Crowd-Based Recognition of Web Interaction Patterns

Walter S. Lasecki, Grant He, Jeffrey P. Bigham
University of Rochester
Computer Science, ROC HCI
Rochester, NY, 14627 USA
{wlasecki,ghe3,jbigham}@cs.rochester.edu

Tessa Lau
IBM Research – Almaden
San Jose, CA 95120 USA
tessalau@us.ibm.com

ABSTRACT

Web automation often involves users describing complex tasks to a system, with directives generally limited to low-level constituent actions like "click the search button." This level of description is unnatural and makes it difficult to generalize the task across websites. In this paper, we propose a system for automatically recognizing higher-level *interaction patterns* from user's completion of tasks, such as "searching for cat videos" or "replying to a post". We present PatFinder, a system that identifies these patterns using the input of crowd workers. We validate the system by generating data for 10 tasks, having 62 crowd workers label them, and automatically extracting 14 interaction patterns. Our results show that the number of patterns grows sublinearly with the number of tasks, suggesting that a small finite set of patterns may suffice to describe the vast majority of tasks on the web.

INTRODUCTION AND BACKGROUND

Users perform a myriad of tasks on the web, from searching for information about a microwave and scheduling vacations, to messaging friends and crowdfunding startups. Automating part or all of these tasks may help make users more efficient. Automation is particularly useful for visually impaired web users, who experience difficulty navigating web sites using existing screen reader technology. While voice-based personal assistant software such as Apple's Siri make more tasks accessible, current functionality is limited to a fixed set of tasks and no equivalent exists for general web browsing.

At the other end of the spectrum, web automation systems such as PLOW [1], CoCo [4], and Mahmud *et al* [5] enable users to automate tasks from traces of low-level actions such as clicking buttons. However these task representations are fairly low level and difficult to generalize to new tasks, or even similar tasks performed on a different vendor's website.

In this work, we propose to create web automation systems based on higher-level *interaction patterns*, such as logging in to a site, searching, and browsing search results. These patterns may correspond to design patterns used by web developers as building blocks for creating websites (e.g., http://www.welie.com). A necessary first step is to identify an appropriate set of interaction patterns and the directives users find natural to use in talking about them. In this paper, we explore a crowdsourced approach to identifying these interaction patterns and directives. Unlike the Mahmud work [5], our goal is to automate this process using the crowd, rather than using a machine learning algorithm.

By asking the crowd to identify and label these patterns, we believe that the generated labels will capture the language people use to communicate about web tasks with each other, and therefore enable the creation of Siri-like agents that can communicate with users using natural terminology.

Our experiments provide evidence for the existence of these interaction patterns, suggest that there are a finite number of them that are frequently reused on many web sites, and that the crowd agrees on how to describe them.

In this paper, we introduce PatFinder, a system that uses *the crowd* to automatically decompose tasks into patterns and associate the patterns with the low-level clickstreams that produced them. In a study involving 12 users and 62 crowd workers we demonstrate that only 14 unique interaction patterns were needed to complete 10 user-generated web tasks, suggesting that a relatively small set of patterns covers a large portion of tasks on the web. We conclude with an outline of future work that extends this system to use these interaction patterns as a way to enable natural language scripting and control of web processes using the crowd.

IDENTIFYING INTERACTION PATTERNS USING CROWDS

In order to identify and recognize interaction patterns on the web, we have developed an algorithm that leverages the crowd to collect and label interaction patterns automatically. Our algorithm starts with a video of a user performing a web task (i.e. 'buying a book about HCI'), and outputs a list of interaction pattern labels performed in the video (i.e. 'search', 'add item to cart', 'check out'). Using video allows us to easily capture low-level interaction data and system state in a format that is easily understood by the crowd labelers. The resulting pattern labels, expressed in natural language, are then post-processed using a natural language toolkit to name the underlying interaction pattern.

PatFinder begins by automatically segmenting video captured of the user's browsing session based on new web pages loaded, using the web browser's built in history logs. Each

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
UIST'12, October 7-10, 2012, Cambridge, MA, USA.
Copyright 2012 ACM 978-1-4503-1580-7/12/10...$10.00.

segment of the video contains all the actions performed within a single web page on the site. We believe that this heuristic will naturally encourage users to generate meaningful labels that describe individual interaction patterns. One of the benefits of our system is that we only need users to record their screen and use a browser with history.

We then recruit *crowd* workers from Amazon's Mechanical Turk, who are asked to view the video segments being performed by the user and provide a text label for each segment to describe the pattern being performed. We define the crowd as a dynamic group of web workers of varying reliability, available on demand. This means that no single worker can be relied upon to provide complete or correct answers. In order to prevent workers from adding too much detail about a pattern after the fact, we require them to generate labels in a single pass, without pausing or replaying the video.

