Evaluating Web Page Reliability across Varied Browsing Environments

Cyntrica Eaton and Atif Memon
Department of Computer Science
University of Maryland, College Park
College Park, MD 20742
Email: {ceaton,atif}@cs.umd.edu

Abstract

Web page reliability is significantly influenced by the browser, browser version, and operating system incorporated in end-user browsing environments. The tool we have developed evaluates reliability of web-bound content across various client platforms based on the relationship between two key accessibility factors: (1) the HTML tags that structure a given web page and (2) the support provided for each tag in target browsing environments. To optimize evaluation accuracy, we have designed a learning mechanism that inductively acquires knowledge of tag support criteria based on observed samples of accessible (properly rendered) and inaccessible (faulty) web page source code. Our contribution is expected to provide web developers with a basis for maximizing audience reach through improved awareness of the browsing environment profiles associated with diminished accessibility to their websites.

1 Introduction

Diversity in browsing environments utilized by the web community presents a unique challenge for web developers to effectively address threats to universal reliability; as reliance on web information and services increases, expectation of fault-free presentation and functionality simultaneously escalates. Although it would be ideal for web pages to render and function consistently across heterogeneous browsing environments, the browser, browser version, and operating system used to navigate and interact with web content are known to be significant factors in subsequent accessibility (Fig. 1).

The fundamental aim of this research is to provide web developers with a practical way of identifying web pages that will render improperly in various browsing, or client, environments in order to establish a basis for correcting corresponding issues and maintaining a favorable level of universal reliability. Our research efforts are a part of a general endeavor to support and improve the quality of user experience on the web.

2 Approaches

Intuitively, one approach to evaluating browsing environment-influenced reliability is to manually launch web pages within varied browser, browser version, and operating system configurations and qualitatively observe the results. While this dynamic testing strategy allows evaluators to gain direct exposure to existing reliability issues, limitations on time, manpower, and computing resources can severely restrict the depth of the website tested and the breadth of browsing environments explored. Even automated tools, such as Browser Photo [2], that essentially deploy web pages in varied environments and provide thumbnails of rendered results, have restrictions on the variations in browsing environment model tested. An alternative, highly effective quality assurance evaluation is static-based and rooted in the fact that the reliability and subsequent accessibility of web pages is heavily dependent upon the HTML tags that structure web pages and the ability of target environments to recognize and correctly process each. In short, unrecognized HTML tags can be regarded as faulty code fragments, or bug patterns, in unsupportive environments and essentially used as reliability predictors. To further illustrate this idea, consider the following definition:

Definition 1: Environment Specific Bug Patterns
Let E denote a browsing environment defined by the triplet $< B, V, O >$ where B is the browser, V is the browser version, and O is the operating system. Consider T, the space of all possible HTML document source tags:

$$\{\{\forall e_j \in E \; \exists I = \{i_1, i_2, \ldots, i_{|I|}\} s.t.$$

$$(I \subseteq T) \wedge (unsupported(I, e_j)\}\} \quad (1)$$

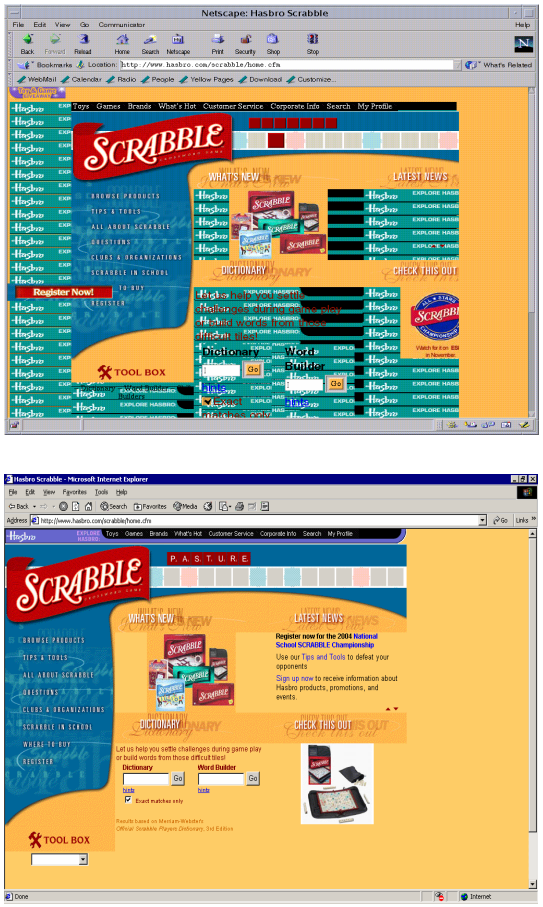

Figure 1. Results of deploying [1] in Netscape 4.7 (top) and Internet Explorer 6.0 (bottom)

That is, each browsing environment supports only a subset of the overall tag space T. All other tags are unrecognized or incompliant in the associated environment. Consequently, tags in I could be considered bug patterns for web pages rendered in environment e_j and cross browser accessibility of a website can be evaluated in a code review by detecting the presence of tags $i_n \in I$.

The tool we have developed employs this evaluation strategy and, in effect, compares the HTML tags appearing in the source document of a web page to an inventory of bug patterns associated with various browsing environments. While other tools, such as Doctor HTML [3] and Bobby [4] incorporate similar assessment techniques, we have designed a tool augmentation that takes into account the need for a complete and accurate definition of I for each client environment. Consider the following:

Definition 2: Bug Pattern Knowledge Completeness
The accuracy of a code review based on the set of tags in I is largely dependent upon the accuracy and completeness of the description of I. If for instance:

$$\{\exists t_i \in T s.t. (t_i \notin I) \wedge unsupported(t_i, e_j)\} \qquad (2)$$

performance of static analysis that does not include t_j as a bug pattern will be compromised.

Given the need to ensure comprehensive bug pattern representation for effectual use of the tool, we have designed a learning algorithm to automate knowledge acquisition of I based on an inductive methodology. In this case, web pages are the raw material for induction; namely, the tags they contain and their classification as either positive (accessible) or negative (inaccessible) examples of web page behavior, presentation, or both in a given environment. The underlying theory of the current learning technique is that HTML tags positively correlated with negative examples are probably bug patterns in the corresponding browsing environment. In an effort to gain input data for the learner, we have developed a website for capturing information about accessible and inaccessible pages in various browsing environments and we encourage you to submit the URLs of malfunctioning or improperly rendered pages in addition to the corresponding browser, browser version, and operating system utilized [5].

3 Conclusion

From our perspective, providing tools capable of detecting environment-influenced accessibility issues is extremely important for supporting universal reliability considering the gamut of browsing platforms in use. Our framework, which features an HTML bug pattern evaluation and a learning mechanism that updates tag support rules based on inductive reasoning, is still in development and we are currently exploring more effective ways of learning in this particular domain.

References

[1] Hasbro Scrabble Home Page. Available at: http://www.hasbro.com/scrabble/home.cfm

[2] Browser Photo by NetMechanic. Available at: http://www.netmechanic.com/browser-index.htm

[3] Doctor HTML. Available at: http://www2.imagiware.com/RxHTML/

[4] Bobby. Available at: bobby.watchfire.com/

[5] Faulty Web Page Submission Portal. Available at: http://www.cs.umd.edu/ ceaton/Faulty.html