Analyzing Web Page Headings Considering Various Presentation

Yushin Tatsumi
NEC Internet Systems Research Laboratories
8916-47 Takayama-cho, Ikoma-shi,
Nara, Japan
+81-743-72-3748
y-tatsumi@cw.jp.nec.com

Toshiyuki Asahi
NEC Internet Systems Research Laboratories
8916-47 Takayama-cho, Ikoma-shi,
Nara, Japan
+81-743-72-3709
t-asahi@bx.jp.nec.com

ABSTRACT

Exploiting document structure can solve the usability problem when browsing web pages designed for PCs with non-PC terminals. For example, by exploiting headings among document structure and showing them selectively within a display, users can easily grasp a page's overview. In this paper, as a basic part of document structure analysis, we propose a heading analysis method for web pages considering various presentation. Results of evaluation experiments confirmed that our proposed method could extract many headings that could not be extracted by using HTML element names.

Categories and Subject Descriptors

I.7.5 [**Document and Text Processing**]: Document Capture – *Document analysis*; H.5.4 [**Information Interfaces and Presentation**]: Hypertext/Hypermedia – *Navigation*.

General Terms: Algorithms, Documentation, Experimentation, Human Factors

Keywords: Web document analysis, heading analysis, content adaptation

1. INTRODUCTION

Recently, we usually browse web pages with such non-PC terminals as mobile phones, PDAs and information appliances. But it is difficult to browse web pages designed for PCs with non-PC terminals. To overcome this difficulty, a method combining tag exchange and page division [3] or a web browser by which web pages for PCs can be accessed with non-PC terminals [4] has been proposed. However, other problem exists regarding usability: users cannot easily grasp a page's overview and reach objective information with less operation when using a terminal whose display size or resolution is small. Generally, web pages have document structure: the layered structure of sections which may have a heading. It is thought that the above usability problem can be solved by exploiting this structure, for example, by selectively showing headings of each section within a display and enabling direct access to section content when its heading is selected. To realize such a user interface, we are researching document structure analysis of web pages. In this paper, we propose a heading analysis method of web pages as a basic part of document structure analysis, and report the evaluation results of our method.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

2. HEADING ANALYSIS

2.1 Problem

HTML specification [2] recommends that document structure and presentation should be written separately on a web page, even though actually, in many web pages they are written in a mixed manner. For example, some sections, which are document structure, are expressed only by layout, which is presentation. Other cases have headings, which are document structure, that are expressed only by background color, which is presentation. Therefore, to get reasonable document structure of a web page, document analysis considering presentation is needed. As such document analysis, the method that use gaps between HTML elements and so on [1] have been proposed. However, since the authors of web pages use more various presentation styles to express document structure, analysis considering varieties of presentation is thought to be needed. So we are researching document structure analysis considering various presentation. At this point, in document structure, headings are meaningful components because they not only give boundaries between sections but also represent each section. Therefore, this paper aims to accurately extract web page headings considering various presentation.

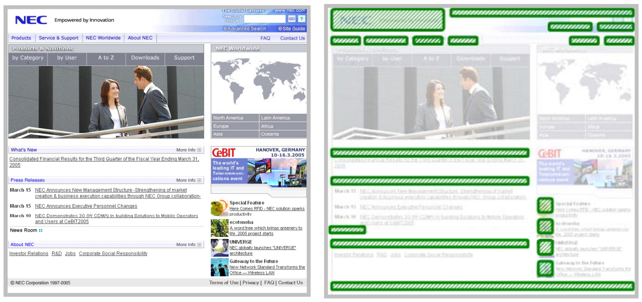
2.2 Algorithm

Our proposed method analyzes web page headings in two steps, HTML parsing and heading determination. The former parses HTML documents that describe a web page and generates a HTML DOM tree. The latter traverses the DOM tree from nodes corresponding to the body element of the leaf nodes, checks each node with a heading determination rule and extracts the node that suits the rule as a heading. By using rules compositively given with HTML element names, and styles and contents as a heading determination rule, heading analysis considering various presentation becomes possible.

