The Evolution of a Goal-Directed Exploration Model: Effects of Information Scent and GoBack Utility on Successful Exploration

Leonghwee Teo (teo@cs.cmu.edu)
Bonnie E. John (bej@cs.cmu.edu)
Human-Computer Interaction Institute, Carnegie Mellon University
5000 Forbes Ave. Pittsburgh, PA 15213 USA

Abstract

We explore the match of a computational information foraging model to participant data on multi-page web search tasks and find its correlation on several important metrics to be too low to be used with confidence in the evaluation of user interface designs. We examine the points of mismatch to inspire changes to the model in how it calculates information scent scores and how it assesses the utility of backing up from a lower-level page to a higher-level page. The outcome is a new model that qualitatively matches participant behavior better than the original model, has utility equations more appealing to "common sense" than the original equations, and significantly improves the correlation between model and participant data on our metrics.

Keywords: ACT-R; CogTool-Explorer; Computational Model; Human-Computer Interaction; Information Foraging

Introduction

Predicting human performance to aid in the design of interactive systems is an important practical use of computational cognitive modeling. Models like SNIF-ACT 2.0 (Fu & Pirolli, 2007) and AutoCWW (Blackmon, Kitajima, & Polson, 2005) focus on predicting user exploration of websites. These models use the common concepts of label-following and information scent (*infoscent*). That is, they posit that the user's choice is partly determined by the semantic similarity between the user's goal and the options presented in the user-interface (UI). Budiu and Pirolli (2007) and Teo and John (2008) began to consider the 2-D spatial layout of the UI when predicting exploration behavior. Budiu and Pirolli (2007) reported a correlation between data and model of R^2 = 0.56 for the number of clicks to success and R^2 = 0.59 for search times in a Degree-Of-Interest (DOI) tree. Teo and John (2008) did not report correlations, but their model successfully predicted the effect of target position in 22 search tasks in a two-column format. This paper furthers this work by considering a multi-page layout of links in a website where previous information is hidden as exploration progresses.

We first describe our metrics and why they are important. We then present the tasks and the operation of a baseline model. After presenting the quantitative performance of the baseline model, we delve into some details of the model's performance to find inspiration as to how to improve the model. Finally, we present the best model found to date and discuss directions for future work.

Our Metrics

Ultimately, a UI designer would want a model to predict the range of human behavior that would be observed in the real world when using the interactive system, on metrics such as number of errors and where they occur, performance time, learning time and what was learned, effects of fatigue, environmental factors, or emotion on performance, and even levels of satisfaction or joy when using the system. No computational model is up to that task at this writing, and more modest metrics are used in current work.

For SNIF-ACT 2.0, Fu and Pirolli (2007) reported the correlation between model and participants on number of clicks on each link (R^2 = 0.69 and 0.91 for two different websites), the correlation for number of go-back actions for all tasks (R^2 = 0.73 and 0.80), and a table of percent of model runs that succeeded on each task juxtaposed with the percent of participants who succeeded on each task (R^2 = 0.98 and 0.94, calculated from Fu and Pirolli, 2007, Figure 13). The first two metrics were for models run under the model-tracing paradigm, that is, at each step the model was allowed to choose its action but was re-set to the participant's action if it did not choose what the participant chose; the last metric was for free-running models. For their free-running model, DOI-ACT, Budiu and Pirolli (2007) did not report percent success because their experiment participants completed all tasks (and the model could run to success on all but 2 of the 16 tasks), but instead reported the correlation between the model and participants for number of clicks to accomplish each task (R^2 = 0.56) and total time for each task (R^2 = 0.59).

We will report similar metrics that are both indicative of model goodness-of-fit and important to UI designers.

1. Correlation between model and participants on the percent of trials succeeding on each task (R^2*%Success*). Percent success is common in user testing to inform UI designers about how successful their users will be with their design, so a high correlation between model and data will allow modeling to provide similar information.
2. Correlation between model and participants on the number of clicks on links to accomplish each task (R^2*ClicksToSuccess*). We eliminated unsuccessful trials because some participants would click two or three links and then do nothing until time ran out whereas others continued to click (as did the model). Also, AutoCWW (Blackmon, et al., 2005) uses this metric.
3. Correlation between model and participants on the percent of trials succeeding without error on each trial (R^2*%ErrorFreeSuccess*). This measure indicates the

model's power to predict which tasks need no improvement and therefore no further design effort.

