Using Eye-tracking Data for High-Level User Modeling in Adaptive Interfaces

Cristina Conati[1], Christina Merten, Saleema Amershi[2], Kasia Muldner[1]

[1]University of British Columbia,
2366 Main Mall, Vancouver, BC, V6T1Z4, CANADA
[conati,kmuldner]@cs.ubc.ca

University of Washington
Box 352350, Seattle, WA 98195-2350, USA
saleema_amershi@hotmail.com

Abstract

In recent years, there has been substantial research on exploring how AI can contribute to Human-Computer Interaction by enabling an interface to understand a user's needs and act accordingly. Understanding user needs is especially challenging when it involves assessing the user's high-level mental states not easily reflected by interface actions. In this paper, we present our results on using eye-tracking data to model such mental states during interaction with adaptive educational software. We then discuss the implications of our research for Intelligent User Interfaces.

Introduction

One of the main challenges in devising agents that can act intelligently is to endow them with the capability of understanding the behaviors of other agents (human or artificial) that are part of their environment. Thus, AI has a long history of research on *plan recognition*, i.e., how to infer an agent's goals and intentions from its observable behaviors. One of the applications of this research has been in Intelligent User Interfaces (IUI), more specifically in devising *adaptive interfaces* that can understand a user's needs (i.e., perform *user modeling*) and act accordingly so as to improve the user's interaction experience. However, providing meaningful adaptivity often requires understanding more than a user's plans. Depending on the nature of the interaction task, the user traits that an interface agent may need to model include domain-dependent cognitive skills (e.g., knowledge of relevant concepts and procedures), meta-cognitive processes that cut across tasks and domains (e.g. general learning and reasoning strategies), and affective states (e.g. emotions and moods). Arguably, the higher the level of the traits to be captured, the more difficult they are to assess unobtrusively from simple interaction events. This problem has generated a stream of IUI research on using innovative sensing devices to enrich the information available for user modeling and interaction adaptation.

The work we report here contributes to this research with our results on using real-time eye-tracking to capture `user attention patterns, in order to inform a user model designed to assess student meta-cognitive behavior during interaction with an Intelligent Learning Environment (ILE). ILEs are educational systems that provide personalized instruction, much as human educators do to accommodate individual students' needs. Personalizing the instruction to target the meta-cognitive processes relevant for learning is important because it can help students improve their overall learning ability in addition to domain-specific skills. However, assessing user meta-cognitive behaviors is difficult, because they cannot usually be inferred from simple interface actions. For this reason, we are exploring the use of eye-tracking data to provide a user model with information on user meta-cognition. In the research described here, we specifically targeted the meta-cognitive skills related to learning from free exploration [11] and to *self-explaining* instructional material, i.e., clarifying the material to oneself in light of the underlying domain theory (e.g., [3]).

There is a well-established body of research on using eye-tracking data for off-line evaluation of interface design [7], or as an alternative form of input to allow a user to explicitly operate an interface [7,10]. However, research on real-time usage of this type of data to enable on-line adaptation of the interaction is still in its infancy. Some of this work uses gaze tracking to understand what a user is doing (e.g., reading email vs. reading a web page) [6], or to help assess user *task performance* (e.g., reading performance for automatic reading remediation [12]) Others have explored using gaze data to assess user *mental states* such as *interest* in various elements of an interactive story [13], *attention* in the context of assessing learner motivation during interaction with an ILE [9] and student *problem-solving strategies* in a tutoring system for algebra [5]. We contribute to this research by showing how eye-tracking can improve recognition of user meta-cognitive skills during interaction with the Adaptive Coach of Exploration (ACE), an ILE that supports exploration-based learning of mathematical functions. The main contribution of our work is a formal evaluation of ACE's user model [4,8], showing that the inclusion of eye-tracking information significantly improves the model assessment of both student self-explanation and exploration, compared with models that use lower-level predictors such as number of actions and action latency.

Research Summary

The Ace Environment

The ACE environment provides a series of activities to help students understand function-related concepts through exploration. Figure 1 shows the main interaction window for the Plot Unit, the activity we have used as a testbed for our research. In the Plot Unit, a learner can explore the relationship between a function's plot and equation by moving the plot in the Cartesian plane and observing the effects on the equation (displayed below the plot area in Figure 1). The student can also change the equation parameters and see how the change affects the plot.

