Web Accessibility Evaluation with the Crowd: Using Glance to Rapidly Code User Testing Video

Mitchell Gordon
University of Rochester
Rochester, NY 14627
m.gordon@rochester.edu

ABSTRACT

Evaluating the results of user accessibility testing on the web can take a significant amount of time, training, and effort. Some of this work can be offloaded to others through coding video data from user tests to systematically extract meaning from subtle human actions and emotions. However, traditional video coding methods can take a considerable amount of time. We have created Glance, a tool that uses the crowd to allow researchers to rapidly query, sample, and analyze large video datasets for behavioral events that are hard to detect automatically. In this abstract, we discuss how Glance can be used to quickly code video of users with special needs interacting with a website by coding for whether or not websites conform with accessibility guidelines, in order to evaluate how accessible a website is and where potential problems lie.

Categories and Subject Descriptors

K.4.2 [**Social Issues**]: Assistive technologies for persons with disabilities; H.5.m [**Information Interfaces and Presentation**]: Misc.

General Terms

User studies, Experimentation, Human Factors.

Keywords

Accessibility, video, data analysis, crowdsourcing

1. INTRODUCTION

User testing is considered an important part of web accessibility evaluation. In the W3C's Website Accessibility Conformance Evaluation Methodology draft, they recommend involving people with disabilities as part of the evaluation methodology. The W3C also published Web Content Accessibility Guidelines (WCAG 2), which help identify the types of important questions to ask when evaluating website accessibility.

However, involving people with disabilities directly in the evaluation process may require both time and training, which developers frequently lack [6]. While testing can be performed remotely, the nature of self-reports mean that the data collected may be limited when compared to in-person studies [7]. In addition to user testing, automated testing is also commonly used to evaluate the accessibility of a website [7]. This type of testing is significantly easier to run, quicker, less costly to evaluate than user testing, and often consists of simply entering a website URL and receiving a list of possible accessibility issues. Though not evaluated in the context of accessibility, the crowd-powered system PatFinder is able to use video of users performing tasks to identify higher-level interaction patterns, which describe how to complete tasks such as 'buying a book about HCI.' [4] However, automated techniques have not previously been able to evaluate subtle human actions and emotions that can result from users during user testing [7].

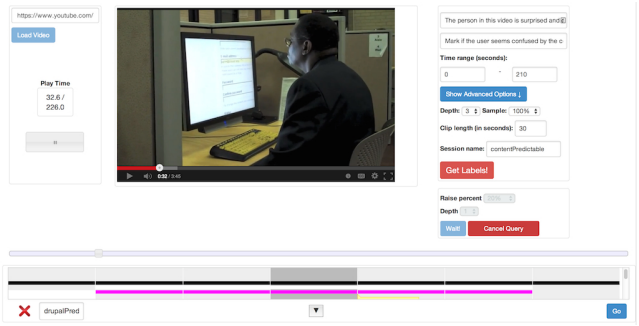

Figure 1: The Glance user interface. Glance can code events in user testing video quickly and accurately. When a usability question is asked, small clips from the video are sent to crowd workers who label events in parallel. The judgements are then quickly merged together and displayed. In this example, we use Glance to determine if and when a disabled user encountered unpredictable content.

This submission introduces the use of Glance, a crowd-powered video coding tool [5], as a way to code video recordings of user testing and significantly decrease the time and cost associated with evaluating the results of a user test.

2. VIDEO CODING

Behavioral video coding allows researchers in the social sciences to study human interactions [2]. In HCI, researchers often use video coding to discover how users interact with technology [3], and to help better explain those interactions [2].

Video coding is important because it provides a systematic measure of behavior. However, it is commonly considered a very time-consuming process, with some researchers claiming that it can take 5-10x longer than the play time of the

video itself [1]. Additionally, video coding requires a significant amount of overhead. In order to perform video coding on data, researchers must develop a reliable coding scheme, acquire and train coders, and check for inter-rater reliability. All these factors combined means that performing video coding to evaluate user tests has previously had a very high barrier to entry.