EXPERIMENTAL RESULTS

To demonstrate that a consistent set of interaction patterns exists, we asked 11 users to generate tasks for 10 of the most popular sites with distinct uses: Google, Facebook, YouTube, Wikipedia, Twitter, Amazon, Blogspot, PayPal, and eBay. Of the 110 tasks, we selected 1 representative per website, then asked a separate user who had not participated in task generation to perform these 10 web tasks using Google Chrome. We recorded their screen using QuickTime.

Each video was automatically segmented using FFMPEG by extracting page-load timestamps from Chrome's browsing history, inserting 15-second 'loading' screens which contain no new content between each segment, to give workers enough time to type their label. These edited video clips were then posted to Mechanical Turk to be labeled. In total, 62 crowd workers responded, with at least 5 workers labeling each of the clips. Workers provided a total of 252 labels.

We observed that workers provided very uniform answers in terms of the patterns they described, but varied too much on terminology to fully automate the extraction of patterns from labels. Thus, responses were manually coded to find individual patterns and parameters described in the labels, using the clustering as a rough guide. In the future, we will use a package such as NLTK for Python[2] to perform clustering of the responses based on phrase similarity using a bag-of-words model, weighted using word frequencies. Preliminary tests indicated the need for more robust metrics that leverage semantic information from sources such as WordNet[6] and VerbNet[7] to handle descriptions of the same concept using different phrases (i.e. 'checking out' versus 'purchasing product') will help make automatic clustering more accurate. Additionally, prior work has shown that displaying input from other workers can promote *convergence* to a shared set of labels [3].

One concern with interaction patterns is scalability: there may be an infinite number of them needed to represent the variety of web tasks people do. To that end, we analyzed how quickly the number of patterns grows as new tasks are added. We plotted the results for the average number of patterns used in all combinations of between 1 and 10 tasks on the selected web sites. Figure 1 shows that the number of

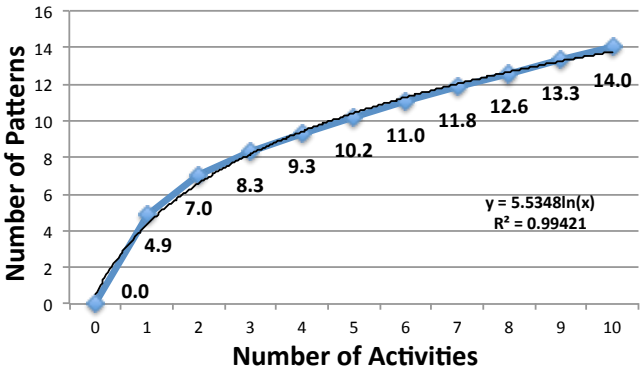

Figure 1: As the number of web tasks grows, the average number of new interaction patterns needed to complete each task increases only logarithmically.

patterns needed to accomplish web tasks grows logarithmically in the number of tasks (R^2=0.994). This suggests that a relatively small number of interaction patterns may be able to account for a large number of tasks performed on the web. In our tests we found the 14 distinct interaction patterns below:

Selecting an element	Navigate to page	Like a page
Browsing a page	Zoom an element	Leave or exit page
View or read content	Check out	Log in
Play a video	Edit a field	Post a reply
Search for a term	Put item in cart	

CONCLUSIONS AND FUTURE WORK

In this paper, we have presented PatFinder, a system that uses the crowd to identify interaction patterns in order to enable robust automation of web tasks. Our results demonstrate that the number of patterns needed to perform common tasks on a set of the most popular web sites grew logarithmically.

Future work will allow PatFinder to automatically extract parameters for each pattern, using descriptions provided by users, which provide a robust means of identifying and generalizing parameters [1]. The patterns and parameters we extract can then be associated with the original low-level interaction data collected from users, in order to enable automatic execution by the system itself. Ultimately, users could be presented with an interface that responds to natural language commands issued at the same style and level of abstraction they would use when interacting with another person.

REFERENCES

1. J. Allen, N. Chambers, G. Ferguson, L. Galescu, H. Jung, M. Swift, and W. Taysom. Plow: a collaborative task learning agent. In *AAAI 2007*, 1514–1519.
2. S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*. O'Reilly Media, 2009.
3. W. S. Lasecki, Y. C. Song, H. Kautz, J. P. Bigham. Real-time crowd labeling for deployable activity recognition. *In Submission*. 2012.
4. T. Lau, J. Cerruti, G. Manzato, M. Bengualid, J. P. Bigham, and J. Nichols. A conversational interface to web automation. In *UIST 2010*, 229–238.
5. J. Mahmud, Y. Borodin, I. V. Ramakrishnan, and C. R. Ramakrishnan. Automated construction of web accessibility models from transaction click-streams. In *WWW 2009*, 871–880.
6. G. A. Miller. WordNet: A Lexical Database for English. In *Communications of the ACM*. Vol. 38, No. 11: 39-41. 1995.
7. K. K. Schuler. Verbnet: a broad-coverage, comprehensive verb lexicon. PhD Thesis. University of Pennsylvania. 2005.