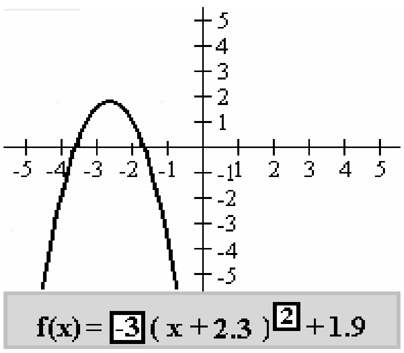

Figure 1: ACE's Plot Unit

Each function type (e.g., constant, linear and power) has an associated set of 'exploration cases' that together illustrate the full range of function attributes. For example, linear functions are defined by two parameters, the function slope and the y-intercept. Therefore, in order to gain a broad understanding of linear functions, the student should study positive and negative intercepts, and positive, negative and zero slopes.

ACE's original version included a model of student exploration behavior that helped the system generate tailored interventions to improve those behaviors deemed to be suboptimal [2]. The model is a Dynamic Bayesian Network (DBN) that includes (i) nodes to represent all possible exploration cases; (ii) nodes to represent student understanding of related mathematical concepts; (iii) links representing how exploration of relevant cases relates to concept understanding. To assess whether a case has been explored effectively, this version of the ACE model used solely evidence of the occurrence of student plot moves and equation changes.

ACE used this model to warn the student when the model predicted insufficient exploration of a given exercise, and also to provide more detailed suggestions on which specific exploration cases the student should attend to. Initial studies of this version of the system generated encouraging evidence that it could help students learn better from exploration [2]. However, these studies also showed that sometimes ACE overestimated students' exploratory behavior, because its user model considered interface actions to be sufficient evidence of good exploration, without taking into account whether a student was reasoning, or *self-explaining* the outcome of these actions. For instance, a student who quickly moves a function plot around the screen, but never reflects on how these movements change the function equation, performs many exploratory actions but can hardly learn from them. The initial ACE user model would likely judge this type of behavior as good exploration.

Using Gaze-data in ACE's User Model

To overcome the aforementioned limitation in ACE's original user model, we extended the model to include the explicit assessment of student self-explanation of exploration cases. We also extended ACE's interface so that, if the model predicts that the cause of suboptimal exploration is lack of self-explanation, then the system generates hints specifically designed to correct this behavior.

But how can the ACE model assess a student's spontaneous self-explanation, i.e., something that happens in the student's head? One predictor that is simple to obtain from log data is the time between exploration cases. This predictor is not very reliable, however, because a student may be distracted rather than engaged in reasoning during a long interval between exploration cases. Thus, we decided to study whether the addition of specific eye-gaze patterns would provide a better estimate of self-explanation. The data for our investigation was obtained through a user study during which each of 36 participants had his/her gaze monitored via an eye-tracker while interacting with ACE [4]. As part of the study, participants took a pre-test and a post-test on mathematical functions. While using ACE, the students were asked to verbalize all of their thoughts, and their interface actions were logged and synchronized with data from the eye tracker. We then used this data to include both latency between actions and gaze patterns as predictors of self-explanation for ACE. Gaze data was limited to gaze shifts between the plot and equation area after a plot move or equation change, which intuitively should indicate the student's reflection after these actions. Gaze data may also be relevant in relation to other interface actions available in the Plot Unit (e.g., moving to a new function, accepting the system's suggestion to further explore the current function; see [1] for a complete list). This work, however, was limited to equation and plot changes as a proof of concept, because of the effort required to generate the hand-labeled data necessary to train the model.

Two researchers (to assure coding reliability) categorized student verbalizations after equation and plot changes as instances of student reasoning vs. speech not conducive to learning. Then, they mapped these verbalizations onto presence/absence of gaze shifts and latency until the next action, to obtain frequency data that we used to add gaze patterns and time as independent predictors of student self-explanation in ACE's DBN user model. We then formally evaluated the new model against (i) a model using only time as predictor of self-explanation and (ii) the original ACE model that ignores self-explanation and uses only the number of interface actions as a predictor of effective exploration. Table 1 shows the results of a leave-one-out cross-validation over the 36 study participants (see [4, 8] for additional data analysis). This procedure involved isolating each of the 36 participants in turn, setting model parameters using the data from all remaining 35 participants, and then using the resulting model to assess self-explanation and exploration behavior for the test subject. Table 1 reports the average accuracy over all 36 participants. Here model accuracy is the average of the model's sensitivity (or true positive rate) and specificity (or true negative rate). We found that

