A Newfangled Template Extraction from Heterogeneous Web Pages using IEPAD

[1]A. Srilakshmi, [2]Dr. Ch. Satyanarayana, [3]N. Ranjana
[1,2]Dept. of Computer Science, JNTUK, Kakinada, AP, India
[3]Scientist 'F', DRDO (ASL), Hyderabad, AP, India

Abstract
Now-a-days the rapid expansion of the web is causing the constant growth of information, leading to several problems such as an increased difficulty of extracting potentially useful knowledge. We have many websites which consists of web pages having common template structures with contents. Templates are the HTML documents, and these documents may contain data that is irrelevant to the query. The structure of these documents may be volatile and this affects the extraction process. Domain knowledge about the data source is also embedded in HTML documents and must be extracted and in addition to answering queries, the wrapper will provide information. In this paper we consider all the web documents as one cluster and perform the partition of cluster based on the support count of the paths.We consider the size of the cluster based on the number of paths produced by the documents given as input. Therefore, there is a significant need of robust, flexible Information Extraction (IE) systems that transform the web pages into structured data ready for post processing. To overcome the drawbacks in existing system, we propose novel algorithm to improve the Efficiency, Accuracy and scalability of template extraction from heterogeneous web pages.

Keywords
Clustering, Minimum Description Length,Template Extraction, Information Extraction,Road Runner, IEPAD

I. Introduction

Due to the huge amount on the web and websites have becomes a key source of information and communication for some organizations, Hence it becomes imperative to use this data effectively to understand the structure of information on the web more precisely and deeply. We have different problems in web content mining. Some of the problems are Data/information extraction, Web information integration. Web document clustering has been extensively investigated as a methodologyfor improving document search and retrieval.In order to achieve high productivity of publishing, the Webpages in websites are populated by using common templates with contents. The template provides readers easy access to the information guided with consistent structures. For machines, the templates are harmful since they degrade the accuracy and performance of applications due to unnecessary contents in templates. Thus template extraction,Detection techniques have received lot of attention.

To overcome the disadvantages in Existing system we propose novel algorithm which increases the Accuracy, Efficiency and scalability. The problem using existing algorithms are [1]

- TEXT-MDL: It is not scalable to the large volume of data.
- TEXT-MAX, TEXT-HASH:We use length of signatures. However, the estimated MDL costs and the real MDL costs are quickly converged as the length of signature becomes longer. Which make algorithms slow without any significant improvement of accuracy.

There are three types of web mining techniques namely web content mining, web usage mining and web structured mining. Web content mining describes the automatic search of information resourceavailable online and involves mining web data content and Involves processing the web pages through clustering (locating documents that are similar to each other) and classification (assigning documents to predetermined classes). This is useful for both visualizing and browsing through the structure of a collection of web documents. There are two types of Web Content Mining respectively, (1) directly mining the content of documents and (2) improve on the content search of other tools like a search engine. There is a lot of data online that needs to be well organized and structured. To do that web content mining is the best mining technique. [2-3] the goal of Web structure mining is to generate structural summary about the Web site and Web page. Technically, Web content mining mainly focuses on the structure of inner-document, while Web structure mining tries to discover the link structure of the hyperlinks at the inter-document level. Based on the topology of the hyperlinks, Web structure mining will categorize the Web pages and generate the information, such as the similarity and relationship between different Web sites. Web structure mining can also have another direction that is discovering the structure of Web document itself. This type of structure mining can be used to reveal the structure of web pages. This would be good for navigation purpose and make it possible to compare/integrate Web page schemes. This type of structure mining will facilitate introducing database techniques for accessing information in Web pages by providing a reference schema.

To increase the scalability, accuracy and efficiency, we use novel algorithm in this paper. Since HTML document can be naturally represented with a Document Object Model (DOM) tree, web documents are considered as trees and many existing similarity measures for trees have been investigated for clustering [4-6]. However, clustering is very expensive with tree-related distance measures. For instance, tree-edit distance has at least O (n1n2) time complexity [7-8], where n1 and n2are the sizes of two DOM trees and the sizes of the trees are usually more than a thousand. Thus clustering on sampled web documents is used to practically handle a large number of web documents.

