ELIMINATING NOISE INFORMATION FROM HTML WEB PAGES

Thandar Win, Khin Nwe Ni Tun
University of Computer Studies, Yangon, Myanmar
thandarwin77@gmail.com, knntun@gmail.com

ABSTRACT

Most of the commercial Web pages contain not only useful information but also noise information such as search and filtering panels, navigational bars, copyright and privacy notice, directory list and advertisement, etc. This noise information cannot give the correct result for the Web user to extract the main content information. This paper presents an effective technique for eliminating the noise data from HTML Web Page. We specify the rules which are noise in HTML Web page and eliminate the noise based on the proposed algorithm. Most of the existing techniques mainly based on the whole Web site not on the Web page. The proposed system mainly focuses on the Web page and tested on a large number of Web pages from the various commercial sites.

Index Terms— HTML Web page, HTML tags, DOM tree, information extraction

1. INTRODUCTION

Nowadays, the World Wide Web is a valuable resource of information which provides people to use the Web more conveniently. A huge amount of data and information is published in numerous Web pages as with the rapid growth of the Web. In the Web application, people want to get the main content information of the Web page without noise information. Noise means it is not useful information for the user. For example, Web page contains noise such as advertisements, navigational links, copyright and privacy notices, etc. These noises disturb the user to get the useful information from the Web page.

The information in a Web page is not equally important [7]. Most people are not interested in the advertisement or the advertisement list and the copyright notice when they browse a Web page. Different information in Web page has different importance weight according to its content, location and occupied area, etc. People can not easily get the main content information because of wanted and unwanted information are mixed together inside the Web page.

In [6] describes Web noises can be grouped into two categories which are global and local (intra-page) noises. Global noises include mirror site, legal/illegal duplicated Web pages, old versioned Web pages, etc. Local noises include banner advertisements, navigational guides, and decoration pictures, etc.

In the previous research in this area uses Web site for eliminating local noise. This paper only focuses on Web pages for eliminating local noise. Eliminating the noise information form Web pages before extraction of the main content becomes improving the performance of extraction results.

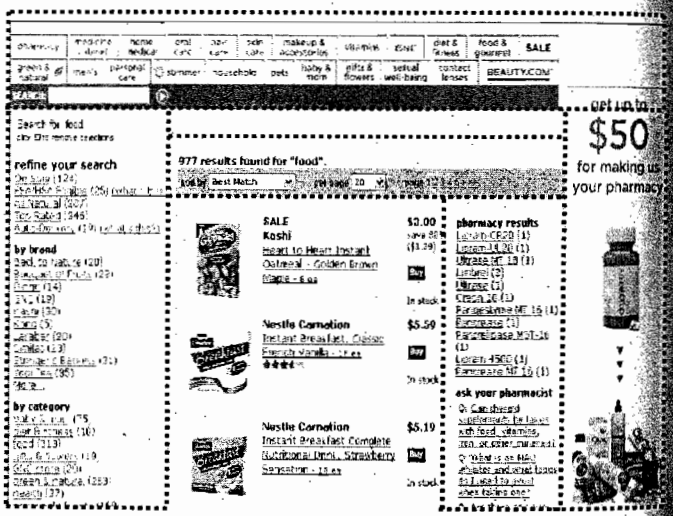

Figure 1. A page segment containing noise information

Figure 1 shows a segment of HTML Web page containing noise information from drugstore site. Noise information in this page such as navigation panel, search and filtering panel, directory list and advertisement are drawn in dotted lines. When examining the HTML Web page, noise information is located in top, left, right and bottom of the page. Mostly, search and filtering panel is located on top of the page, directory list is located in left and right of the page and privacy and copy right notices are located in bottom of the page. Therefore, the main content information is inside the noise information. Because of this noise, we cannot get the correct information when extracting the main content. For improving the performance of information extraction, we need to eliminate this noise information before extraction.

In this paper, we propose a new approach called eliminating the noise in Web page based on the proposed rules. Depending on the nature of the HTML Web pages, we defined the rules which are noise information in Web page. The proposed system works in five steps: (1) tidy the page (2) construct the DOM tree (3) mine the noise data (4) clean the noise data and (5) show the cleaning data of the Web page.

The rest of the paper is organized as follows: An overview on the related work is described in section 2. In section 3, we described the noise information in HTML Web page. The system architecture is presented in section 4. Our system performance result is shown in section 5. Section 6 summarizes our contribution and concludes the paper.