The Tasks

To test and improve our model, we chose a multi-page layout used in AutoCWW experiments (Toldy, 2009, Experiment 1), shown in Figure 1; Dr. Marilyn Blackmon generously provided the participant log files from 36 exploration tasks performed on this layout. The participants were given a search goal (at the top of each page) and had 130 seconds to complete each task. There were 44 to 46 valid participant trials recorded for each task.

CogTool-Explorer: Mechanisms & Parameters

We start our exploration with CogTool-Explorer (CT-E), developed in the ACT-R cognitive architecture (Anderson, et al., 2004) to account for the effects of 2-column layout on link choice in web search tasks (Teo and John, 2008). CT-E added ACT-R's simulated "eyes" and "hands" to SNIF-ACT 2.0 and interacts with a spatially accurate ACT-R "device model" generated by CogTool (John, Prevas, Salvucci, & Koedinger, 2004), including the position, dimension and text label of every link on a webpage.

Given a text description of a goal and a device model with at least one visible link, CT-E moves its visual attention to a link, visually encodes the text label of the link and evaluates its infoscent relative to the goal. Three ACT-R productions then compete, (1) clicking on the best link so far, (2) reading another link on this page, or (3) going back to the previous page. If CT-E decides to click on the best link it has seen so far, it looks back at that link, moves a virtual mouse pointer over it, and clicks, bringing the next webpage into the model's visual field. If it decides to go back, the previous page is brought into the model's visual field. If it decides to read another link, it moves its visual attention to the next closest link and continues. Of course, this simple see/decide/act cycle is controlled by mechanisms and parameters that can be manipulated to produce the best predictive model possible.

In more detail, CT-E uses ACT-R's "eye" as described in Anderson, et al. (2004) with Salvucci's EMMA model of visual preparation, execution and encoding (Salvucci, 2001), a long-standing implementation within CogTool. A visual search strategy adapted from the Minimal Model of Visual Search (Halverson & Hornof, 2007) guides where to move the eye. The strategy starts in the upper-left corner and proceeds to look at the link closest to the model's current point of visual attention, moderated by its noise function. This strategy will not look at a link more than once on each visit to the web page. Other noise parameters and strategies are possible (e.g., see Budiu and Pirolli, 2007), but as the strategy and noise setting from Halverson and Hornof (2007) produced good results in the two-column tasks in Teo and John (2008), the models in this paper will not vary any aspects of visual processing. Likewise, CT-E uses ACT-