We have to manage unknown number of templates to increase the efficiency, accuracy and scalability of template extraction from heterogeneous web pages.. We consider templates as input. The page which start with <HTML> (start tag) to </HTML> (end tag) is called web page which is represented as a subset of the root-to-link paths in the corresponding DOM tree representation. We have taken four documents and their paths with respect to their support count.

- In our paper we perform top-down clustering algorithm to perform an unknown number of clusters.
- We extract template for each cluster.
- We give the cluster template as input to the IEPAD (Information Extraction Based on Pattern Discovery).

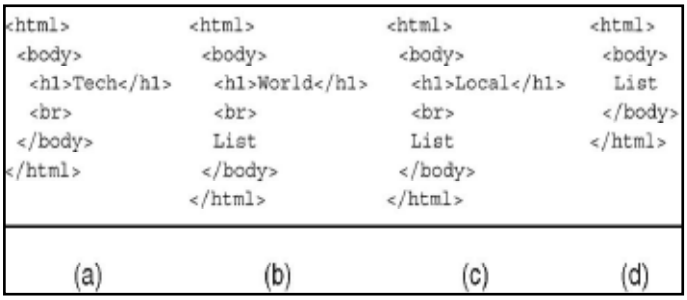

Fig. 1. Web documents. (a) Document d1. (b) Document d2. (c) Document d3. (d) Document d4.

Table 1: Paths of Tokens and Their Supports

ID	Path	Support
p_1	Document\⟨html⟩	4
p_2	Document\⟨html⟩\⟨body⟩	4
p_3	Document\⟨html⟩\⟨body⟩\⟨h1⟩	3
p_4	Document\⟨html⟩\⟨body⟩\⟨br⟩	3
p_5	Document\⟨html⟩\⟨body⟩\List	3
p_6	Document\⟨html⟩\⟨body⟩\⟨h1⟩\Tech	1
p_7	Document\⟨html⟩\⟨body⟩\⟨h1⟩\World	1
p_8	Document\⟨html⟩\⟨body⟩\⟨h1⟩\Local	1

II. Related Work

The issue of modeling the logical structure of Web sites for extracting purposes has been studied in several research projects. Many methods have been proposed for template extraction problem. A page let is determined by the number of hyperlinks in a HTML element. The page let whose frequency exceeds a threshold is identified as template. The template extraction problem can be categorized into two areas.

- Site-level template detection
- Page-level template detection

Detection oftemplatethrough site-level is decided based on several web pages from a web site.Before, we use only tags to find templates, but later we observed that any word can also be a part of the templates. So we used every word equally in our solution. However, they detect elements of a template by the frequencies of words but we consider the MDL principle as well as the frequencies to decide templates from heterogeneous webdocuments. The page-level template detection where the template is computed within a single document. The template extraction from heterogeneous webpages has two disadvantages. The first failure is, it consists only static web pages. The second failure is it extracts the entire site.

III.Preliminaries

A. Document Clustering

Throughout this paper,n denotes the number of documents in the collection, and kdenotes the desired number of clusters. In order to cluster documents one must first establish a pairwise measure of document similarity and then define a method to partition the collection into clusters of similar documents. Numerous document similarity measures have been proposed, all of which treat each document as a set of paths, often with their support count information, and measure the degree of paths overlap between documents [9].

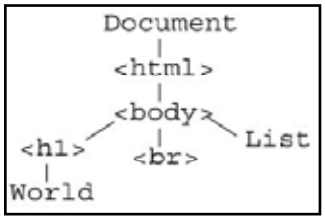

Fig. 2: DOM Tree for HTML Document

We already studied about the paths and supports for four documents in fig. 1. Based on the paths and support In Table 1, we perform document clustering.

B. Top-Down Clustering algorithm

The proposed algorithm first calculates the support count ofeach path (node) in the document. Finds the path with highest support count. It then inserts it into a cluster which is initially empty. Starting from the highest support count as seed node, the algorithm selects the node with highsupport count and inserts it into the cluster. Itkeeps doingthis until the cluster has the proper number of nodes. Andthen it checks the size of a cluster. If the cluster has more than k nodes, then the cluster should split further by calling the algorithm recursively. We again assume that the number n of paths in the input network is kcsuch that the hierarchical system is implemented with a full k-ary tree and c is a positive integer. Note that for a recursive call, The DOM tree H is a sub graph of the input graph G.P denotes the paths of the web document, E is the set of Edges connecting the paths in the Graph H.