2. RELATED WORK

There is few number of research has been done in this area, although noise elimination in Web pages is important. In [5] [6], the main idea of the approach is that in a given Web site, noisy blocks usually share common contents or presentation styles, while the main content blocks of the pages are often diverse in their actual content and presentation styles. Based on the ideas, first create a Style Tree to represent both the layout and content of the page. Node importance is defined as the entropy of the node in the whole Style Tree for a site. By mapping a page of this site to the Site Style Tree (SST), noisy information in the page is detected and cleaned. This mapping method is not efficient for many Web pages from different Web sites because it need to map a page into SST every time to detect and clean the noise.

A DOM-based content extraction method has been proposed in [9] to facilitate information access over constrained devices like PDAs. They implemented an advertisement remover by maintaining a list of advertiser hosts, and a link list remover based on the ratio of the number of links and non-linked words is greater than a specified threshold. They also implemented a table remover removes tables if it has no substance. In this way they remove all the unnecessary information and output the plain text. Hence, different methods are required to get the plain text in PDAs.

In [10] proposed Information Discoverer approach, partitions a Web page into several content blocks according to TABLE tags. Terms are extracted as features and entropy is calculated for each term and block entropy is calculated accordingly. Dynamically select the entropy threshold value to decide whether a block is informative or redundant. This approach is only useful for TABLE tags Web pages.

Web page cleaning is defined as a frequent template detection problem in [11]. They proposed a frequent based tree mining algorithm to detect templates and views those templates as noises. The partitioning of a Web page is pre-fixed by considering the number of hyperlinks that an HTML element has. This partitioning method is simple and useful for a set of Web pages from different Web sites, while it is not suitable for Web pages that are all from the same Web site.

3. NOISE INFORMATION IN HTML WEB PAGE

Noise information in HTML Web page is based on the following observations:

(1) Directory list, navigational panel are typically presented in a contiguous region of a page and are formatted using the same HTML tag. In Figure 2, the dotted-lined box is the directory list. In this case, the same tag nodes are using in one contiguous region in a page and are formatted using almost the same sequence of "**a**" tags. Therefore text within this tag is called noise information for main content extraction.

(2) The next observation is advertisement in a Web page is presented one of the regions of a page and is formatted using different HTML tags. Texts within different tag nodes are called noise information for Web page. In Figure 3, the dotted-lined box is the advertisements and used the different tag nodes such as "**img**", "**a**" tags. We called noise information which is within different tags.

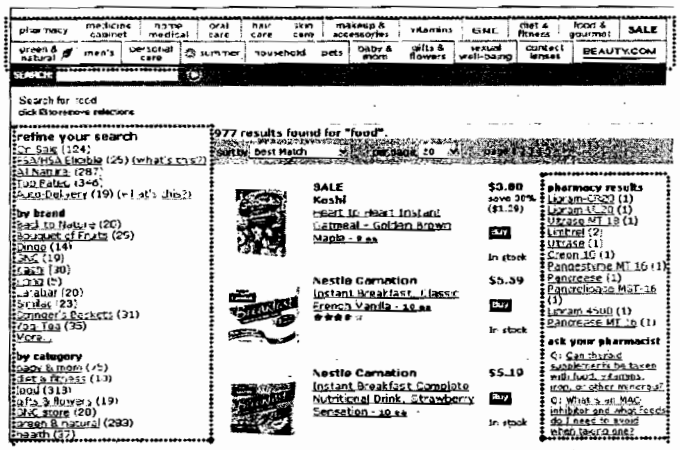

Figure 2. A page segment of directory list

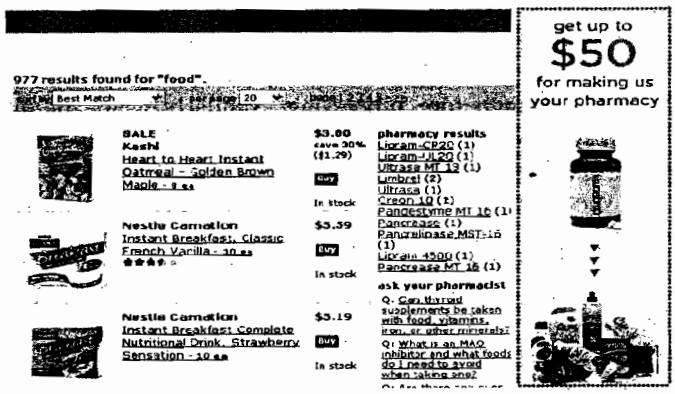

Figure 3. A page segment of advertisement

4. SYSTEM ARCHITECTURE

This section presents the detail of the approach of the proposed system which consists of five steps. As input, our system accepts the commercial HTML Web pages form any Web site. These pages contain product information as well as unwanted information such as product advertisement, pop-up, navigation bar, and directory list, etc are noise information. This noise information can not give the correct result for information extraction. The proposed system intends to eliminate this noise information from the pages.

