRR alignment aligns the data value in the pair of QRR to provide the evidence for how the data value should be aligned among all QRR. It is based on the observation that the data values belonging to the same attribute usually has the same data type and contain similar strings. The pairwise alignment of the two QRR is performed to determine whether the paired value belong to the same attribute, according to calculated data value similarity. During the pairwise alignment, we require the data value alignment must satisfy the following constraints:

1. Same record path constraint. The record path of the data value f comprises the tag from the root of the record to the node that contains f in the tag tree of the query result page. Each pair of matched values should have the same tag path.

2. Unique constraint. Each data value can be aligned to at most one data value from the other QRR.

3. No cross alignment constraint. If one data value is matched with other data value then there should be no data value alignment between two data value.

Based on these constraints, a dynamic programming algorithm aligns the two records. It is obvious that the similarity is 0 if one of the QRR is empty. Otherwise, if the two data values have the same tag path, then only one of the following three data value alignments is possible.

$$\mathbf{L_{ij} = \max(L_{(i-1)(j-1)} + S_{ij}, L_{(i-1)j}, L_{i(j-1)})}$$

The alignment with the largest summing similarity score among these three alternatives is chosen.

Data value similarity calculation.

Given two data values f_1 and f_2 from different QRRs, we require their similarity,s_{12}, to be real value in [0,1]. The data value similarity is according to the data type. Each child node is subset of its parent node. The similarity between two data values f1 and f2 with data types nodes n1 and n2 defined as

$$\begin{cases} 0.5 & n_{21}=p(n_2)\& \ n_1!=\text{String OR} \\ & n_2=p(n_2)\& \ n_2!=\text{String} \\ 1 & n_1=n_2! = \text{String} \\ \text{Cosine similarity} & n_1=n_2 = \text{String} \\ 0 & \text{otherwise} \end{cases}$$

- 0.5, if they belong to different specific data types that have a common parent.
- 1, if they belong to same specific data type.
- Cosine similarity of f1 and f2, if both f1 and f2 belong to string data type
- 0 otherwise, which occurs when one of f1 and f2 belong to the string data type and other one belong to specific data type.

B. Holistic QRR alignment.

The holistic alignment performs globally among all QRRs to construct a table in which all data values of same attribute belong to the same column. Each data value in the QRR as vertex and pair wise alignment between

QRR two data value as an edge, pair wise alignment can be viewed as an undirected graph. Thus holistic alignment problem can be finding connected components in the undirected graph. There are two heuristics for our holistic alignment problem

1. Vertices from the same record are not allowed to be included in the same connected component as they are considered to come from two different attributes of the record.
2. Connected components are not allowed to intersect each other.

We design 3-step algorithm for holistic alignment problem. First, we transverse the graph once by a depth-first search to discover the preliminary connected components. At the same time we mark those components containing the breach paths. Next, we transverse the some components containing breach paths to remove some edges so as to break the breach path. Finally, we use the divide and conquer method to identify and split up the intersecting components to enforce the second constraint.

C. Nested structure processing

The nested structure processing identifies the nested structure exists in the QRRs. The holistic alignment constraints a data value in a QRR to be aligned to at most one data value from another QRR. If a QRR contains a nested structure such that an attribute has multiple values, then some of the value may not be aligned to any other values. Therefore, nested structure processing identifies the data valuesof the QRR that are generated by nested structure. In existing method, we have done only tag to identify nested structure so it may be incorrectly identify plain structure as nested one. To overcome this problem, CTVS uses both HTML tag and value to identify the nested structure.

IV. PERFORMANCE MEASUREMENT

The evaluation metrics used to compare the performance. The record level and includes the precision and recall metrics defined as

$$P_r=C_c/C_e$$

and

$$R_r=C_c/C_r$$

Where C_c is the count of correctly extracted and aligned QRRs, C_e is the count of extracted QRRs, and C_r is the actual count of QRRs in the query result pages

V. RESULTS AND TABLES

About	More	Like	Games	This
About	Like	More	This	Shareware
About	Part	Like	This	More
About	Games	More	Like	This
About	Part	Games	More	Part

Table 3: Pair wise QRR alignment

About	**Like**	**More**	**This**	**2004**	**Broad**
About	Like	More	This	2004	Broad
About	Like	More	This	2004	Broad
About	Like	More	This		Broad
About	Like	More	This		

Table 4: Holistic QRR Alignment

About	**Like**	**More**	**This**	**Broad**
About	Like	More	This	Broad
About	Like	More	This	Broad
About	Like	More	This	Broad
About	Like	More	This	

