Measuring the Interaction Value of Widgets

Praphul Chandra
Hewlett-Packard Labs
India
praphul.chandra@hp.com

Geetha Manjunath
Hewlett-Packard Labs
India
geetha.manjunath@hp.com

ABSTRACT

Widgets are an increasingly popular model of web interactions, especially in mobile environments. By packaging multiple web actions into a 1-click interaction, widgets can radically simplify web interaction. In this paper, we propose a method for quantitative evaluation of the reduction in interaction complexity due to widgets. We apply our evaluation on a special class of widgets called Tasklets that model a widget as an interaction flow over websites. We then use an information theoretic definition of complexity to measure the reduction in user interaction complexity achieved by these user created widgets. Our approach is structural in nature and can thus be applied independent of the underlying semantics of the widget.

Categories and Subject Descriptors

D.1.2 [**Automatic Programming**] H.5.3 [**Web-based Interaction**] H.5.2 [**User Interfaces**]: Evaluation / methodology.

General Terms

Measurement, Human Factors, Algorithms.

Keywords

Interaction, complexity, widgets, Programming-By-Example, HCI

1. INTRODUCTION

Current web interfaces require the user to remember / search websites, navigate through web-pages, fill multiple forms - essentially figure out a sequence of actions that are needed to accomplish a particular task. Even when she succeeds in doing it once, the next time a similar goal arises, a user has to recall and manually repeat the same exercise. In contrast, 1-click access to personally valuable web tasks is a compelling user experience as demonstrated by the growing popularity of widgets and mashups. While users have validated the value of widgets in improving convenience of mobile internet [1], there is little work on quantitatively measuring the reduction in interaction complexity due to web widgets. In this paper, we propose a method for quantitative evaluation of the reduction in interaction complexity due to widgets vis-à-vis on-website interaction. We apply our evaluation on a special class of widgets called Tasklets, (a concept earlier proposed by us [2]) that models a web interaction as a sequence of browser-based web actions. We extend our analysis of navigational complexity in web interactions [3] to widgets and we propose a measure for the reduction in complexity achieved by packaged interactions like widgets, mashups & Tasklets.

The rest of the paper is organized as follows. Section 2 discusses the related work in quantitative measures proposed to measure interaction complexity. Section 3 describes our approach of modeling widgets as interaction flows over a web graph and Section 4 defines the measure for widget complexity reduction. Section 5 concludes.

WWW 2010, April 24-30, 2010, Raleigh, North Carolina.

2. RELATED WORK

Efforts to quantify user experience & usability have a rich history. One of the early works [4] to use a graph theoretic approach to measure interaction complexity used a Petri net based approach to measure the cognitive complexity of a human interaction with an interactive (dialog-based) system. The MAUSE project [5] is a collaborative effort to measure user experiences provided by graphical user interfaces in general, using multiple Usability Evaluation Methods (UEM). On similar lines, [6] has focused on analyzing the complexity of layout design on GUI usability. Our work is different from these, our specific focus is on measuring the reduction in complexity due to the packaging of web actions.

3. WIDGETS AS INTERACTION FLOWS

Figure 1: Recording browser actions to author widgets.

Figure 1 shows screenshots from a user recording web browser actions to *navigate to MSN page, go to the Horoscope page and extract content for the user's zodiac sign*. To create a personal widget, a user performs such a task (web interaction) once using her web browser by giving some sample inputs on all the relevant web sites. Our composition platform records and analyzes these browser actions to auto-generate custom personal widgets which compress all the required actions into a single-click interface (a desktop widget). Thus, in this composition platform a widget is modeled as a web interaction as a sequence of web actions needed to perform the task on different web pages. We have built a toolkit and a platform to enable end-users to create such widgets just by browsing. Figure 2 shows a graph model of a TaskLet as a navigation path over the graph of web pages w_k.

4. SIMPLIFICATION DUE TO WIDGETS

We define ***navigational complexity*** as follows: if the user is on page w_i and should ideally go to w_{i+1} to achieve her goal, the cognitive load on the user is the *choice among the number of outgoing hyperlinks*, r_i *from* w_i.

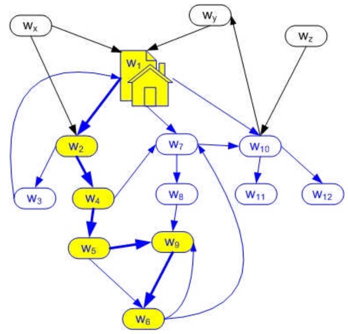

Figure 2: An interaction flow through the web (highlighted)

