 ISSN: 2321-8169
1654 – 1656

Extracting Information from website using Site Style Tree

Pratiksha U. Jadhav
Department of Computer Engineering,
K..K..Wagh Institute of Engineering Education and Research,
University of Pune,
Nashik, India
pratiksha.jdhv@gmail.com

Sayali P. Badhan
Department of Computer Engineering,
K.K.Wagh Institute of Engineering Education and Research,
University of Pune, India
sayalibadhan@gmail.com

***Abstract*—** Now a days world wide web is a main source of information. A webpage generally contains large data along with navigation panels, advertisements, copyright and privacy notices. Except main data these other things generally does not contain any important information. These blocks can be called as non-informative blocks. As these blocks are non-informative, they can affect the result of web data mining. To avoid this it is important to separate the main data i.e.. informative blocks and non-informative blocks from the web page. In a website these non-informative blocks are generally present in different web pages and have same format. Also the data contained in these blocks is also same. In case of informative blocks, data contained by the block and their format are different. We need a structure at site level to capture the same format of the blocks and the data present in the blocks. DOM Tree structure is available at page level. Many tools are available to construct a DOM Tree of a webpage. But DOM Tree structure is not useful at site level. So we need to construct a Site Style Tree(SST) for a website. After analyzing this SST we can identify which part of SST is informative and which is non-informative. There is no tool available to construct a style tree for a given website. This work aims at constructing a style tree for given website and separating informative and non-informative blocks from the website.

***Keywords*—** *Informative blocks, Noise Detection, Non-informative blocks, Web mining.*

*****

I. INTRODUCTION

These days we can get any information on the world wide web. It is the main source of information. As large amount of information is available on web, it is very important that one should get useful or required information from the web. To provide user friendly environment; navigation bars, copyright notices are present along with main data. Also different types of advertisements are also included on website. To make the website attractive many decorative images are also included.

These all items are useful for viewers and necessary for website. Due to these items, retrieving required information from the web becomes very difficult. To improve the process of web data mining it is necessary to remove non-informative blocks from the web pages. For this we first need to identify such blocks from the website. This work aims at identifying such non-informative blocks from the website. These non-informative blocks can be removed to make the data mining more efficient.

Non-informative blocks generally share same contents and presentation style in multiple web pages. So to capture this at site level a structure called Site Style Tree (SST) is needed. Once the SST for a website is built, informative and non-informative parts can be easily identified. The example SST formation from [1] is shown below.

In Fig. 1. an example DOM tree D_1 of a web page is shown. Intermediate nodes in the DOM Tree represents different HTML tags from corresponding web page and leaf node contains the actual content from the web page.

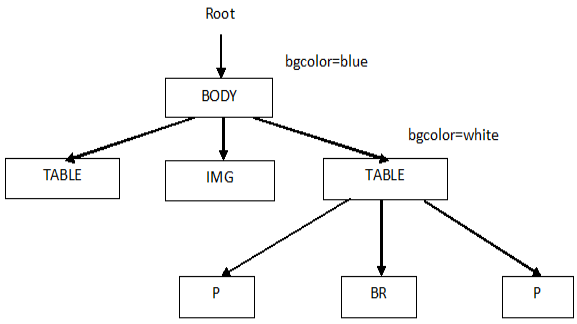

Figure 1. DOM Tree D_1

Fig. 2. shows another DOM tree D_2. All tags in D_1 has its corresponding tags in D_2 except the bottom level tags.

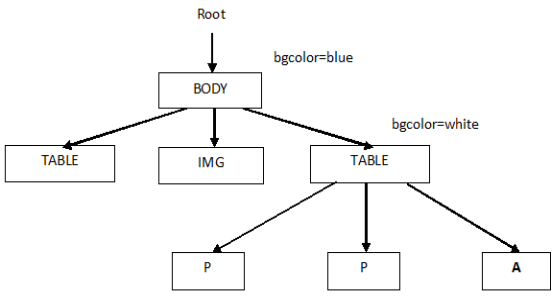

Figure 2. DOM Tree D_2

So these two DOM trees can be combined to generate a Style tree. The page count in incremented for the common nodes while merging. The resultant Style tree is shown on Fig. 3.

This style tree contains all the nodes from D_1 and D_2. There are two types of nodes in style tree, element nodes and style nodes. In Fig. 3. P-BR-P and P-P-A are two style nodes. Tag nodes in the style node are called element nodes. A count is maintained which indicates how many pages have same presentation style at that level.

