The Transparency of Automatic Accessibility Evaluation Tools

Parvaneh Parvin*
Vanessa Palumbo*
parvaneh.parvin@isti.cnr.it
vanessa.palumbo@isti.cnr.it
CNR-ISTI, HIIS Laboratory
Pisa, Italy

Marco Manca
Marco.Manca@isti.cnr.it
CNR-ISTI, HIIS Laboratory
Pisa, Italy

Fabio Paternò
Fabio.Paterno@isti.cnr.it
CNR-ISTI, HIIS Laboratory
Pisa, Italy

ABSTRACT

Several web accessibility evaluation tools have been put forward to reduce the burden of identifying accessibility barriers for disabled people. One common issue in using accessibility evaluation tools in practice is that the results provided by different tools are often variable. Such variability may confuse the users who may not understand the reasons behind it, and thus limits their possible adoption. Hence, there is a need to put light on the tools actual functioning, indicate what criteria they should have to be transparent and to help users to better interpret their results. In this communication paper, we discuss such issues, analyse how they have been addressed by a representative set of tools, and provide indications useful for obtaining user-centred accessibility evaluations.

CCS CONCEPTS

• **Human-centered computing** → **Accessibility systems and tools**.

KEYWORDS

accessibility, automatic validation tools, transparency

ACM Reference Format:
Parvaneh Parvin, Vanessa Palumbo, Marco Manca, and Fabio Paternò. 2018. The Transparency of Automatic Accessibility Evaluation Tools. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Following the adoption of accessibility laws, various government organizations started paying more attention to accessibility guidelines. However, Web accessibility requires constant monitoring of many details across many pages. Thus, to simplify the monitoring, analysis, detection, and correction of website accessibility problems, several automatic and semi-automatic tools have been proposed. Accessibility validation is a process that cannot be fully automated [9]. Nevertheless, automated tools still play a crucial role in ensuring the accessibility of websites. They help human operators collect and analyse data about the actual application of accessibility guidelines, detect non-compliance, and provide relevant information about addressing the possible problems. The goal of such tools is to support stakeholders (i.e., developers, accessibility experts, web commissioners) in making better decisions faster while reducing their workload. Brajnik [3] discussed the effectiveness of accessibility evaluation tools in terms of completeness, correctness, and specificity. [6] Ivory and Hearst provides a taxonomy to analyse the quality of automatic evaluation tools based on the usefulness of what they suggest and the type of support that they can provide. In another work, Vigo et al. [9] analysed the effectiveness of six frequently used accessibility evaluation tools in terms of coverage, completeness, and correctness with respect to WCAG 2.0 guidelines. Molinero et al. [8] conducted a study showing that the results provided by Web accessibility tools are often variable, and thus users may conclude that they are not reliable. Generally, it is easy to see that when applying different validation tools to the same web content, they provide different results, and users have difficulties understanding the reasons for such variability, and to what extent the results are meaningful. Thus, there is a need for more transparency to help users better interpret their results.

This paper aims at providing criteria to analyse the transparency of such accessibility validation tools. The motivations are twofold: on the one hand, tools should provide transparent detailed information on how they operate and how many of the techniques in the guidelines they cover in evaluating the web pages. On the other hand, the representation and visualisation of the evaluation results determine how users can make a decision based on the accessibility assessment.

In general, little attention has been paid to how the automatic accessibility evaluation tools should be designed to provide clear information about their coverage and working. In the paper, we identify some relevant design aspects and undertake an analysis of four different web accessibility evaluation tools (Mauve ++, WAVE, AChecker, Qualweb). The focus of the analysis and subsequent discussion is to show how these tools make their capabilities clear to users, and present the results of the accessibility assessment, concentrating on how transparent and understandable such results are. This can affect the users' trust in the information provided and help them act accordingly to remedy any accessibility problems.

*Both authors contributed equally to this research.

Woodstock '18, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

2 TRANSPARENCY IN ACCESSIBILITY VALIDATION TOOLS

As stated, transparent tools should enable users to make fully informed decisions based on a clear understanding of how the automatic validation tools work. In particular, in order to be transparent an automated validation tool should make explicit the following information on its operations.