In Figure 4, we present the process of eliminating the noise form HTML Web page. Firstly, we accept the HTML page and then we tidied the HTML Web page. According to the tidied page, we construct a DOM tree. Based on the DOM tree, we mine the noise data and eliminate the noise nodes by using noise nodes elimination algorithm. Finally, we show the cleaning data of Web page. The following subsection describes the detail process of our proposed cleaning process.

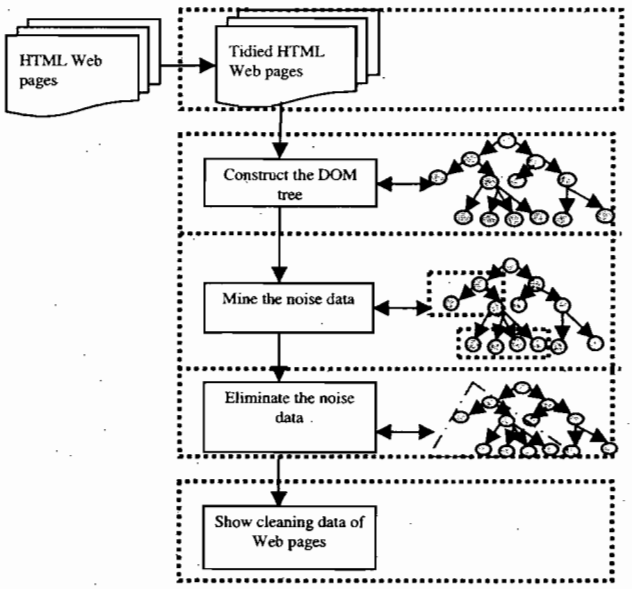

Figure 4. Process of eliminating the noise

4.1. Tidied HTML Web Pages

Web pages are not well formed documents, it is necessary to clean up them before processing. Similarly, some of the Web pages contain invalid tag structures where there are some opening tags without corresponding closing tags and vice versa. In addition, some HTML tags are not properly nested and also some are missing tags. However, some of the modern Web browser can ignore this invalid tag structure and browse the page even if it does not follow the HTML rules.

For the proposed system, these invalid tag structures cannot construct the DOM tree correctly. Therefore, we need to tidy these invalid tags to get the proper DOM tree. Thus we perform this Web page normalization tasks by using HTML tidy online check tool [12].

4.2. Construct the DOM Tree

In this step, we construct the DOM tree from the tidied HTML page. HTML pages are composed of tags and text, image, anchor text are enclosed within the tags. In a DOM tree where tags are internal nodes and the detail texts, images or hyperlinks are the leaf nodes for each HTML page. A sample DOM tree is shown in Figure 5. In this DOM tree, each solid-lined box is a tag node and the shaded box is the actual content of the node, eg., for the tag **img**, the actual contents is "src=image.gif". In Figure 5, the arrows which exist under the tag nodes are also the content of the node. In our case, we are interested in tag node, therefore we omit the contents in the tag node.

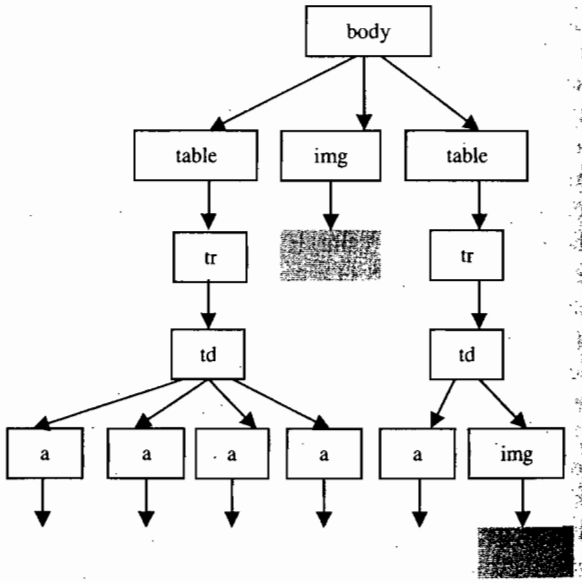

Figure 5. Sample DOM tree for HTML Web page

4.3. Mine the Noise Data

In this step, the system mines noise data in Web page such as directory list, search and filter panels, advertisement lists, privacy notice and copy right notice, etc. Firstly, we mine noise data instead of eliminate the noise data record directly. In section 3, we had already described noise information in Web page. Base on this, we define the rules which are noise information in Web page.