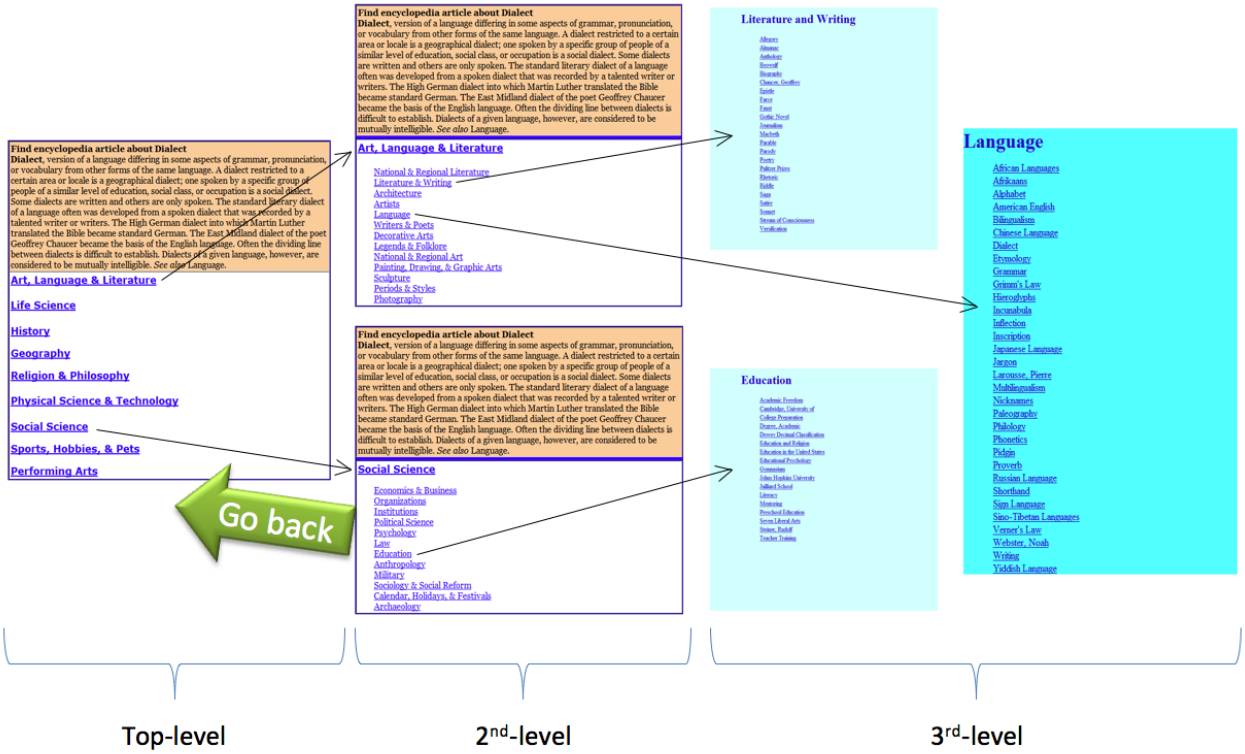

Figure 1: Multi-Page Layout from Toldy (2009). Participants start on the top-level page (leftmost) and on selecting a link, transition to 2nd-level pages. Participants may go back to the top-level page, or may select a link to go to a 3rd-level page. 3rd-level pages explicitly inform participants if they are on the correct path or not.

R's standard "hand," used in many CogTool models, and will retain that mechanism through this paper's exploration.

CT-E's estimation of information scent has used latent semantic analysis (LSA; Landauer, McNamara, Dennis, and Kintsch, 2007) to calculate the semantic relatedness of the search goal to links on the screen. We will continue using LSA throughout this paper, although other estimation procedures are possible (e.g., Fu and Pirolli (2007) and Budiu and Pirolli (2007) used pointwise mutual information). A noise function moderated the infoscent values to reflect the variability a person might display when assessing relatedness (baseline noise = ACT-R default = 1), and a scaling factor of 50 (set by Teo and John, 2008) transforms the infoscent values provided by LSA to the range of values expected by SNIF-ACT 2.0.

CT-E uses the same equations as SNIF-ACT 2.0 to decide which action to take based on what has been seen and evaluated so far, equations which also achieved good results in Teo and John (2008). These equations include two parameters, *k*, a "readiness to satisfice" factor, and the *GoBackCost*. Both of these were set to 5 in Fu and Pirolli (2007), but Teo and John's tasks required a *k* value of 600 to fit the data well, which we will continue to use here. The baseline *GoBackCost* parameter is set to Fu and Pirolli's value of 5.

Finally, when SNIF-ACT 2.0 went back to a page already seen, the link associated with the page backed-up from was marked as having been selected, and SNIF-ACT 2.0 would not select it again (not reported in Fu and Pirolli, 2007, but extracted from the SNIF-ACT 2.0 code). Presumably, since Fu and Pirolli's data come from naturalistic tasks, the link color changed when a link had been selected and thus this "perfect memory" was "in the world". This mechanism is also in CT-E's baseline model.

Performance of the Baseline CT-E Model

We ran the baseline CT-E model until the model runs converged. That is, we ran a set of 44-46 runs of each of the 36 tasks (equal to the number of valid participant trials on each task, for a total of 1649 runs in each set) and calculated the %Success for each task. We then ran an additional set, combined it with the previous set to form a new combined set and compared its values of %Success per task to the previous set's values. If all values were within 1% of each other, we considered the model converged and stopped. If any of the tasks had a %Success value greater than 1% from its counterpart in the previous set, we ran an additional set, combined it with the previous combined set to form a new combined set and compared its values of %Success per task to the previous combined set's values. The baseline model converged after 12 sets (~20,000 runs), with the following calculated values for our metrics and their 95% confidence intervals:

R^2*%Success* = 0.28 (0.21, 0.35)
R^2*ClicksToSuccess* = 0.36 (0.29, 0.43)
R^2*%ErrorFreeSuccess* = 0.44 (0.37, 0.51)

These values are disappointing for UI design because design practice requires far higher confidence in a model's predictions to be a useful alternative to user testing. These values are also substantially lower than the comparable values reported by other SNIF-ACT derivatives, SNIF-ACT 2.0's R^2*%Success* was 0.98 and 0.94 for the two websites modeled (Fu & Pirolli, 2007) and DOI-ACT's R^2*ClicksToSuccess* was 0.56 (Budiu & Pirolli, 2007).