- The model including both gaze and time data provides better assessment of student self-explanation than the model using only time. The difference is statistically significant ($p < 0,05$, one-tailed t-test).
- Assessing self-explanation improves the assessment of student exploratory behavior, and the accuracy of the latter increases with increased accuracy of self-explanation assessment. All improvements are statistically significant (as per ANOVA analysis with adjusted t-tests for pairwise comparisons)

	No SE	**Time only**	**Time and gaze**
Accuracy on self-explanation	N/A	67.2%	76.4%
Accuracy on exploration	68.4%	70.4%	77.5%

Table 1: Model accuracies with different types of evidence

Discussion and Implications for IUI

We believe these results are important because they provide support for the effort required to build the type of sophisticated user modeling described here. In the early 1980s, John Self, one of the precursors of user modeling research, wrote a seminal paper pointing out the challenges involved in providing a computer system with a good understanding of its users [14]. He gave "*do not overmodel*" as one of the key solutions to these challenges. Do not overmodel means including in a user model only information that is *useful*, either because the system can directly use it to adapt the interaction to the user, or because it increases the model's accuracy along other dimensions important for effective adaptation.

As simple as this guideline seems, it is not always obvious what it means in the context of specific applications. In particular, there are few research results showing when and how the complexity of modeling one or more additional user traits/behaviors in an adaptive system is worth the effort. We showed that adding information on user-attention patterns to ACE's user model is indeed useful because it increases the model's accuracy in assessing user self-explanation. Accurate assessment of self-explanation, in turn, is useful both because ACE can directly use it to generate more precise interventions to help students explore effectively, and because it improves the model's accuracy in assessing student exploration, the main meta-cognitive process that ACE is designed to stimulate. Thus, we have provided ground for further investigating eye-tracking data on user attention as a form of non-intrusive evidence when it is important to model the user's high-level mental processes in an adaptive interface.

Next we want to investigate *how* important it is to have fine-grained information on the user's high-level mental states in the type of interaction supported by ACE. Obtaining this information for ACE's user model required understanding both the connection between exploration and self-explanation, as well as the connection between self-explanation and its potential predictors. We will refer to this model as *knowledge-based*, because, although it is partially built from data, these data were- hand-labeled by means of laborious protocol analysis. We want to explore how this fine-grained user model compares with a coarser model based on a fully data-based approach on the same type of data, i.e., interface actions and gaze information. By fully data-based, we mean an approach that does not require hand-labeled data (as is the case for unsupervised machine learning techniques). The advantage of fully data-based approaches is that they are less resource-intensive because they do not rely on human knowledge. The disadvantage is that the models they generate tend to be black boxes, which do not allow one to understand the relationship between input data and model prediction. We want to investigate how this reduction in model information affects the quality of interaction adaptation.

We have taken a step in this direction by exploring a fully data-based approach to develop a student model for ACE [1]. This approach combines unsupervised and supervised machine learning to automatically develop user models from interaction data. Unsupervised clustering is used to identify classes (clusters) of interaction behaviors that can influence interaction effectiveness. Supervised machine learning is applied to these clusters for on-line recognition of user types. This approach is especially promising for applications that support novel, unstructured activities, because it does not require human experts to foresee how the many possible user behaviors may relate to interaction efficacy (as would be necessary with a knowledge-based approach). Instead, since unsupervised clustering requires representing individual user interactions as multidimensional vectors of raw data features, human input is limited to deciding which features to use. Because adding features does not involve substantial extra work, it is possible to consider a much larger variety of features than when using knowledge-based approaches. For instance, when we applied this approach to ACE, we extracted from the log data described in the previous section features for all the 13 exploration actions available in the Plot Unit (e.g., overall number, as well as mean and standard deviation of the latency after each action type). We also included the presence/absence of gaze shifts after each of the possible actions. We found that by using these features, unsupervised clustering (based on the k-means algorithm in this particular application) was able to distinguish two groups of students, one including students who learned well with ACE, one including students with poor learning. Post-hoc cluster analysis revealed that distinguishing interaction behaviors included both the intuitive patterns we analyzed in the previous section (i.e., gaze shifts after equation changes), as well as less intuitive combinations of behaviors that would have been difficult to identify via intuition or observation [1]. When we trained a supervised classifier on the two student clusters identified in the unsupervised phase, it achieved an encouraging level of accuracy in dynamically classifying students as successful vs. unsuccessful learners as they interacted with ACE.