- what standards, success criteria, and techniques are considered.
  This point is critical because it helps clarify the reasons for the different results in different tools. Besides, the more techniques a tool covers, the more precise the results are because different techniques reveal different accessibility problems [1].
- how it categorizes accessibility issues
  The classification of the accessibility validation results indicated by the W3C organization [2] recommends using one of the following categories to indicate the tests' results: passed, failed, cannot tell, inapplicable and untested. The more a tool utilises this standard classification for the accessibility issues, the more understandable its results will be for the user. If a tool uses different terms to categorize the accessibility issues, their explanation must be clear and available for users.
- how the reported information is organised
  (1) is the tool capable of giving overall accessibility measures for entire web pages or sites ? In addition to the ability to report lists of issues, the use of relevant metrics can help indicate the overall accessibility level of the considered websites. These metrics can be defined on the basis of success criteria and their corresponding sufficient, advisory and failure techniques.
  (2) are there different report formats depending on the target users ? Different report formats can fulfil the needs of users with various expertise. For example, a code overview is more suitable for developers and a report with charts and statistics summarising the detected issues can be sufficient for non-technical users.
- whether the tool is able to evaluate dynamic pages or not.
  Indicating whether a tool can evaluate dynamic web pages is useful information that can help users interpret the results properly. Several accessibility validations tools rely on static HTML to analyse the DOM composing the web page; however, often current web sites have evolved into more dynamic applications. By evaluating the static HTML composing a modern web application, a tool may erroneously return no errors. However, in this case, the absence of errors may not mean that the target application is fully accessible but rather that the tool is unable to assess it without giving any indication of this limitation to users.

Thus, to be transparent, the accessibility validation tools should provide their users with clear information about the full functionality of the tool. In the following sections, we explore these criteria for each of four tool (Mauve ++, WAVE, AChecker, Qualweb) and will compare these tools considering the above-mentioned points of transparency.

3 AN ANALYSIS OF FOUR TOOLS

In the choice of the analysed tools we considered: their ability to test web pages against at least the WCAG 2.0 guidelines, and they should be non-commercial and freely accessible on the Web. In this way, they are to some extent representative of the several automated accessibility evaluation tools that have been put forward in this area. In particular, we looked at AChecker, which is the longest-lived of the group of tools, and one of the most used over the years; QualWeb is a recent tool and therefore representative of the most recent developments in this field; WAVE was chosen as representative of the tools that present the accessibility validation results directly in the user interface of the analysed web page; MAUVE++ was chosen because of its expandability due to the separation it effects between the guidelines specification and the validation engine. Table 1 provides some insight into the features of each tool.

3.1 Mauve++

MAUVE++ [4] is able to analyse websites according to their compliance to WCAG 2.0 and 2.1 guidelines (both for Level A, AA, AAA). The tool currently supports "107 WCAG 2.1 HTML" and "8 WCAG 2.1 CSS" Techniques, and addresses 46 success criteria. This information is at the user's disposal in the *Help* menu of the tool, and they can view the full list of all the techniques and criteria supported.

Mauve++ can evaluate the accessibility of entire websites. Users who want to use this service need to register to the tool. The interface offers the possibility to evaluate the web pages in three different ways, i) entering the URL of a web page; ii) uploading an HTML file; iii) pasting the complete HTML source code from a Web page. Besides, users can choose different viewport settings (e.g., mobile device, desktop, tablet) based on their needs for selecting the version of the web content to assess.

Users also can choose the WCAG guidelines and the conformance level they desire to apply in their assessment. Mauve++ also provides a Chrome plugin (which can also be installed on Edge Browser) to evaluate i) non publicly available web pages, ii) dynamic pages that can change depending on the interaction with the users, iii) password-protected pages.

3.2 WAVE

Wave - Web Accessibility Assessment Tool is a free tool provided by the Web Accessibility In Mind (WebAIM) [7] organization. Its functionalities are also available as browser plugins (for Chrome and Firefox) to evaluate dynamic web content.