Since the baseline CT-E model used the same utility equations and most of the same parameters as SNIF-ACT 2.0, it is necessary to understand why the R^2*%Success* results are so different. Our first hypothesis is that different data collection processes are to blame. Fu and Pirolli's (2007) data were from participants doing eight tasks on each of two websites, at their leisure, on their own computers. Their participants could abandon the task at will whereas the Toldy's tasks were collected in the lab and participants had 130s to complete each task (Toldy, 2009). Allowing the participants to abandon tasks probably eliminated the most difficult tasks with their higher variability. Not compelled to continue until success, not a single participant in Fu and Pirolli's data succeeded on 4 of their 16 tasks, in contrast to the range seen in Toldy's tasks (average *%Success*=71%, min=13%, max=100%). Since SNIF-ACT 2.0 also failed on these tasks, these four points provided a strong anchor at the origin for their R^2*%Success* value. Another major difference that might have led to better performance is that SNIF-ACT 2.0 used infoscent scores calculated with reference to only the website in the task (E. Chi, personal communication, June 18, 2010), whereas our infoscent scores were calculated with reference to the college-level TASA corpus (from Touchstone Applied Science Associates, Inc.). A corpus comprised of the task website might have produced infoscent scores with less noise (from word sense ambiguity, etc.) that the more general college-level corpus. Finally, simply switching tasks can illuminate deficiencies in any model, which will be the focus of the rest of this paper.

Inspirations for What to Change in the Model

Two glaring deficiencies in the behavior of the baseline model, relative to that of participants, inspired changes in the model. The first is that participants revisit links that they clicked before (13% of their actions) and the model never does. This means that the mechanism in SNIF-ACT 2.0 that perfectly remembers which links have been clicked on and never re-selects them must be changed to allow the possibility of matching the behavior in these data. We cannot tell from the data whether a revisit is a deliberate decision to click on the link a second time or that the participant forgot that link had been clicked (the links in this experiment did not change color when clicked); we chose to model the latter with the following mechanism in our baseline model. Each link is represented as a visual object that has a "status" attribute whose value is set to "chosen" when the link is clicked on by the model and then stored in declarative memory. ACT-R's decay mechanism governs

whether the fact that the link had been chosen will be retrieved when it is next seen and evaluated by this model. We set ACT-R's base level learning activation parameter, :bll, to 0.5 as recommended in the ACT-R tutorial (section 4.3), the retrieval activation threshold to -0.5 as shown in section 4.2, and both the permanent noise, :pas, and the instantaneous noise, :ans, to nil (section 4.5).

The second deficiency in the baseline model is that 22% of the participants' actions involve going back from a page and only 7% of the models' actions do. This behavior is comparable to Fu and Pirolli's 5% go-back actions, which, we believe matched their data because they allowed their participants to abandon tasks instead of going to completion. This calls into question the SNIF-ACT 2.0 mechanisms that govern go-back behavior, that is, both the GoBack utility equation and the *GoBackCost* parameter. We will lower the *GoBackCost* from 5 to 1 to get the exploration started and examine the GoBack utility equation with a more detailed examination of the model behavior.

After making the two fundamental changes motivated by global behavior of the baseline model (call this model *baseline++*), we guided our investigation by examining tasks where participants were least likely to be exploring in a random fashion, i.e., on tasks where participants were most successful. We sorted the 36 tasks by highest *%ErrorFreeSuccess* and then focused on the top four tasks.

The third task in this list, to search for information about pigeons (correct top-level link = "Life Sciences", correct 2nd-level link = "Birds") had infoscent scores that were all very low and not widely distributed for the top-level headings. Budiu and Pirolli (2007) discuss this problem as well; misleading and/or non-discriminating infoscent scores will plague any model and we did not consider this task further for inspiration about what to change. However, the other three tasks inspired three ways to change the baseline++ model.