Although these results are preliminary and need to be validated with larger datasets, they begin to show the potential of data-based approaches to reduce development effort for user modeling in novel interaction tasks with rich input data. Still, the 2-level classifier model learned using this approach suffers from the drawback common to many purely data-based models of being a black box that provides limited information on why certain patterns of input data result in a given user classification. For instance, it does not allow for isolating the specific behaviors that are causing the student to be classified in a specific cluster at any given time. Thus, if ACE were to use this model, it would not be able to generate precise interventions targeting the suboptimal behavior(s) that cause a student to be a poor learner. We plan to investigate the impact of this loss of precision on the effectiveness of ACE's adaptive interventions by empirically comparing the interactions supported by the more detailed knowledge-based model and by the coarser data-based model. There have been very few such studies to date, but we believe they are key to understanding how different AI techniques can contribute to the design of interfaces capable of effective autonomous adaptation, and to generally improve Human Computer Interaction.

Acknowledgments

We thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and the BC Innovation Council for funding this research. We also thank David Ternes for developing the algorithm that parses raw eye-tracker data.

References

1. Ameshi S. and Conati C. (2007). Unsupervised and Supervised Machine Learning in User Modeling for Intelligent Learning Environments. *Proceedings of the 2007 Int. Conf. on Intelligent User Interfaces*, Honolulu, HW, ACM Press.
2. Bunt, A. and C. Conati (2003). Probabilistic Student Modeling to Improve Exploratory Behaviour. *Journal of User Modeling and User-Adapted Interaction*, 13(3).
3. Chi, M.T.H., (1996). Constructing self-explanations and scaffolded explanations in tutoring. *Applied Cognitive Psychology*, 10, 33-49.
4. Conati, C and Merten C. (2007). Eye Tracking for User Modeling in Intelligent Learning Environments: an empirical evaluation. In *Knowledge-Based Systems*, Kluvier. Forthcoming.
5. Gluck, K.A. and J.R. Anderson (2001). What Role do Cognitive Architectures Play in Intelligent Tutoring Systems? *Cognition and Instruction: Twenty-five Years of Progress*, D. Klahr and S.M. Carver, Editors. Erlbaum.
6. Iqbal, S.T. and B.P. Bailey (2004). Using Eye Gaze Patterns to Identify User Tasks. Proc. of *The Grace Hopper Celebration of Women in Computing*, 2004.
7. Jakob, R. (1998). *The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look at is What You Get*. Morgan Kaufmann Press: San Francisco.
8. Merten C and Conati C. (2006). Eye-Tracking to Model and Adapt to User Meta-cognition in Intelligent Learning Environments. *Proc. of IUI 06, Int. Conf. on Intelligent User Interfaces*, Sidney, Australia, ACM Press.
9. Qu, L. and L. Johnson (2005). Detecting the Learner's Motivational States in an Interactive Learning Environment. *Proc. of 12th Int. Conf. on AI in Education*. Amsterdam, The Netherlands. IOS Press.
10. Salvucci, D. and J. Anderson. *Intelligent Gaze-Added Interfaces* (2000). *Proc. of Human Factors in Computing Systems, CHI 2000*, New York: ACM Press.
11. Shute, V.J., (1993). A comparison of learning environments: All that glitters..., *Computers as Cognitive Tools*, S. Lajoie, P. and S. Derry, Editors, Lawrence Erlbaum Associates: Hillsdale, NJ.
12. Sibert, J.L., M. GokTurk, and R.A. Lavine (2000). The Reading Assistant: Eye Gaze Triggered Auditory Prompting for Reading Remediation. Proc. of the 13th Annual ACM Symposium on User Interface Software and Technology. San Diego, California: ACM Press.
13. Starker, I. and R.A. Bolt, A Gaze-Responsive Self-Disclosing Display (1990). *Proc. of Human Factors in Computing Systems, CHI 1990*, New York: ACM Press.
14. Self, J. Bypassing the intractable problem of student modeling. In C. Frasson and G. Gauthier (eds.). *Intelligent tutoring systems: at the crossroad of artificial intelligence and education*. Norwood, NJ: Ablex Publishing Corporation.