Round-Trip Web Page Rendering and Analysis for Testing, Indexing, and Security

(Extended Abstract)

Thomas M. Breuel and Daniel Keysers

No Institute Given

1 Introduction

The widespread adoption of HTML, DHTML, and web technologies has had many benefits, but a number of undesirable uses and problems have emerged as well. Some of these problems are unreliable cross-platform rendering of web pages, attempts to create web pages that deceive either web users or search engines, and lack of accessibility of some web pages by users with vision impairments or users with small screen devices. Standard approaches to addressing these problems rely on syntactic and semantic analysis of the web page source; for example, to determine whether a page is likely to render correctly, a style checker may check for the absence of certain tags or constructs known to cause problems on some browsers. Source based methods are fast, conceptually easy to implement, and can be built using standard parsing and text analysis tools, but they also have significant limitations. For example, the presence of style sheets, JavaScript, and other HTML and plug-in features makes it hard to make statements about the final, rendered form of a web page based on an analysis of its source text. Cross-platform browser problems can only be detected by such methods if the cause of the problem is understood and known, and if appropriate patterns have been formulated that can detect these problems in web page sources; such rules are likely to remain incomplete and their coverage spotty given the evolution of web standards. Similarly, detecting phishing or search engine spam is a co-evolutionary process between adversaries and tool creators–phishers and spammers will develop new attacks in response to each countermeasure.

As part of the image based personal computing project in our laboratory, we are developing round-trip rendering and analysis methods for addressing these problems. The foundation of our approach is the observation that the image presented to the end user is ultimately what determines the meaning of a piece of HTML (see also Breuel, 2004, Lopresti, 2005). In this talk, we report on on-going work in our laboratory on developing systems that address cross-platform browser and web page design testing, efforts for fighting phishing and search engine spam, and for improving accessibility.

2 Browser and Design Testing

There are multiple implementations of HTML rendering engines; some common ones are Microsoft's Internet Explorer, Mozilla's Gecko, Apple's Safari, Opera's browser, and KDE's KHTML. Each of these render web pages differently due to bugs and incomplete specifications of web standards. Common defects are missing text, text that is unintentionally rendered overlapping, text that unintentionally overlaps graphical elements, bad font substitutions, bad spacing, and unreadable choices of foreground and background colors.

Our approach to this problem is to render the HTML into an image-based representation and then subject the image-based representation to OCR (including layout analysis). Common rendering problems can be detected by comparing the HTML input against the OCR output. For example, incorrect rendering due to missing text, overlapping text, bad font substitutions, and text in invisible colors can be flagged by detecting text that is present in the original HTML but missing in the OCR output. Incorrect layouts can be detected by comparing the paragraph structure and reading order of the original HTML against the layout analysis output.

We automate this process by using user interface scripting support. Our initial prototype is implemented on Apple Macintosh OS X, where we use a combination of AppleScript, the Firefox ScreenGrab extension, and the Safari Snagit extension to automatically capture web pages with different browsers, versions, and browser settings, and to send the captured page to be processed by the OCR system; analogous technologies exist for Windows and Linux. This way, a large collection of web pages can be rendered, analyzed, and verified without the need for operator intervention in different browsers and browser versions. The approach can detect HTML layout problems without prior assumptions about layout engines and for browsers returning a wide variety of layouts; the approach correctly distinguishes incorrect and correct layouts even in the presence of JavaScript and style sheets.

The same approach can be used for checking HTML layouts against design rules. Design rules for HTML are intended to ensure readability, accessibility, and easy interpretation by readers, as well as to ensure correct representations of organizational identity. Design rules specify such features as minimum font sizes, acceptable fonts and color choices, and minimal spacings between logical groupings of page elements.

3 Phishing and Search Engine Spam

Phishing (www.antiphishing.org) is a problem in which an adversary attempts to obtain personal and private information by creating E-mails and web pages that belong to a trusted organization (e.g., the recipient's bank), but actually transmit the information to the adversary. Closely related to phishing is search engine spam, where a web site will present HTML content to a search engine that gives indications to the search engine that the web site contains relevant information about a popular topic but actually renders as an advertisement when viewed by the user. Web page source-based approaches have so far not been able to detect these manipulations reliably. Our approach to this problem is analogous to verifying rendering and accessibility of HTML: for each site, analyze the rendered image of the HTML, not the HTML source.