Refinement of Infoscent Values for Top-level links

The topmost task was to search for information about ferns and its correct top-level link was "Life Sciences". The 46 participants only selected other top-level links 8% of the time and but went back from those 2nd-level pages to select "Life Science" and then "Plants" (in all but 2 cases) to complete the task. In contrast, the baseline++ model selected other top-level links about 70% of the time before selecting "Life Sciences", and on some model runs it never selected "Life Sciences" and failed the task.

One possible explanation for the model behavior was that it did not look at "Life Science" before deciding to select a link on the top-level page. When we examined the details of the model runs, this was not the case, as the model runs did see "Life Science" before selecting a link in over 95% of first-visits to the top-level page. A second possible explanation was that the model looked at too many links and saw other higher infoscent links before selecting a link on the top-level page. This also was not the case because, in all model runs up to the point where it finished looking at "Life Science", if we forced the model to choose the best link so far, it would have selected "Life Science" in over 60% of the runs. A third possible explanation lies in the infoscent values used by the model.

Given a particular goal, the baseline models followed AutoCWW (Blackmon, et al., 2005) by using LSA to compute an infoscent value for each link, based on the cosine value between two vectors, one representing the words in the goal description and the other the words in the link text. To approximate how a reader elaborates and comprehends the link text in relation to his or her background knowledge, AutoCWW adds all the terms from the LSA corpus that have a minimum cosine of 0.5 with the raw text and a minimum word frequency of 50 to the raw link text before using LSA. Kitajima, Blackmon and Polson (2005) explained that "elaborated link labels generally produce more accurate estimates of semantic similarity (LSA cosine values)." Our baseline model used the same method, thus, for the link "Life Science", the words *"science sciences biology scientific geology physics life biologist physicists"* were added and then submitted to LSA to compute the infoscent value.

AutoCWW uses a further elaboration method motivated by UI layouts with links grouped into regions labeled with a heading. Kitajima et al. (2005) explained that "readers scan headings and subheadings to grasp the top-level organization or general structure of the text". To represent a region, AutoCWW first elaborates the heading text as described in the previous paragraph, and then adds all the text and their elaborations from links in the same region. The baseline model did not use this elaboration method for top-level links because their subordinate links appeared on 2nd-level pages, different from Kitajima et al.'s assumption. However, participants did practice trials on the same multi-page layout as the actual trials, and perform all 36 test trials on the same layout. Therefore, we would expect that this experience would influence how participants assessed infoscent of the top-level link. This reasoning motivated our first refinement to the baseline++ model to better represent these participants: for the infoscent of a top-level link, we elaborate the top-level link and then add the text from all links in the corresponding 2nd-level page. While this refinement is similar to AutoCWW's procedure, the justifications are different. This refinement is also in line with Budiu and Pirolli's (2007) use of category-based scent, but approximates their human-generated categories with an automated process.

Refinement of Mean Infoscent of Previous Page

The second task on our list was to search for information about the Niagara River. The baseline++ model selected the correct link "Geography" on the top-level page, but went back from the 2nd-level "Geography" page over 60% of the time, while participants never did. To investigate, we looked at how the model decided to go back. Recall that like SNIF-ACT 2.0, after looking at and assessing the infoscent of a link, the baseline CT-E models choose between reading

another link, selecting the best link seen so far, or going back to the previous page using utility functions. The utility functions of reading another link and selecting the best link so far have both strong theoretical support (Fu & Pirolli, 2007) and empirical support from several studies that did not use or emphasize go-back behavior (Fu & Pirolli, 2007 and Teo & John, 2008). However, the utility function for going back has less support and was therefore a focus of our attention. From SNIF-ACT 2.0, the baseline CT-E models used the following GoBack utility equation.

$$\begin{aligned} \text{Utility}_{\text{GoBack}} &= \text{MIS(links assessed on previous page)} \\ &\quad - \text{MIS(links assessed on current page)} \\ &\quad - \text{GoBackCost} \\ &\textit{where MIS is Mean Information Scent} \end{aligned} \qquad \text{[Eq. 1]}$$