Top-down Approach(H, N)
//Input:H= (P, E), where V= {1, 2, 3, …,N}
andE is the set of edges connecting the cells
//Output: C1, C2, … ,Ck
fori=1 to k do
1. Calculate the visit count of each cell in H
2. Select the cell x with the largest visit count
3. Insert x into Ciand remove x from H
4. while(the number of cells in Ci≤N/k)
5. Select the cell y that has the largest sum of the
weights of the edges between the cell and each of the cells
6. Insert y into Ciand remove y from H
forj=1 to k do // splitting clusters if their sizes >k
if(|Cj| >k) then Cj=Top-down Approach(Cj, N/k)
returnC1, C2, … , Ck;

C. Minimum Description Length Principle

In order to manage the unknown number of clusters and to select good partitioning from all possible partitions of HTML documents, we employ Rissanen's MDL principle [10], [11]. The MDL principle states that the best model inferred from a given set of data is the one which minimizes the sum of 1) the length of the model, in bits, and 2) the length of encoding of the data, in bits, when described with the help of the model.

D. Template Extraction

We give the clusters which are formed using Top-Down Clustering is given as input to the Road-Runner. Target of this research are the so-called data-intensiveWeb sites,HTML-based sites with large amountsof data and afairlyregular structure. Generating awrapper for a set of HTMLpages corresponds to inferringa grammar for the HTML code – usually a regulargrammar – and then use this grammar to parsethe page and extract pieces of data. Grammar inferenceis a well-known and extensively

studied problem. Asa consequence, the large body of research that originatedfrom Gold's seminal works has concentrated onthe development of efficient algorithms that work inthe presence of additional information (typically a setof labeled examples or a knowledgeable teacher'sresponsesto queries posed by the learner).We have to extract template for the clusters formed, so we used Roadrunner matching technique to compare the HTML documents which are present in clusters to form wrapper. It is based on a matching technique calledACME, for Align, Collapse under Mismatch, and Extract.There are essentially two kinds of mismatches that can be generated during the parsing: (a) String mismatches, i.e., mismatches that happen when different strings occur in corresponding positions of the wrapper and sample. (b) Tag mismatches, i.e., mismatches between different tags on the wrapper and the sample, or between one tag and one string. In the following paragraphs we discuss how mismatches can be solved, with the help of the simple example in Figure 3. At the end of this section we will generalize the technique and show how it can be applied to more complex examples that are closer to real sites.

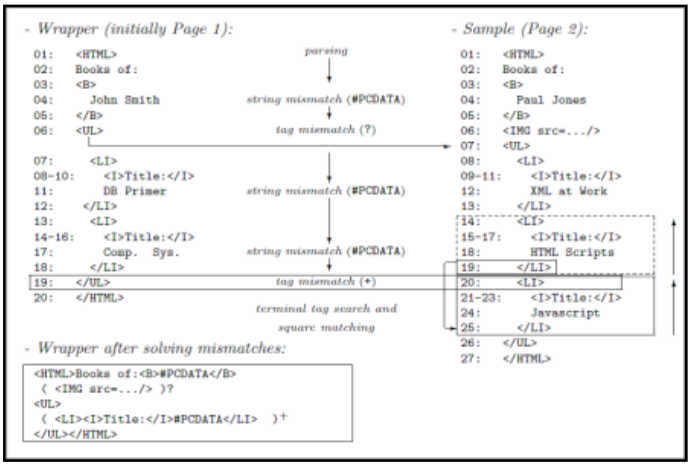

Fig. 3:

It can be seen that, if the two pages belong to the same class, String mismatches may be due only to different values of a database field. Therefore, these mismatches are used to discover fields (i.e., #PCDATA). Figure 3 shows several examples of string mismatches during the first steps of the parsing. Consider, for example, strings 'John Smith' and 'Paul Jones' at token 4. Which initially equals page 1, is generalized by replacing string 'John Smith' by #PCDATA. The same happens a few steps after for 'Database Primer' and 'XML at Work'. It is worth noting that constant strings in the two pages, like 'Books of:' at token 2, do not originate fields in the wrapper.Tag Mismatches: Discovering Optionals Tag mismatches are used to discover iterators and optionals. In the presence of such mismatches, our strategy consists in looking for repeated patterns as a first step, and then, if this attempt fails, in trying to identify an optional pattern. Let us first discuss how to look for optionals based on tag mismatches. Consider fig. 3. The first tag mismatch occurs at token 6, due to the presence of an image in the sample and not in the wrapper. This image should therefore be considered as optional.We may therefore assume that the tag mismatch is due to the presence of optionals. This means that, either on the wrapper or on the sample we have a piece of HTML code that is not present on the other side, and that, by skipping this piece of code; we should be able to resume the parsing. This is done in two main steps: [12]

- Optional Pattern Location by Cross–Search
- Wrapper Generalization

E. Extraction Rule Generator

After the template extraction. The template may consists of The system IEPAD includes three components, an extraction rule generator which accepts an input Web page, a graphical user interface, called pattern viewer, which shows repetitive patterns discovered, and an extractor module which extracts desired information from similar Web pages according to the extraction rule chosen by the user.When we submit an HTML web page to IEPAD, the translator will receive the HTML page and translate it into a string of abstract representations, referred to here as tokens. Each token is represented by a binary code of fixed length l. The PAT tree constructor receives the binary file to construct a PAT tree. The pattern discoverer then uses the PAT tree to discover repetitive patterns, called maximal repeats. The maximalrepeats are forwarded to validator, which filters out undesired patterns and produces candidate patterns.

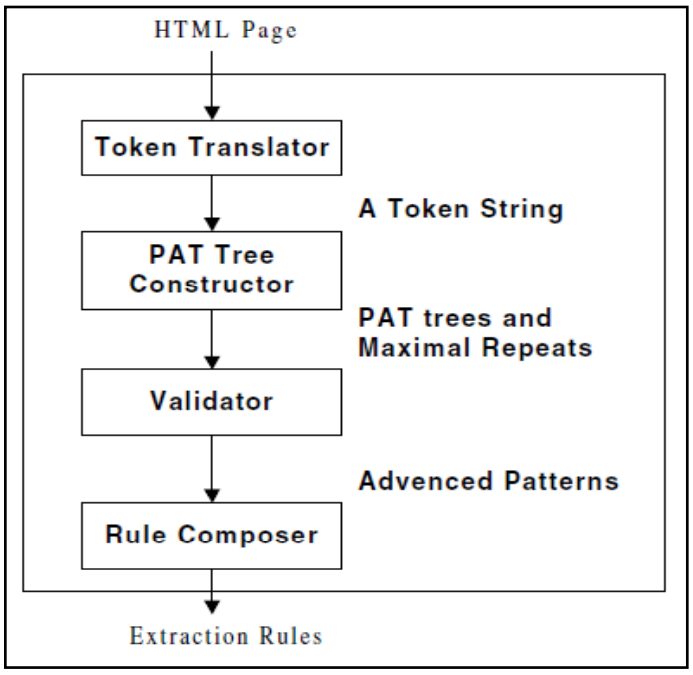

Fig. 4:

Finally, the rule composer revises each candidate pattern to form an extraction rule in regular expression.It is easy to detect repetitive for small code. However, a typical web page usually contains a large number of maximal repeats, not all of which contain useful information. To eliminate undesired maximal repeats, IEPAD [13] uses the validator to determine whether or not the maximal repeats contain useful information. In addition to the occurrence frequency and pattern length of a maximal repeat, the validator employs a number of criteria, including regularity, compactness, and coverage. In this way we can increase the performance of our paper.

IV. Experimental Results

All experiments were performed with the configurations Intel(R) Core(TM) 2 CPU 2.13GHz, 2 GB RAM, and the operation system platform is Microsoft Windows XP Professional (SP2).