The heading determination rule consists of three subrules, name, style, and content (Table 1). First, a processing node is checked with a name rule that extracts a heading that conforms to the original HTML grammar by using element names. Second, a node not extracted as a heading by name rule is checked with a style rule that extracts a heading expressed with a presentation, such as background color or image, text size, font weight and layout by using element style properties. In the third step, an extracted node as a heading by a style rule is screened by a content rule, which screens such an invalid node that does not have alt attribute or enough amount of text according to a kind of the node and extracts a valid heading by using element contents. The subnodes of a node, which is not extracted as a heading by the heading determination rule, are checked with the same rule succeedingly. Figure 1 shows the heading analysis results of the proposed method on NEC's homepage. In this figure, headings extracted by system are represented by hatched rectangles.

Table 1. Main heading determination rules

Name rule		Does it have h1-h6?
Style rule		Does it have background color or image?
		Does it have big text size?
		Does it have big font weight?
		Is it aligned with other headings?
		Does it have enough width, height, aspect-ratio?
Content rule	Multimedia	Does it have alt attribute?
		How many times is it used?
	Others	Does it have enough amount of text?
		Doesn't it have h1-h6 leaf nodes?



(a) Original page (b) Heading analysis results

Figure 1. Originalpage and heading analysis results of NEC's homepage

3. EVALUATION

Evaluation experiments were conducted to determine the accuracy of proposed method in which headings determined by users and headings extracted by the proposed method's system were compared. Two values were investigated: the value of headings determined by users that are correctly extracted by system, and the value of headings extracted by the system that failed to match user's determination.

Before the experiment, fifteen evaluation pages were randomly chosen between our company's portal page and pages linked to it. Headings determined by users were defined for every evaluation page. In concrete terms, three users determined headings of each page based on the exposition that the general document has the layered structure of sections and a section may have a heading as its representation, and the 182 headings commonly determined by all the three users are set as headings determined by users. In the experiments, system's heading determination rules remained the same. Headings determined by users that corresponded perfectly with those extracted by the system are regarded as correctly extracted by the system.

Table 2 shows the experiment results. First, among the 182 headings determined by users, 143 headings were correctly extracted by the system, a 78.6% correct extraction rate. Second, among the 199 headings extracted by the system, 56 did not match ones determined by users, a 28.1% failure to match users rate.

As mentioned above, 78.6% of the headings determined by users were extracted by the proposed method, which exceeded the heading extraction percentage using HTML element name, which was 13.2% (24/182). On the current Web, so many web pages express headings by presentation that the proposed method is thought to be able to extract many headings that could not previously be extracted. At this time, 28.1% of the headings extracted by the system do not match users' determination. In setting with the 240 headings all determined by any one of three users as headings determined by users, this percentage was 19.1%. This means that 19.1% of the headings extracted by the system completely did not match users' determination. When investigating the results of each page, typical problems seem to exist: for example, headings determined by users correspond to those extracted by the system one to many, so both of these investigated values are expected to improve.

Table 2. Results comparing headings determined by users with ones extracted by system

		Both		
			Written by HTML	Written by Presentation
Headings determined by users	Total	182 (100.0%)	24 (100.0%)	158 (100.0%)
	Correctly extracted by system	143 (78.6%)	24 (100.0%)	119 (75.3%)
	Incorrectly extracted by system	39 (21.4%)	0 (0.0%)	39 (24.7%)
Headings extracted by system	Total	199 (100.0%)	24 (100.0%)	175 (100.0%)
	Matched with user's determination	143 (71.9%)	24 (100.0%)	119 (68.0%)
	Unmatched with user's determination	56 (28.1%)	0 (0.0%)	56 (32.0%)

4. CONCLUSIONS

In this paper, we proposed a heading analysis method of web pages considering various presentation. This method is a basic part of a document structure analysis that aims for usability improvement of browsing web pages for PCs with non-PC terminals. We also reported evaluation experiment results of the accuracy of the proposed method, which confirmed that it could extract many headings that could not be extracted by previous methods. Future works will develop in three directions: improving and expanding the heading analysis method into a document analysis method: applying the heading analysis and a document analysis method to user interfaces that improve usability of web browsing with non-PC terminals: and stepping up evaluation in terms of expanding evaluation pages and discerning the accuracy level needed for usability improvement.

5. REFERENCES

[1] Chen, Y., Ma, W.Y., and Zhang, H.J., Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices, Proceedings of WWW2003, May 20-24, 2003, Budapest, Hungary, pp.225-233.

[2] HTML 4.01 Specification, http://www.w3.org/TR/REC-html40/.

[3] IBM WebSphere Transcoding Publisher, http://www-306.ibm.com/software/pervasive/transcoding_publisher/.

[4] Opera Software, http://www.opera.com/.