From this style tree we can observe that two presentation styles are present under rightmost table tag.
Thus by applying informative measure we can identify non-informative blocks. To clean the website we can remove these non-informative parts. Also when the new pages are added in the website, that page can be mapped on the SST of that site.

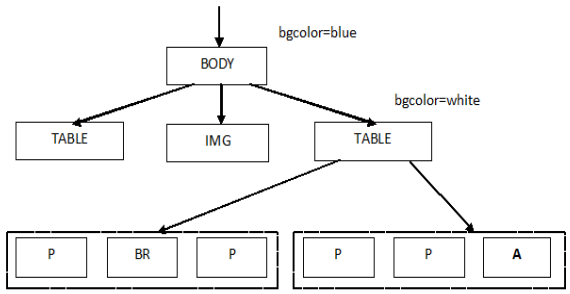

Figure 3. Style Tree

II. Literature Survey

In this section current work related to our work and different approaches to extract informative blocks are described.

A method is proposed in [4] to detect informative blocks from the news website. This work assumes that system already knows how webpage is partitioned and which blocks contains similar information from different web pages. But partitioning webpage and identifying corresponding blocks in different web pages are big issues. Also in [4] web page is considered as collection of blocks and each block as collection of words.
This is true in case of news website. Generally these assumptions are very strong.

Web page cleaning is considered as frequent template detection problem in [3]. In [3] webpage partitioning depends on the number of hyperlinks of an HTML element. This partitioning method is not useful in case of web pages from the same website.

In [8] some learning mechanisms are proposed. These helps to identify banner advertisements, redundant links of web pages. But these techniques require large training data set. These also require domain knowledge to generate classification rules. Some work includes duplicate records detection and cleaning of data for data mining .

In [6] one approach called block analyzer is mentioned. Using this approach blocks are preclustered. An entropy based value is assigned to each cluster. Using this value the blocks in the cluster can be classified as informative or non-informative blocks. But it may cluster some informative blocks into a noisy cluster.

III. System Architecture

In this section we represent the details of proposed approach. As shown in Fig. 4 a web page is downloaded from the website. Using HTML parser DOM tree of that page is generated. From this DOM tree tags which generally does not contain any information e. g. script tag are filtered. Now other web pages from the websites are taken as input. Their resultant DOM tree is generated in the same way as mentioned above.

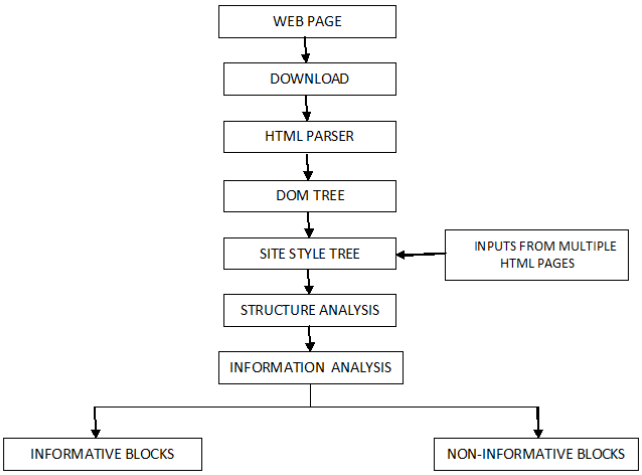

Figure 4. System Architecture

These DOM trees are then merged one by one with the previously obtained DOM tree. The result of this merging is the final Site Style Tree. Now structure of this style tree is analyzed. Also some informative measures are applied on the site style tree to identify informative and non-informative parts.

To identify which part from the site style tree are informative there is a need to calculate importance values of the different nodes from the site style tree. From [2], importance of a node E, is calculated as,

$$\begin{aligned} \text{NodeImp}(E) &= -\sum_{i=1 \text{ to } l} p_i \log_m p_i \quad \text{if } m>1 \\ &= 1 \quad \text{if } m=1 \end{aligned} \tag{1}$$

Here, m is the number of pages containing E and l is the number of child style nodes of E (i.e., l = |E.Ss|). Also p_i is the probability that a Web page uses the i^{th} style node in E.Ss.

