International Journal of Research in Advent Technology, Vol.2, No.4, April 2014
E-ISSN: 2321-9637

Primary Content Extraction Based On DOM

Ms. Pranjali G. Gondse
M.E.(CSE) Second Year
H.V.P.M.C.O.E.T., Amravati
Email id: pgondse@gmail.com

Prof. Anjali B. Raut
Associate Proffessor(CSE Deptt.)
H.V.P.M.C.O.E.T., Amravati
Email id: anjali_dahake@rediffmail.com

Abstract: As we know internet web pages contains information which are not considered as primary content or informative content such as advertisements, headers, footers, navigation links and copyright information. User is interested only in the informative contents and not in non-informative content blocks. To extract the informative content of the web page correctly, the informative content and the non-informative content of the web page must be known clearly. The informative content is the main content of the web page that gives some information to the user e.g., articles about technology, health or education, etc. The non-informative content of the web page contains fixed description noise such as site logos, copyright notices, privacy statements, etc., service noise, irrelevant services, such as the weather, stock or market index, etc., navigational links, advertisements, header, footer and so on. To distinguish between the informative and non-informative content in a web page, it needs to segment the web page into semantic blocks.

Keywords: DOM Tree, information extraction, web mining;

1. INTRODUCTION

Today internet has made the life of human dependent on it. Almost everything and anything can be searched on net. It delivers the information mainly in the form of the Web pages. However, useful information is often accompanied by a large amount of noises. Almost all web pages on the Internet contain noises irrelevant to the main content, such as navigation bar, copyright information, survey or feedback, questionnaire etc. These noises affect the efficiency of algorithms for web page classification, clustering, information extraction and searching although they could be useful for other purposes, such as to ease browsing the web pages. It is important to distinguish the informative blocks from the noisy blocks. The rapid expansion of the Internet has made the WWW a popular place for disseminating and collecting information. Extracting useful information from Web pages thus becomes an important task. Usually, apart from the main content blocks, web pages usually have such blocks as navigation bars, copyright and privacy notices, relevant hyperlinks, and advertisements, which are called Non-informative blocks. These blocks are not relevant to the main content of the page. These items are required for web site owners but they will hamper the web data mining and decrease performance of the search engines. These blocks are very common in web pages.

Major efforts have been made in order to provide efficient access to relevant information within the web pages. Today's Web pages are commonly made up of more than merely one cohesive block of information. So extracting exact information content becomes difficult.

Efficiently extracting high-quality content from Web page is crucial for many Web applications such as information retrieval, automatic text categorization, topic tracking, machine translation, abstract summary, helping end users to access the Web easily over constrained devices like PDAs and cellular phones. The extracted results will be the basic data for the further analysis. So content extraction from Web page has attracted many researchers recently.

The advantage of identifying non-content blocks from web pages is that if user does not want non-content blocks these can be deleted. These non-content blocks are normally large part of the web pages so eliminating them will be a saving in storage and indexing.

2. RELATED WORK

To identify Informative content from web page is relatively easy task for human being because he can easily identify important content by visual inspection but it is difficult task for computer.

The Feature Extractor (FE) algorithm by Debnath et al[8] is a content extraction algorithm which based on DOM block structures. The algorithm segments a web document into blocks and selects certain blocks to be extracted. A block here corresponds to the DOM sub tree nodes. The algorithm will start working from the root node and recursively splitting the document into blocks. They defined a set of HTML tags which denotes a block namely table, tr, hr, and ul. FE uses the feature such as the presence of nested blocks, texts, images, applets, or contained JavaScript code.

FE will extract the blocks which is dominant in certain features. In the context of content extraction, for example, we can set the feature we need it's the text properties.

As a result the blocks that will be extracted will be those which is rich with text.

Rahman et al[2]. propose another technique that uses structural analysis, contextual analysis, and summarization. The structure of an HTML document is first analyzed and then properly decomposed into smaller subsections. The content of the individual sections is then extracted and summarized. However, this proposal has yet to be implemented.

Gupta et al[7] .developed a program for content extraction known as Crunch. Instead of using raw HTML text, it uses the DOM tree representation of a web document. Receiving input of a HTML page, Crunch will parse the HTML string, construct the DOM, traverse the nodes recursively and filter out the non-informative content behind. Each of the filters can be turned on or off and customized to certain degree.

Lin and Ho[4] proposed an extraction method based on information entropy, the web page is divided into content block according table tag, each block has entropy, and then information blocks are obtained by comparing with threshold value. But this method just applies to web pages which contain table tags, while increases the complexity of the algorithm.

3. PROPOSED WORK

Proposed approach concentrates on web pages where the underlying information is unstructured text. The technique used for information extraction is applied on entire web pages, whereas they actually seek information only from primary content blocks of the web pages.

The user specifies his required information to the system. Web crawlers download web pages by starting from one or more seed URLs, downloading each of the associated pages, extracting the hyperlink URLs contained therein, and recursively downloading those pages. Therefore, any web crawler needs to keep track both of the URLs that are to be downloaded, as well as those that have already been downloaded.

DOM analyzer defines the concept of blocks in web pages. Most web pages on the internet are still written in HTML. Even dynamically generated pages are mostly written with HTML tags, complying with the SGML format. The layouts of these SGML documents follow the Document Object Model tree structure of the World Wide Web Consortium.2. The relevant pages given out by the web crawler are represented in a form of DOM tree HTML DOM is in a tree structure, usually called an HTML DOM tree. Following Figure illustrates a simple HTML document and its corresponding DOM tree. We are interested only in the <BODY> node and its offspring. In this example, <BODY> node has three children: element nodes <B> and <I>, and text node #and. Element node <B> has a text node child #bold, and element node <I> has a text node #italic. Following the DOM convention, we use <> to indicate element node, and use # to indicate text node.