Rule 1: For a node N in the tree, if all of its decedent nodes have only one the same tag type which has no subchild, then we say node N is noise.

We detect the noise node in the DOM tree based on the rule 1 is as shown in Figure 6.

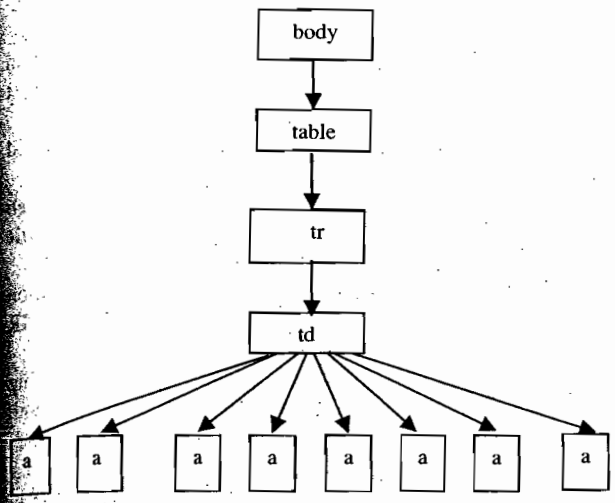

Figure 6. Example of noise node in DOM tree

Rule 2: For a node N in the tree, if its decedents nodes have different tag types and less than threshold t, then we say node N is noise.

In Figure 7, we detect the noise node in the DOM tree which is base on the rule 2.

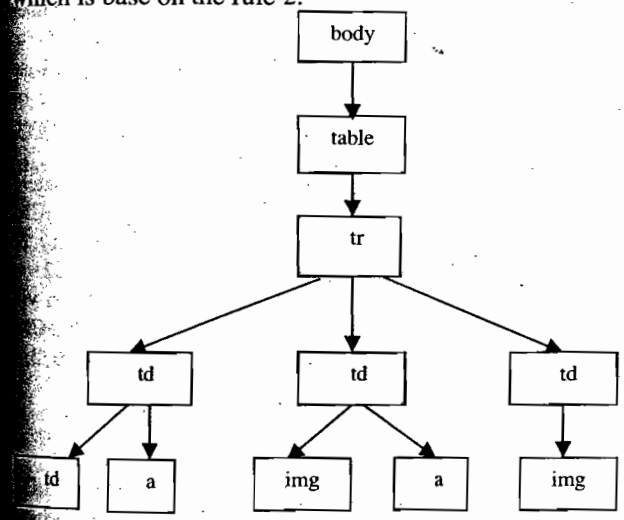

Figure 7. Example of noise node in DOM tree

4. Eliminate the Noise Data

the previous step, we have identified the noise data in eb page. In this step, we can eliminate the noise formation in HTML Web page. Noise data in Web page is, directory list which includes many links intend to go ther related topic. Similarly, in an advertisement lists, ich also include product advertisements that are not ated product information in Web page. We can eliminate noise information by using noise nodes elimination orithm in Figure 8.

gorithm: NoiseEliminate(threshold **dt**,DOMtree **t**)
While (dt.leaf!= null)
2. If node N has the same tag node type
then delete its deeedents as noise
Else if node N has the different tag node type and less than threshlod t
then delete its decedents as noise
3. End if

Figure 8. Noise nodes elimination algorithm

In this way we can eliminate the noise information in HTML Web page. Finally, we show the cleaning data without the noise information.

5. PERFORMANCE EVALUATION

For the system performance evaluation, we have downloaded HTML Web pages from the various commercial Web sites. Commercial Web sites as described in Table 1 and these sites are composed of HTML tags.

Table 1. Commercial sites with HTML Web pages

URL	Pages
www.drugstore.com	1033
www.bookpool.com	1245
www.booksamillion.com	1131
www.amazon.com	1589
www.barnes&noble.com	2161
Total	7159

To evaluate our proposed noise elimination method, we have tested on various commercial sites which include product information such as various kinds of book, lotion, food as shown in Table 1. Several commercial Web sites have different presentation style. We cannot use the same threshold for different sites. Therefore we use adaptive threshold for various sites to eliminate the noise information. Our proposed system can eliminate effectively all noise information such as search and filtering panel, directory list, advertisement and copy right and privacy notices. Other sites, we can eliminate search and filtering panel, copy right and privacy notices effectively but only to eliminate some of the directory list. Figure 9 shows the recall rate of the proposed system. Mostly, threshold value is less than 10; we can not eliminate the noise effectively. Likewise, Figure 10 shows the precision rate for eliminating the noise.