The infoscent values for the nine top-level links are sensible: the correct link, "Geography", has the highest LSA value by an order of magnitude. After selecting the top-level link with the highest infoscent and visiting the corresponding 2nd-level page, Eq. 1 includes "Geography's" high scent in its first operand, which attracted the model back to the top-level page. This behavior violates common sense; since the model had just selected the best top-level link to visit its 2nd-level page, it should not be pulled back to the previous page by the infoscent of the selected link. This reasoning inspired another refinement to the baseline++ model, changing Eq. 1 to Eq. 2:

$$\begin{aligned} \text{Utility}_{\text{GoBack}} &= \text{MIS(links assessed on previous page} \\ &\quad \textbf{excluding the selected link)} \\ &\quad - \text{MIS(links assessed on current page)} \\ &\quad - \text{GoBackCost} \\ &\textit{where MIS is Mean Information Scent} \end{aligned} \qquad \text{[Eq. 2]}$$

Refinement of Mean Infoscent of Current Page

The last task on our list of four was to find information about the Hubble Space Telescope. While both participants and model in this task selected the correct link "Physical Science & Technology" on the top-level page, the model went back from the corresponding 2nd-level page 50% of the time, but participants never did. Inspection of the model runs in the Hubble task revealed a different problem from that in the Niagara River task, however. After selecting the link with the highest infoscent and visiting the corresponding 2nd-level page, if the first link the model saw on that page had very low infoscent, the GoBack utility would be high because the value of the second operand would be low. This behavior also violates common sense; since the model had just selected the best link on the top-level page because it looked promising, the model should carry that confidence into the next page and should not immediately go back just because the first link it saw on the 2nd-level page did not relate to the task goal. This reasoning inspired our last refinement to the baseline++ model, changing Eq. 2 to Eq. 3:

$$\begin{aligned} \text{Utility}_{\text{GoBack}} &= \text{MIS(links assessed on previous page} \\ &\quad \textbf{excluding the selected link)} \\ &\quad - \text{MIS(links assessed on current page)} \\ &\quad \textbf{including the selected link)} \\ &\quad - \text{GoBackCost} \\ &\textit{where MIS is Mean Information Scent} \end{aligned} \qquad \text{[Eq. 3]}$$

This change has a nice symmetry with the previous change, carrying along the "confidence" inspired by the high infoscent top-level link. If the selected link's infoscent score is very high compared to the other top-level links, those other top-level links alone will not exert much pull to go back. If the selected link's infoscent score is high relative to the first few links it sees on the 2nd-level page the model will not go back until it "loses confidence" by seeing several low infoscent links, thereby diluting the effect of the high infoscent link that led the model to this page.

We ran one set of many preliminary models to get a feel for the contributions of these changes. The combination of all changes described here seemed to be the best model.

Performance of the Best Model So Far

With all the changes described above combined, we ran the model to convergence (10 sets, a total of 16490 runs), and attained the following calculated values for our metrics and

Table 1. Summary of Results. Gray shading indicates mechanism and parameters that did not change.

Mechanism, Parameter, or Metric	Baseline Model	Best Model So Far
Visual processes	ACT-R + Salvucci, 2001 + Halverson & Hornoff, 2007 [2]	No change
Manual processes	ACT-R [2]	No change
Information Scent Process		
Heading-level input	link labels	link labels + lower link labels
Link-level input	link labels	No change
Decision Process		
Click best link utility eq	SNIF-ACT2.0[1]	No change
k (readiness to satisfice)	600[2]	No change
Read next link utility eq	SNIF-ACT2.0[1]	No change
GoBack utility equation	SNIF-ACT2.0: Eq. 1[1]	Improved here Eq. 3
GoBackCost	5[1]	1
Memory of selected links	Perfect[1]	Imperfect :bll = 0.5 :rt = -0.5 :ans = nil :pas = nil
Metrics		
R^2%*Success*	0.28 (0.21, 0.35)	0.72 (0.66, 0.76)
R^2*ClicksToSuccess*	0.36 (0.29, 0.43)	0.66 (0.60, 0.71)
R^2%*ErrorFreeSuccess*	0.44 (0.37, 0.51)	0.82 (0.79, 0.85)

[1] from Fu & Pirolli, 2007
[2] from Teo & John, 2008

their 95% confidence intervals (Table 1):

R^2*%Success* = 0.72 (0.66, 0.76)

R^2*ClicksToSuccess* = 0.66 (0.60, 0.71)

R^2*%ErrorFreeSuccess* = 0.82 (0.79, 0.85)

Discussion and Future Work

The improved model presented above made large and significant improvements on all our metrics over the baseline model coming into this investigation. R^2*%Success* more than doubled and the other two metrics increased by more than 50%. Although there is room for improvement, these values are in the range where UI designers could use them to identify the tasks at the extremes. That is, this analysis identifies which tasks are sufficiently supported by the interface that effort can be diverted to other areas and which tasks are in most need of attention.