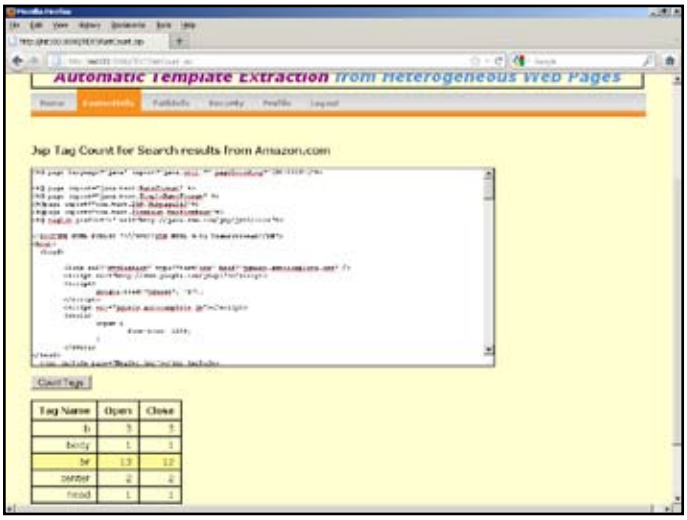

Fig. 5:
Two web page given to RoadRunner

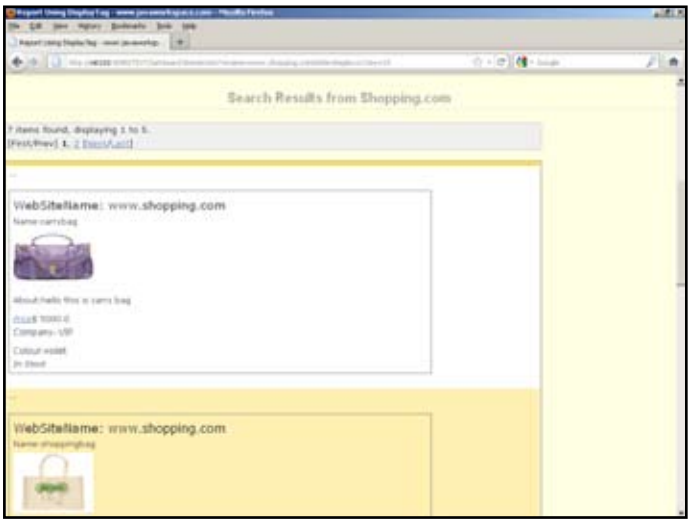

Fig. 6:

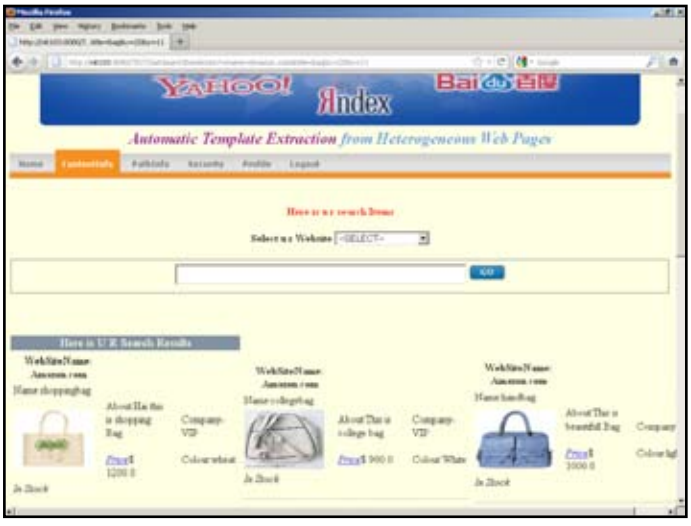

Fig. 7:

These are the input screenshot given to the Roadrunner and give the wrapper as input to IEPAD to remove repetitive patterns gives the better result.