For internal nodes, importance of node is affected by its descendent nodes. So there is a need to compute combined importance. Lets call it as composite importance. From [2],

$$\begin{aligned} \text{CompImp}(E) = (1-\gamma^l)\text{NodeImp}(E) + \\ \gamma^l \sum_{i=1 \text{ to } l} (p_i \text{CompImp}(S_i))) \end{aligned} \tag{2}$$

Here pi is the probability that E has the i^{th} child style node in E.Ss and γ is the attenuating factor.

From [2], style node's composite importance is calculated as,

$$\text{CompImp}(S_i) = \sum_{j=1 \text{ to } k} \text{CompImp}(E_j)/k \tag{3}$$

where E_j is an element node in $S_i.E$, and $k = |S_i.Es|$, which is the number of element nodes in S_i.

For a leaf element node E, from [2],

$$\text{CompImp}(E) = 1 - ((\sum_{i=1 \text{ to } l} H(a_i))/l) \text{ if } m>1$$
$$= 1 \text{ if } 1 \quad (4)$$

Here, l is the number of features (i.e., words, image files, link references, etc) appeared in E and m is the number of pages containing E. Also a_i is an actual feature of the content in E. $H(a_i)$ is the information entropy of a_i within the context of E,

$$H(a_i) = -\sum_{j=1 \text{ to } m} p_{ij} \log_m p_{ij} \quad (5)$$

where p_{ij} is the probability that a_i appears in E of page j.

After calculating importance values from above mentioned equations noisy elements are identified.

The overall algorithm of the proposed approach is given as follows,

Input: An URL having complete address of a website.
Output: Separated informative part from the website.

1. Download the web page from the website.
2. Build DOM Tree for that web page.
3. Crawl remaining pages from the web site
4. Build DOM Trees for these pages
5. Merge these DOM trees with the one by one with previously generated DOM tree of a web page
6. Build SST from DOM Trees
7. Calculate importance values for different nodes
8. Find Noisy Elements
9. For new pages map its DOM Tree to SST
10. Separate noisy and non-noisy elements

IV. Conclusion And Future Work

In this paper, a Site Style Tree based approach is proposed to extract informative i.e. non-noisy parts from different web pages of a website. This site style tree is created using DOM trees of different web pages. To improve web data mining, web pages should be clean. Thus using proposed approach web data mining can be improved.. After removing non-informative blocks the storage space and time for a webpage can be saved.

Research shows that such style tree based approach can be used for efficient data extraction.

References

[1] R.Gunasundari, S.Karthikeyan, "Removing Non-informative Blocks from the Web Pages", Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference.
[2] B. Liu, K. Zhao, and L. Yi, "Eliminating Noisy Information in Web Pages for Data Mining" , Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery an Data Mining, pp. 296-305, 2003.
[3] Bar-Yossef, Z. and Rajagopalan, S., "Template Detection via Data Mining and its Applications", WWW 2002, 2002.
[4] Shian-Hua Lin and Jan-Ming Ho., " Discovering Informative Content Blocks from Web Documents", KDD-02, 2002.
[5] S. Debnath, P. Mitra, and C.L. Giles, N.Pal "Automatic Identification of informative sections of Web Pages" , IEEE Transaction on Knowledge and Data Engineering , 2005.
[6] Chia-Hsin Huang, Po-Yi Yen, Yi-Chan Hung, Tyng-Ruey Chuang, and Hahn-Ming Lee, "Enhancing Entropy-based Informative Block Identification Using Block Preclustering Technology ", 2006 IEEE International Conference on Systems, Man, and Cybernetics October 8-11, 2006, Taipei, Taiwan
[7] Hung-Yu Kao, Jan-Ming Ho, Member, IEEE, and Ming-Syan Chen, Fellow, IEEE, " WISDOM: Web Intrapage Informative Structure Mining Based on Document Object Model", IEEE transaction on Knowledge and Data Engineering, VOL. 17, NO. 5, MAY 2005.
[8] Jushmerick, N., "Learning to remove Internet advertisements", AGENT-99, 1999.
[9] Yao, Z. and Choi, B., 2007. "Clustering Web Pages into Hierarchical Categories," International Journal of Intelligent Information Technologies, Special Issue on Web Mining, Vol. 3, No. 2, pp.17-35.
[10] Peng, X. and Choi, B., 2005. "Document Classifications Based on Word Semantic Hierarchies," The IASTED International Conference on Artificial Intelligence and Applications, pp.362-367.