The "Help" section of the Wave website provides information concerning the evaluation result features. There is documentation for each icon and information boxes used to indicate the accessibility issue on the page during the evaluation process. WAVE claims to detect compliance issues found in the WCAG 2.0 guidelines, many of those in Section 508 (U.S accessibility law) and the WCAG 2.1 guidelines.

In the documentation, they also provide guidance on the most relevant and impactful WCAG 2.0 (level A and AA) success criteria that the tool supports during the evaluation of the web pages (i.e. Errors (13 SC), Alerts (6 SC), Features (6 SC), Structural elements

(4 SC) and Aria (7 SC). One of the limitations of the public version of WAVE is that it does not allow the user to choose between the W3C Web Content Accessibility Guidelines (WCAG 1.0, 2.0, 2.1) and Priority Level (A/AA/AAA) they need for evaluating the web page.

3.3 AChecker

AChecker [5], is an Open Source accessibility tool to evaluate HTML content for accessibility problems. The interface offers the possibility to evaluate the web pages in three different ways: i) entering the location of a web page; ii) uploading an HTML file; or iii) by pasting the HTML source code from a Web page. The tool allows users to enable or disable options including HTML Validator, CSS Validator and Show Source. Moreover, the user can select amongst nine international accessibility standards when conducting an assessment, namely the HTML Validator, BITV, Section 508, Italian Stanca Act, WCAG 1.0 and WCAG 2.0 and also the "Report Format" (by the guideline or by line number). Unfortunately AChecker does not support the most recent version of WCAG guidelines (v.2.1 released in 2018). Achecker does not provide any support for evaluating dynamic web pages (no server-side rendering capabilities nor browser extensions).

3.4 QualWeb

QualWeb is an open-source automated web accessibility evaluation service which incorporates contributions from different research projects and efforts. A user can automatically check a web page directly by inserting its URL. The tool can evaluate a set of WCAG 2.1 Techniques (43 WCAG 2.1 HTML and 5 WCAG 2.1 CSS Techniques) and ACT Rules (69 in total). While this information is accessible for technical users at the *https : //github.com/qualweb*, the tool gives limited information to non-technical users regarding the guidelines support in the tools' *"About"* section. In the *About* section, they declare that a browser extension will be released soon, which means currently they do not provide any support for dynamic pages evaluation. However, by analysing the report provided for a Single web Page, it seems the tool can evaluate it correctly through a server-side rendering (SSR) capability.

4 RESULTS DISPLAY

The presentation of results is important especially when medium-large scale evaluations have to be performed or when evaluation reports are needed in real time. In particular, in this section, we examine how the accessibility issues found are categorized and how understandable the reported results are.

4.1 Mauve++

Mauve++ provides a double view of the accessibility results. One is a code-oriented view where errors and warnings are presented in the line number of the evaluated web page source code; the other is a graphical view for users without programming skills, which shows errors and warnings through charts and tables. The tool also provides an overview of the validation results showing the number of errors, warnings and successes. In Mauve++, *Error* represents an accessibility violation that can be detected automatically. *Warning*, represents a possible problem that cannot be verified automatically and needs a manual review. *Success* is associated with the elements that passed the test.

In the code-oriented view, each violated or potentially violated technique is represented with a different colour (red for errors, yellow for warnings). Besides, for each detected issue there is a link to the corresponding technique in the W3C website. In addition, the tool provides two accessibility measures. *Accessibility Percentage*, which shows the accessibility of the website in terms of the number of checkpoints successfully evaluated over the total number of evaluated checkpoints for which the tool has been able to make the validation, and *Evaluation Completeness* which indicates the percentage of evaluated checkpoints for which the tool has been able to make a validation. As a result, the tool communicates the results in the following ways:

- a PDF with the validation report emailed to the registered user when an entire web site or multiple pages are assessed:
- a web report in the case of a single web page validation, which has separate views depending on whether the user is a web developer or an end-user without programming knowledge.