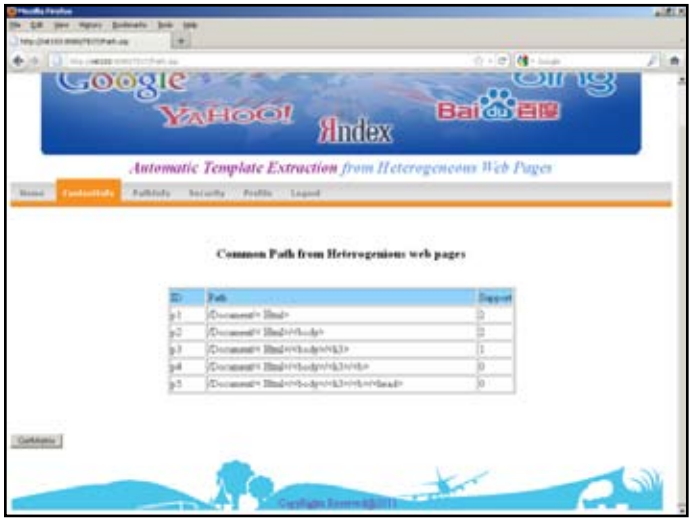

Fig. 8:

V. Conclusion

In this paper we introduced a novel algorithm Top-Down clustering to increase the accuracy and efficiency. We extract template by giving the HTML pages of the clustered document as input to the Roadrunner, Which generates the wrapper based on the similarities and differences as output. Therefore, there is a significant need of robust,flexible Information Extraction (IE) systems that transform the web pages into program friendly structures such as a relational database will become essential. IE produces structured data ready for post processing. Roadrunner will be used to extract information from template web pages. Then the resultant template is given to IEPAD to remove repetitive patterns which are not necessary. In our experiments, the extraction rule generalized from multiple string alignment can achieve 97% retrieval rate and 94% accuracy rate with high matching percentage.

References

[1] TEXT: Automatic Template Extraction and Kyuseok Shim, Member, IEEE Transaction data.And knowledge Engineering Vol. 23, No. 4, April 2011.

[2] [Online] Available: http://en.wikipedia.org/wiki/Web_crawler WebCrawler.

[3] [Online] Available: http://www.waset.org/journals/waset/v52/v52-54.pdf --- Bayesian Networks

[4] M. de Castro Reis, P.B. Golgher, A.S. da Silva, A.H.F. Laender,"Automatic Web News Extraction Using Tree Edit Distance", Proc. 13th Int'l Conf. World Wide Web (WWW), 2004.

[5] K. Vieira, A.S. da Silva, N. Pinto, E.S. de Moura, J.M.B. Cavalcanti, J. Freire,"A Fast and Robust Method for Web Page Template Detection and Removal", Proc. 15th ACM Int'l Conf. Information and Knowledge Management (CIKM), 2006.

[6] S. Zheng, D. Wu, R. Song, J.-R. Wen,"Joint Optimization of Wrapper Generation and Template Detection", Proc. ACM

[7] M. de Castro Reis, P.B. Golgher, A.S. da Silva, A.H.F. Laender,"Automatic Web News Extraction Using Tree Edit Distance".

[8] K. Vieira, A.S. da Silva, N. Pinto, E.S. de Moura, J.M.B. Cavalcanti, J. Freire,"A Fast and Robust Method for Web Page Template Detection and Removal", Proc. 15th ACM Int'l Conf. Information and Knowledge Management (CIKM), 2006.

[9] C.J. van Rijsbergen,"Information Retrieval", Butter-worths, London, second Edition, 1979
[10] J. Rissanen,“Modeling by Shortest Data Description”, Automatica, Vol. 14, pp. 465-471, 1978.
[11] J. Rissanen,"Stochastic Complexity in Statistical Inquiry", World Scientific, 1989.
[12] V. Crescenzi, G. Mecca, P. Merialdo,“Roadrunner: Towards Automatic Data Extraction from Large Web Sites”, Proc. 27th Int’l Conf. Very Large Data Bases (VLDB), 2001
[13] V. Crescenzi, G. Mecca, P. Merialdo,“Roadrunner: Towards Automatic Data Extraction from Large Web Sites”, Proc. 27th Int’l Conf. Very Large Data Bases (VLDB), 2001.

Srilakshmi Attuluri was born in Rajahmundry in 1989, and received the B-Tech degree in Computer Sceience Engineering from Chaitanya Institute of Engineering and Technology, Rajahmundry, Andhra Pradesh in 2010. She is working toward her M.Tech Post-graduate in Computer Science at JNTUkakinada in 2012. areas of interest in research are Data and Knowledge Engineering.