3. GLANCE

Previously, we have presented Glance, a system that allows researchers to analyze and code events in large video datasets by segmenting videos and then parallelizing the video coding process across a crowd of online workers (Figure 1) [5]. This approach significantly reduces the amount of time required to gather information from video data, and allows video to be coded in a fraction of the actual play time, depending on the size of the available crowd. To ensure accuracy, Glance can distribute the same video segments to multiple unique workers, and then calculate the variance to provide quick feedback. Glance provides a front-end interface for analysts to enter natural-language queries and to visualize coding results as they arrive.

Coding video with Glance to evaluate the results from user studies can significantly reduce the amount of time and effort required to obtain actionable information from user tests using the power of the crowd. Additionally, the time, cost, and effort savings that Glance affords makes the use of video coding to evaluate user tests feasible for a larger number of web developers.

We can use the WCAG 2 guidelines, created by the W3C, to determine what to code video of user studies for. For example, some questions we might ask are "did the user encounter this type of problem: no caption or other alternative provided for multimedia" or "did the user encounter unreadable or difficult to understand text?". By using sighted crowd workers as video coders, we are able to identify problems which may not be identified through remote self-reported accessibility evaluations [7] without requiring developers to take time from the development process to conduct usability testing themselves.

4. EVALUATION AND RESULTS

To evaluate Glance's ability to code video of accessibility user testing and return reliable results, we ran a feasibility experiment using Glance's "gist" mode, which asks workers to simply mark if any instance of the event occurs within a small clip, rather than asking them to mark the exact range in which it occurs. We believe that gist mode is appropriate for this scenario because determining, for example, exactly what time a user starts and stops being confused is highly subjective and may require additional context that only the web developer is able to provide.

Our evaluation used a video of a user test that consisted of multiple visually-impaired users both using a website on a desktop computer and providing verbal feedback. We coded for the WCAG 2 guideline "make content appear and operate in predictable ways" (though slightly re-worded to help the crowd better understand what to code for). This small evaluation consisted of 20 Mechanical Turk workers coding a three and a half minute video. The crowd correctly coded clips with a precision of 80% and recall of 100%. These scores are comparable to the scores obtained from Glance's initial evaluation, showing that video coding of accessibility user tests is no less accurate than other anticipated uses of Glance.

5. FUTURE WORK

We would like to expand our evaluation to include a larger variety of user study videos and code them for all WCAG 2 guidelines, as well as other sets of accessibility guidelines.

We also believe that, in addition to website user test evaluations, Glance has many possible applications within accessibility. These are limited only by what types of videos can be created and by what types of events can be accurately coded for. Some of these possibilities include:

- Visually-impaired users taking a panoramic video with their phone and asking a question about it.
- Disabled users can upload a video of their interaction with a website or app that is not accessible. If they were not able to completed a desired action in the inaccessible app, they could ask a question to help figure out where in the process of using the app they went wrong.

6. REFERENCES

[1] *Handbook of Research Methods in Social and Personality Psychology*. Cambridge University Press, 2000.

[2] R. Bakeman and J. M. G. PhD. *Observing Interaction: An Introduction to Sequential Analysis*. Cambridge University Press, 1997.

[3] B. Jordan and A. Henderson. Interaction analysis: Foundations and practice. *The Journal of the Learning Sciences*, 4(1):pp. 39–103, 1995.

[4] W. Lasecki, T. Lau, G. He, and J. Bigham. Crowd-based recognition of web interaction patterns. In *Adjunct Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST Adjunct Proceedings '12, pages 99–100, New York, NY, USA, 2012. ACM.

[5] W. S. Lasecki, M. Gordon, D. Koutra, M. Jung, S. P. Dow, and J. P. Bigham. Glance: Rapidly coding behavioral video with the crowd. UIST 2014, New York, NY, USA, 2014. ACM.

[6] J. Lazar, A. Dudley-Sponaugle, and K.-D. Greenidge. Improving web accessibility: a study of webmaster perceptions. *Computers in Human Behavior*, 20(2):269–288, 2004.

[7] J. Mankoff, H. Fait, and T. Tran. Is your web page accessible?: A comparative study of methods for assessing web page accessibility for the blind. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pages 41–50,