4.2 WAVE

In the public free version, the Wave evaluation report consists of different areas which allow the user to explore the results at different abstraction levels. At a higher level, it shows a summary view of all the errors. It contains the total number of issues for each of six categories (errors, alerts, features, structural elements, HTML5 and ARIA, and contrast errors). At a lower abstraction level, it shows what type of barriers have been addressed and, for each issue, users can see the meaning of the flagged problem, the solution and also the standards and the reference guidelines in the "references" section.

However, rather than providing a structured technical report, WAVE shows the original Web page with embedded icons and indicators that reveal the accessibility information within the page. It also shows a code panel with the problematic part of the code. Once the evaluation of the web page has been finished, WAVE does not offer the possibility to create and download the evaluation report. This could be a problem for those who need to save the various states of the accessibility evaluation process or for those who need to communicate the results.

4.3 AChecker

AChecker produces a report of all accessibility problems based on the user-selected guidelines and identifies three types of problem: Known problems, that have been identified with certainty as accessibility barriers; Likely problems, that have been identified as probable barriers but require a human to make a decision; Potential problems, which are problems that AChecker cannot identify and require a human decision. The specifics of each check can then be viewed for full details of check requirements, the procedures for conducting the test, as well as information on resolving the barriers, along with example files that pass or fail the check. Thus, reviewers have full access to all the details of their accessibility evaluations and can be confident in knowing precisely what was evaluated.

Table 1: compare tools features

	Supported Success Criteria	ACT rules	Priority level	Dynamic web page support	Report	Accessibility metric	Errors category
Qualweb	43 HTML,5 CSS	69	A/AA/AAA (Filter after evaluation)	Yes	Web Report	No Metrics	Passed/Failed/Warning/Not Applicable
Mauve++	107 HTML,8 CSS	7	A/AA/AAA (Before Evaluation)	Yes	Web Report/PDF	Accessibility Percentage/Accessibility Completeness	Errors/Warning/Success
Achecker	No Info	No Info	A/AA/AAA (Before Evaluation)	No	Web Report/HTML/Earl/PDF/CSV	No Metrics	Known Problems/Likely Problems/Potential Problems
Wave	23 HTML/CSS	No Support	No Options	Yes	Web Report	No Metrics	Errors/Alerts

If the user clicks on the detected barrier title, a new window appears to demonstrate a range of information on the issue type and instructions on how to repair it. In AChecker, it is possible to download the evaluation results. The user can choose whether to download the complete report or only one accessibility problem in the preferred format among those available (i.e., PDF, EARL, CSV, HTML).

4.4 QualWeb

The web report of the Qualweb tool consists of several sections. The *summary* shows the total number of errors, *Filter*, allows users to filter the results that match their particular needs. The *Evaluation Report*, gives a complete report of all the errors combined based on the previous section (i.e., Filter).

The tool's report includes the description and the results of the tested rules (i.e., passed, failed, warning and not applicable), a link to the full description of the rule at the $https://www.w3.org/$ website, the related success criteria along with the priority level, and finally, the HTML code line related to the recognized issue. A *failure* occurs when the tool can detect automatically and unambiguously if a given HTML element has an accessibility problem. A *pass*, generates from elements that, unambiguously, are classified as having no accessibility problems. *Warning* raises when the tool can partially detect accessibility problems, but which might require an additional inspection (often by experts). A *Not Applicable* issue occurs when there are not relevant elements on the web page to be tested.

Another interesting functionality in the report section of this tool is the *Visual Representation* button, which triggers the preview of the specific issue on the evaluated website.

5 COMPARE TOOLS TRANSPARENCY

In this section, we analyse to what extent the tools fulfil the properties we defined in the transparency section (see section 2). Regarding the first point of the Transparency section, Mauve++, Wave and QualWeb indicate which SC and Techniques they can verify in the about/help page. Only AChecker does not provide this information in detail. In general, if accessibility evaluation tools bring to light the success criteria and standards they apply in assessing the accessibility of the web page, users will have a better understanding of why the results from various tools are different.