With a Markov assumption of web interactions, we defined [3] a measure of navigational complexity as:

$$p(w_{i+1}|w_i) = f(1 + r_i) \sim 1/(1 + r_i) \qquad (Eq.\ 1)$$

$$p(W) = p(w_1).p(w_2|w_1)\, p(w_3|w_2) \ldots\ldots p(w_k|w_{k-1}) \qquad (Eq.2)$$

$$p(w_1) = g\,(PageRank(w_1)) \qquad (Eq.\ 3)$$

The exact form of function *f()* will depend on how we differentiate among outgoing links. Under the simplifying assumption of treating all outgoing links homogeneously, eq. 1 approximates to $1/(1 + r_i)$. The value of p(W) has a range between (0, $p(w_1)$]. The first term, $p(w_1)$ captures the probability of the user reaching the starting web page of the user interaction. This can either be '1' if the starting page of the web interaction is known or it can be defined using eq. 3 if the starting point of the interaction is reached via a search engine. The exact form of *g()* is search engine dependent. For widgets (mashups) which operate across website boundaries, the function *f()* can subsume Eq.3 if there is no direct link between the two websites. Thus, the reduction in interaction complexity due to mashups includes the 'combination' of two websites into an interaction flow. Restating the above as an information theoretic measure of complexity (entropy):

$$H(w_i) = -\sum_{r_i} p(w_{i+1} \mid w_i) \log_2 p(w_{i+1} \mid w_i)$$

$$H(w_i) = -\frac{r_i}{1+r_i} \log_2 (\frac{1}{1+r_i})$$

$$H(W) = G + \sum_{2}^{k} H(w_i) \qquad (Eq.4)$$

Eq. 4 is a quantitative measure, in bits, of the reduction in interaction complexity due to a packaged web interaction like a widget. This entropy measures (in *bits*) the information content that a first time user is to be able to decide where to navigate on the web to achiever her goal. The form of G will depend on *g()* in eq. 3. We postulate that this model captures the reduction in complexity due to a simple navigational widget by automating the "flow of user actions" over a web graph. Many widgets package more complex interactions e.g. user parameter instantiation in form fields, selection from drop-down lists etc. To expand our model of a web interaction, we now let w_i represent not only a web page but also the set of potential user web interactions possible on that page i.e. $w_i = \{u_i, a_i\}$ where u_i is the web page (or the URL of the web page) and a_i is the set of user actions available on the particular web page. Thus a_i may include user actions like filling out a text-box, selecting a radio button, selecting from a drop down menu etc. To capture this, eq. 1 can be modified to $p(w_{i+1}|w_i) = h(1+r_i, a_i)$ *(Eq. 5)*. The exact form of *h()* will depend on the variation in cognitive load due to different actions (a_i) but we propose some guidelines. The fundamental notion is to evaluate each user action in terms of its contribution towards helping the user achieve her goal. In our model, a user action is evaluated in terms of its flow-selection strength over the web graph. Therefore, the reduction in complexity due to a user parameter encapsulated in a widget when used for a drop down list will be proportional to the number of elements available in the drop down menu. A purely structural approach like this has potential limitations for e.g. a radio button has a binary selection – however, it may contribute significantly in selection of the interaction flow. However, this is dependent on the interaction semantics and should be captured in the definition of *h()*.

We applied our proposed complexity measure to a few task-based web interactions and found that the results matched our own perception of the complexity of the task. For e.g. the complexity of accessing a personalized horoscope from www.msn.com is 16 bits whereas the complexity of getting weather information from www.weather.com is 8 bits. In all cases, we assumed that the user knew the starting webpage on which to start her interaction in order to achieve her goal. Also, user parameters from drop down selections are treated on par with selecting a hyperlink from all outgoing hyperlinks and thus eq.4 reduces to $1/(1+r_i + a_i)$ & complexity is $H(w_i) = -\frac{r_i + a_i}{1+r_i+a_i} \log_2 (\frac{1}{1+r_i+a_i})$)

An important insight from the analysis of this approach suggests that reduction in interaction complexity is higher for widgets which involve interactions with pages further away from the starting interaction page. One explanation may be that most websites are designed to cater to the needs of a majority of the website users - hence 'popular' information elements or links there to, are placed closer to the front page whereas content which is of interest only to some users appears further down on the site. It is specifically for interaction with these elements that widgets achieve the most significant reduction in interaction complexity.

5. CONCLUSION

We propose a measure for the reduction in interaction complexity due to web widgets. The structural nature of our approach makes it applicable to a variety of web widgets. Our future work will validate this measure with user studies & extend the proposed measure to include interaction semantics of a web widget.

6. REFERENCES

[1] Nuthall, P. et al Widgets Improve The Convenience Of The Mobile Internet, Forrester (2008). Available at http://www.forrester.com/rb/Research/widgets_improve_convenience_of_mobile_internet/q/id/45934/t/2

[2] Manjunath, G. et al. Creating Personal Mobile Widgets without Programming in Developers Track at 18th international conference on World wide web (2009)

[3] Chandra, P. & Manjunath, G. Navigational Complexity in Web Interactions. 19th international conference on World wide web (2010)

[4] Rauterberg, M. How to Measure Cognitive Complexity in Human-Computer Interaction, Cybernetics and systems research V2 (1996)

[5] MAUSE project, http://cost294.org

[6] Comber T. and Maltby J.R. Layout Complexity: Does It Measure Usability? IFIP TC13 Intl Conference on HCI (1997). pp: 623 – 626