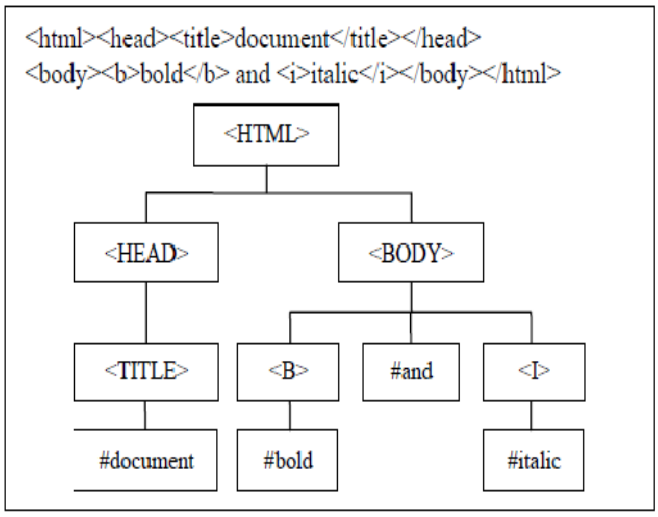

Figure .Simple HTML document and its corresponding DOM tree.

Steps for content Extraction:-

Step 1: Cleaning the HTML page:

a. Symbols, ”<” and ”>”, should only contain html tags. When used in other place, they should be replaced by ”&lt” and ”&gt” respectively.
b. All tags must be matched, i.e. every starting tag has a corresponding endin tag.
c. Attributes of all tags must be encircled by quotation marks.
d. All tags must be nested correctly. For example, <a>. . .. . is a correct nest, while <a>. . .. . is incorrect.

Step 2: Preprocessing the web page tags.

All tags on the page form a tree structure. Those nodes that do not contain any text should be removed, as well as invalid tags such as<script> <style> <form> <marquee> <meta> etc, which are unrelated to the content. Then the structure tree is built.

Step 3: Judging the location of content

The aim of this process is to select the optimum node containing content. If a node is not satisfied with this condition, the text under this node is not identified.

Step 4: Extracting the content

The content is extracted by tools such as html parser. If the node is not satisfied with the conditions, return the step 3 in order to find the optimal nodes of the next level nodes (the child nodes of the node).

Step 5: Adjusting the extraction results from step 4

In step 3, only the node that most likely contains the content is selected. But if the structure of a web page is relatively decentralized, it is very prone to extract a section or a paragraph of the whole content. As the adjacent nodes on the same level are free of judge, in this step, we must adjust the above result. The text also should be extracted from the adjacent nodes that meet the conditions of the precise content extraction. So all text will be extracted from the qualified nodes on the same level.

4. CONCLUSION

In this paper I have proposed a method which gives the informative content to the user. Using DOM tree approach contents of the web pages are extracted by filtering out non informative content.

With the Document Object Model, programmers can build documents, navigate their structure, and add, modify, or delete elements and content. With this features it becomes easier to extract the useful content from a large number of web pages.

REFERENCE:

[1] Jae-Woo LEE "A Model for Information Retrieval Agent System Based on Keywords JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN COMPUTER ENGINEERING ISSN: 0975 – 6760| NOV 12 TO OCT 13 | VOLUME – 02, ISSUE – 02 Page 297 Distribution" International Conference on Multimedia and Ubiquitous Engineering(MUE'07), IEEE 2007 0-7695-2777-9/07

[2] A. F. R. Rahman, H. Alam and R. Hartono "Content Extraction from HTML Documents"

[3] Wolfgang Reichl, Bob Carpenter, Jennifer Chu-Carroll, Wu Chou "Language Modeling for Content Extraction in Human-Computer Dialogues".

[4]S.-H. Lin and J.-M. Ho, "Discovering informative content blocks from web documents," in KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. New York, NY, USA: ACM, 2002, pp. 588–593.

[5] Jinbeom Kang, Joongmin Choi, "Detecting Informative Web Page Blocks for Efficient Information Extraction Using Visual Block Segmentation", International Symposium on Information Technology Convergence, pp 306-310, November 2007.

[6]Shian-Hua Lin, Jan-Ming Ho, "Discovering informative content blocks from Web documents", Proceedings of ACM SIGKDD'02, July 2002.

[7]David Neistadt Suhit Gupta, Gail Kaiser and Peter Grimm, Dom-based content extraction of html documents, Proceedings of the 12th International Conference on World Wide Web, 2003, pp. 207–214.

[8]C. Lee Gilles Sandip Debnath, Prasenjit Mitra, Automatic extraction of informative blocks from web pages, Proceedings of the 2005 ACM Symposium on Applied Computing, 2005, pp. 1722–1726.

[9] Thomas Gottron, Content extraction: Identifying the main content in html documents, Ph.D. thesis, Johannes Gutenberg-Universitt Mainz, 2008.

[10] Ryan Coleman-W. Bruce Croft Matthew King Wei Li David Pinto, Michael Branstein and Xing Wei, Quasm: a system for question answering using semi-structured data, In JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, 2002, pp. 46{55.