Considering the categorisation of the issues detected by the evaluation tools, QualWeb provides the most complete report by categorising the results as passed, failed, warning and not applicable issues[2]. Mauve++ report only errors, warning and passed checkpoints. Wave reports only Errors and Alert issues, while AChecker provides a report adopting a different error categorisation (i.e., Known, Likely and Potential Problems).

The third point of the transparency list pinpoints how the tools present the accessibility issues detected during the evaluation. Only Mauve++ provides two metrics that help users to better understand the website accessibility percentage and how much the tool has been able to evaluate the guideline checkpoints. Regarding the evaluation reports, WAVE directly renders the view of the evaluated page by overlaying an icon for each issue on the user interface; however, the display may be confusing due to absolute positioning, and does not show any other information about the related elements such as markup or ID/class attributes. This is somewhat problematic if users have issues on the user interface parts that are not yet visible, such as hidden forms. While it takes some expertise to interpret some icons, users can toggle the 'no styles' mode which will give the plain 'HTML view' of the user interface but still show WAVE icons next to the related elements. This also has the useful effect of displaying user interface parts that are visually hidden with the CSS.

Mauve++ provides a double visualization mode: the first one is a code-oriented style for developers that shows the errors or the warnings above the code line which generated it; while the latter is designed for non-developer users (i.e. accessibility expert, web commissioners) as it presents the accessibility issues through tables and charts. QualWeb mixes the developer display supported by Mauve++ and a preview of the user interface part that generated the accessibility issue. Finally, AChecker provides only a code-oriented report. Regarding the last transparency point, only AChecker does not provide any support for dynamic pages evaluation. While Mauve++ and WAVE support browser extensions, QualWeb provides a server-side rendering (SSR) capability.

6 CONCLUSIONS AND FUTURE WORK

We have presented and discussed some design criteria for supporting the transparency in accessibility validation tools. They have been used to analyse a set of four publicly available tools, which have various features and address them differently. In general, they tend to consider some relevant aspects, but there is still need for substantial progresses in order to achieve full transparency.

In future work, we plan to deepen and extend the analysis of the automated evaluation tools in terms of transparency criteria in order to provide more detailed information for users, which can also be useful for choosing the most suitable tool for their purposes. Besides, we plan to carry out usability studies to validate the criteria indicated for accessibility evaluation transparency.

REFERENCES

[1] Siddikjon Gaibullojonovich Abduganiev. 2017. Towards automated web accessibility evaluation: a comparative study. *Int. J. Inf. Technol. Comput. Sci.(IJITCS)* 9, 9 (2017), 18–44.

[2] Shadi Abou-Zahra. 2017. Evaluation and Report Language (EARL). Retrieved February 2, 2017 from https://www.w3.org/TR/EARL10-Schema/#OutcomeValue

[3] Giorgio Brajnik. 2004. Comparing accessibility evaluation tools: a method for tool effectiveness. *Universal access in the information society* 3, 3-4 (2004), 252–263.
[4] Giovanna Broccia, Marco Manca, Fabio Paternò, and Francesca Pulina. 2020. Flexible automatic support for web accessibility validation. *Proceedings of the ACM on Human-Computer Interaction* 4, EICS (2020), 1–24.
[5] Greg Gay and Cindy Qi Li. 2010. AChecker: open, interactive, customizable, web accessibility checking. In *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*. 1–2.
[6] Melody Y Ivory, Jennifer Mankoff, and Audrey Le. 2003. Using automated tools to improve web site usage by users with diverse abilities. *Human-Computer Interaction Institute* (2003), 117.
[7] Leonard R Kasday. 2000. A tool to evaluate universal Web accessibility. In *Proceedings on the 2000 conference on Universal Usability*. 161–162.
[8] Ashli M Molinero, Frederick G Kohun, and R Morris. 2006. Reliability in Automated Evaluation Tools for Web Accessibility Standards Compliance. *issues in Information Systems* 7, 2 (2006), 218–222.
[9] Markel Vigo, Justin Brown, and Vivienne Conway. 2013. Benchmarking web accessibility evaluation tools: measuring the harm of sole reliance on automated tests. In *Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility*. 1–10.