In the case of search engines, OCR engine output can be used in place of HTML for deriving index terms and analyzing web site content (in order to achieve better performance, OCR only needs to be used when there are indications in the HTML that the rendered HTML output might differ substantially from the input); initial testing on search engine spam sites shows this to be an effective technique for deriving correct indexing terms. Furthermore, inconsistencies between the OCR output and a simple textual analysis of the source HTML appears to be indicative of search engine spam attempts and could be used to exclude sites from further indexing.

To prevent phishing attacks based on the analysis of a rendered representation relies on a combination of checks: a round-trip analysis of the appearance of the user-visible URL, a check for the presence of logos and graphical trademarks on the page, and finally a recognition of all the text (whether present in images or not) on the page; together, these three checks can prevent most of the known phishing attacks.

We note that both Wenyin *et al.*, 2005, and Yu *et al.*, 2005, also take an image-based approach to phishing detection. However, they rely on visual similarity between phishing pages and a collection of known pages, which means that they cannot detect phishing pages that utilize the same style and visual components as a target page but differ in layout or detail. In contrast, we believe that our combination of approaches can detect many such pages with high probability.

4 Accessibility and Indexing

Another application of round-trip HTML analysis is for improving accessibility and search engine indexing of content. While web page authors are supposed to provide textual equivalents ("ALT" attributes) describing the content of all images, this practice is not consistently followed and is difficult to follow consistently in general. Our approach to improving accessibility and indexing consists of several stages. First, embedded images are individually analyzed and categorized into several categories, including images containing only text, business graphics with and without text, and arbitrary photographic images. They are also analyzed to derive a coarse description of the content, currently in terms of size, overall color, and color distribution, and, in the future, using more complex descriptors based on face detection and shape recognition. If there is text contained in an image, that text is recognized and made available. Second, the layout of the rendered web page is analyzed, to determine the logical functions (e.g., navigational elements, footnotes, annotations, body text) of different components of the web page source in the final document; this information cannot easily be derived from the web page source without rendering because function depends on the precise location and spatial relationships of page elements. The combination of the output from these analysis steps (image categorization, description, text recognition, and layout analysis) is then used to derive textual representations that permit access by the visually impaired, rendering in text browsers, and potentially improved indexing by search engines.

5 Discussion

Above, we have seen a number of applications for round-trip HTML rendering and analysis, in areas such as verifying correct cross-platform rendering, combating spam and fraud, and making web pages accessible both to vision impaired users and users on text-based or bandwidth limited devices. Compared to approaches that attempt to analyze and rewrite the HTML syntactically, the round-trip approach can cope with a much wider variety of HTML sources, including those using style sheets and JavaScript. The round-trip approach is also robust to future changes in HTML and browser standards and technologies. On the other hand, syntactic and text-based technologies are faster and comparatively simpler; furthermore, they also have some potential for improvement.

We are improving the system in several areas. While the current OCR system permits the principle to be demonstrated, they system we are using was optimized for recognition from scanned documents; recognition from captured web pages has significantly different characteristics and requirements (e.g., smaller font size, prior knowledge about exact font shapes, etc.). Furthermore, the current system is a prototype composed of several different applications; while this demonstrates the feasibility of automated capture and analysis, we are planning on integrating both capture and analysis into a single, deployable server-side application that can be used for testing, security, and accessibility applications. Finally, it will be necessary to develop standardized benchmarks to evaluate and compare the performance of different approaches to problems like accessibility, phishing prevention, and automated image descriptions.

References

Thomas Breuel: *The future of document analysis in the era of electronic documents.*, Keynote, Document Analysis Systems Workshop, Florence, Italy, 2004.

Daniel Lopresti: *Web Document Analysis–The Case of the Missing Dimensions*, Invited Talk, Int. Workshop on Web Document Analysis, Seoul, Korea, August, 2005.

Yingjie Fu, Liu Wenyin, Xiaotie Deng, EMD based Visual Similarity for Detection of Phishing Webpages, Proc. Int. Workshop on Web Document Analysis, Seoul, Korea, August, 2005.

Liu Wenyin, Guanglin Huang, Liu Xiaoyue, Zhang Min, Xiaotie Deng, Phishing Webpage Detection, Proc. 8th International Conference on Documents Analysis and Recognition (ICDAR 2005), pp.560-564