Table 5: Nested structure

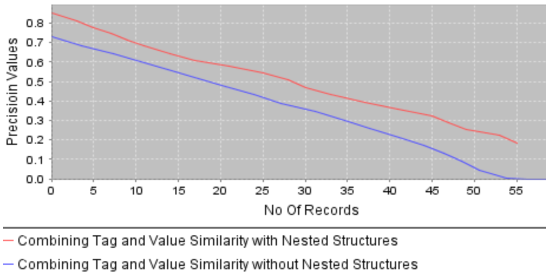


Fig 3: Record Level Precision graph

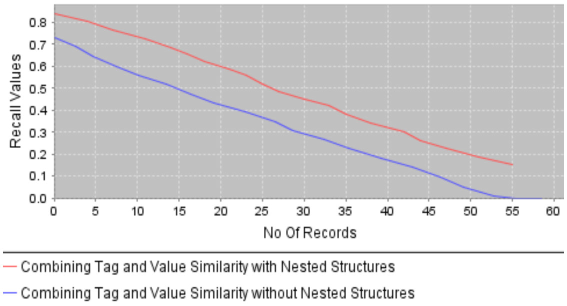


Fig 4: Record Level Recall graph

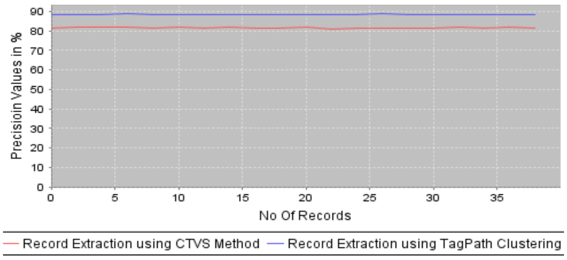


Fig 5: Precision level comparison graph

VII. CONCLUSION

Automatic data extraction from multiple databases is necessary for many web applications. The extracted query result pages contain some non contiguous QRR. This irrelevant information is removes by two step process called QRR extraction and aligning the QRR. Record extraction first detects the visually repeating information on a web page and then extracts the data record using tag path clustering. The notion of visual signal is introduced to simply the web page representation as set of binary visual signal vectors instead of a traditional DOM tree. Record alignment is done in CTVS method to extract data automatically from query result page. First, pair wise and then holistically align the data in the QRRs. Thus, CTVS automates the data extraction from multiple databases which supports many web applications. And also CTVS removes the nested structure using nested structure processing for accurate alignment.

VIII. REFERENCES

[1]. A. Arasu and H. Garcia-Molina "Extracting structured data from Web pages" In Proceedings of the 2003 ACM SIGMOD International Conference on the Management of Data, pages 337-348, 2003.

[2]. D. Buttler, L. Liu, and C. Pu ."A fully automated object extraction system for the World Wide Web "In Proceedings of the 21st IEEE International Conference on Distributed Computing Systems, pages 361-370, 2001

[3]. B.Liu and R. Grossman,"Mining Data Records in Web Pages", proc.9th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 601-606,2003.

[4]. A. Y. Ng, M. I. Jordan, and Y. Weiss" On spectral clustering: Analysis and an algorithm" In Proceedings of the Neural Information Processing Systems Conference, pages 849-856, 2001.

[5]. J. Shi and J. Malik" Normalized cuts and image segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8):888-905, 2000.

[6]. K.Simon and G.Lausen,"VIPER: Augmenting Automatic Information Extraction with Visual Perception", proc.14th ACM Int'l Conf. Information and Knowledge Management, pp. 381-388, 2005.

[7]. H.Snoussi and L.Magnin, "Heterogeneous Web Data Extraction Using Ontologies", proc.5th Int'l Conf. Agent Oriented Information Systems, pp 99=110, 2001.

[8]. J.Wang and F.H.Lochosky,"Data rich Section Extraction From HTML Pages", proc.3rd Int'l Conf. Web Information System Eng., 2002

[9]. Wang and F.H.Lochosky, "Data Extraction and Labeled Assignment for Web Database, proc.12th WWW conf,pp. 187-196,2003.

[10]. Y.Zhai and B.Liu, "Structured Data Extraction from the Web based on Partial Alignment", IEEE Trans. Knowledge and Data Eng, vol.18.12, pp.1614-1628, Dec 2006.

[11]. H.Zhao and Z.Wu, "Fully Automatic Wrapper Generation for Search Engines", proc. 14th WWW Conf.,pp. 66-75,2005.

Ms. J. Kowsalya received B.Tech in Computer Science and Engineering. She is doing her Master degree in Software Engineering. Her area of Interest includes Data Mining and Warehousing and Database.

Mrs.K.Deepa received the Master's Degree in Software Engineering in 2006. She has obtained Anna University 2nd rank in M.E., [Software Engineering]. She is currently working towards the Ph.D Degree in the Department of Computer Science and Engineering, Anna University Chennai. Her research interest includes Data Mining and Warehousing and Database Management Systems.