Future work will take at least two paths. First we must systematically explore the benefits of the model mechanisms and parameters described in this paper. We have presented only the conjunction of these elements, with a single set of parameters, but we will examine the mechanisms' individual and pairwise effects on model performance and explore the parameter space before moving on to other UI layouts and tasks.

Second, we should reconsider the metrics and how to use them. Although we believe the metrics presented here are both meaningful for goodness of fit and useful for UI design, other metrics should be considered. For example, Fu and Pirolli (2007) reported the correlation between the number of go-back actions by the model and participants; how might this help inform model improvements or design? As a second example, consider root mean square error (RMS error), a standard metric for quantifying the difference between the values estimated by a model and what is observed in empirical trials. UI designers often need to know absolute quantities when making decisions about design and development effort and cost trade-offs. Thus, a low RMS error would be as valuable as a high correlation (the RMS error did reduce for each metric with our improved model, but are not yet <20% which is desirable for UI design practice). In addition, we need to understand how to combine or trade-off metrics against one another, as it is unlikely that model exploration will produce the most desirable levels of all metrics at once.

In the meantime, AutoCWW has shown it could be used to improve the design of website links with only 54% of the variance explained for *ClicksToSuccess* (Blackmon, et al., 2005) and this improved version of CogTool-Explorer exceeds that level. If these results can be shown to extend beyond simple web search tasks, to other layouts, types of interfaces, and tasks, CogTool-Explorer will be well on its way to being a useful tool for design.

Acknowledgments

The authors thank the anonymous reviewers whose probing questions improved the science reported in this paper and Dr. Marilyn Blackmon for sharing the experiment data. This research was supported in part by funds from IBM, NASA, Boeing, NEC, PARC, DSO, ONR, N00014-03-1-0086. The views and conclusions in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of IBM, NASA, Boeing, NEC, PARC, DSO, ONR, or the U.S. Government.

References

ACT-R 6.0 Tutorial (June, 2010) Available for download at http://act-r.psy.cmu.edu/actr6/units.zip, June 13, 2010.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review. 111*, 4, 1036-1060.

Blackmon, M. H., Kitajima, M., & Polson, P. G. (2005). Tool for accurately predicting website navigation problems, non-problems, problem severity, and effectiveness of repairs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. ACM, New York, NY, 31-40.

Budiu, R. & Pirolli, P. (2007), Modeling navigation in degree-of-interest trees. In N.A. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31th Annual Conference of the Cognitive Science Society* (pp. 845-850). Austin, TX: Cognitive Science Society.

Fu, W.-T., & Pirolli, P. (2007). SNIF-ACT: A cognitive model of user navigation on the World Wide Web. *Human-Computer* Interaction*, 22,* 355-412.

Halverson, T. & Hornof, A. J. (2007). A minimal model for predicting visual search in human-computer interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '07. ACM, New York, NY, 431-434.

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '04. ACM, New York, NY, 455-462.

Kitajima, M., Blackmon, M. H. & Polson, P. G. (2005). Cognitive architecture for website design and usability evaluation: Comprehension and information scent in performing by exploration. *HCI International 2005*.

Landauer,T. K., McNamara, D. S., Dennis, S., & Kintsch W. (Eds). (2007). *Handbook of Latent Semantic Analysis*.

Salvucci, D. D. (2001). An integrated model of eye movements and visual encoding. *Cognitive Systems Research*, *1*, 201-220.

Teo, L., & John, B. E. (2008). Towards a tool for predicting goal-directed exploratory behavior, *Proceedings of the Human Factors and Ergonomics Society 52nd Annual Meeting* (pp. 950-954). Santa Monica, CA: Human Factors and Ergonomics Society.

Toldy, M. E. (2009) The Impact of Working Memory Limitations and Distributed Cognition on Solving Search Problems on Complex Imformational Websites. Unpublished Doctoral Dissertation, University of Colorado – Boulder, Department of Psychology.