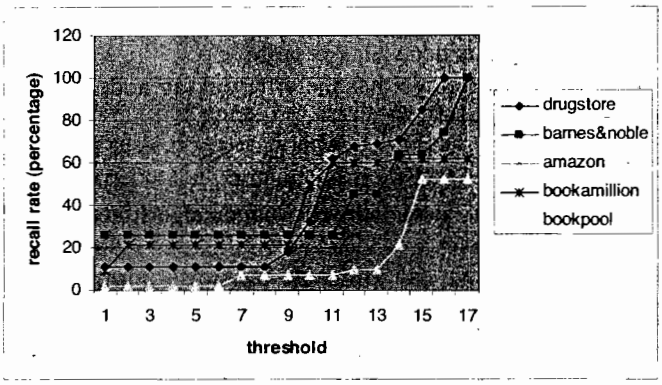

Figure 9. Recall rate of each Web site

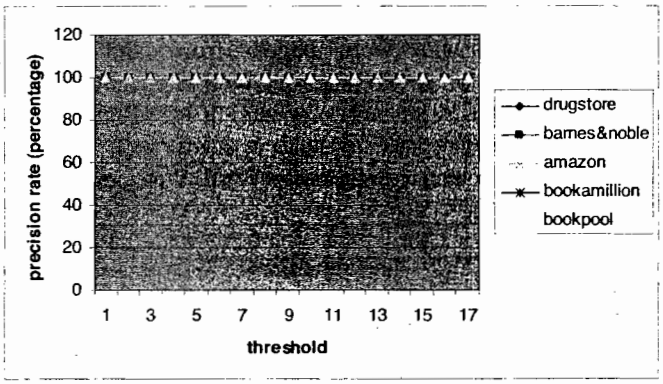

Figure 10. Precision rate of each Web site

6. CONCLUSION

Noise information in commercial Web pages is clutter around the body of the main content information. This noise information cannot give the correct result to the user who needs the main content. Our approach, working with the Document Object Model tree based on the proposed rules, thus enables to perform eliminate the noise information. By adopting noise information elimination as the preprocessor of information extraction application, the extraction precision will be increased and extraction complexity will also be reduced. The techniques that we have employed simple, are quite effective for eliminate the noise information form HTML Web page before main content extraction. In the future, we will add other method be applicable to general Web pages instead of restricting to commercial Web page.

7. REFERENCES

[1] B. Lui, R. Grossman and Y. Zhai, "Mining Data Records in Web Pages", SIGKDD 2003, Washington, DC, USA, August 2003.
[2] B. Liu, "Web Content Mining", the 14th International World Wide Web Conference (WWW), Chiba, Japan, May 2005.
[3] D. Cai, S. Yu, J.R. Wen and W.Y. Ma, "Block-based Web Search", in the proceedings of the ACM conference (SIGIR 2004), Sheffield, South Yorkshire, UK, 2004.
[4] I.Chibane and B. L. Doan, "A Web Page Topic Segmentation Algorithm Based on Visual Criteria and Content Layout", SIGIR 2007, Amsterdam, Netherlands, July 2007.
[5] L. Yi and B. Liu, "Web Page Cleaning for Web Mining through Feature Weighting", in the proceedings of Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03), Acapulco, Mexico, August 2003.
[6] L. Yi, B. Liu and X. Li, "Eliminating Noisy Information in Web Pages for Data Mining", in the proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2003), Washington, DC, USA, August 2003.
[7] N. Kushmerick, "Learning to remove Internet Advertisements", Autonomous agent 1999, Seattle WA, USA, 1999.
[8] P. S. Hiremath, S. S. Benchalli, S. P. Algur and R. V. Udapudi, "Mining Data Regions from Web Pages", International Conference on Management of Data (COMAD), Hyderabad, India, December 2005.
[9] S. Gupta, G. Kaiser, D. Neistadt and P. Grimm, "DOM-base Content Extraction of HTML Documents", in the proceeding of the 12th World Wide Web conference (WWW 2003), Budapet, Hungary, May 2003.
[10] S.-H. Lin and J.-M. Ho, "Discovering Informative Content Blocks from Web Documents", in the proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (SIGKDD'02), Edmonton, Alberta, Canada, 2002
[11] Z. Bar-Yossef and S. Rajagopalan, "Template Detection via Data Mining and its Applications", in the proceedings of the ACM WWW conference (WWW 2002), Honolulu, Hawaii, USA, 2002.
[12] HTML Tidy, http://www.infohound.net/
[13] DOM parser, http://www.w3c.org/DOM