AIED 2013 Workshops Proceedings
Volume 2

Scaffolding in Open-Ended Learning Environments (OELEs)

Workshop Co-Chairs:

Gautam Biswas
Vanderbilt University, Nashville, TN. USA

Roger Azevedo
McGill University, Canada

Valerie Shute
Florida State University, FL. USA

Susan Bull
University of Birmingham, UK

https://sites.google.com/site/scaffoldingoeles/home

Preface

Open-ended learning environments (OELEs) offer students opportunities to take part in authentic and complex problem solving and inquiry tasks by providing a learning context and a set of tools for exploring, hypothesizing, and building their own solutions to problems. Also referred to as exploratory environments, examples include hypermedia learning environments, modeling and simulation environments, microworlds, scientific inquiry environments, and educational games featuring open worlds. OELEs may be characterized by choices students have as they are involved in their learning and problem solving tasks; in OELEs, students are faced with a multitude of decisions about what, when, and how to learn. Naturally, these choices offer critical opportunities for students to exercise higher-order skills that include:

- *Cognitive processes* for accessing, organizing, and interpreting information, constructing problem solutions, and assessing constructed solutions;
- *Metacognitive monitoring and self-regulatory processes* for coordinating the use of cognitive processes and reflecting on the outcome of solution assessments; and
- *Emotional and motivational self-regulatory processes* that include curiosity and persistence, especially in the face of difficulty.

This presents significant challenges to novice learners because they may not have the proficiency for using the system's tools, nor the experience and understanding necessary for explicitly monitoring and regulating their emotions and behaviours as they pursue learning goals. Not surprisingly, research has shown that novices often struggle to succeed in OELEs. Without *adaptive scaffolds*, these learners typically use tools incorrectly, adopt sub-optimal learning strategies for goal selection and planning, and fail to regulate key cognitive, motivational, and emotional processes. Adaptive scaffolds in OELEs refer to actions taken by the learning environment, based on the learner's interactions, intended to support the learner in completing a task and understanding the topic. Broadly, providing adaptive scaffolds consists of two sub-problems: (1) measuring and interpreting student behaviours to determine which adaptive scaffolds will be beneficial for their learning, and (2) providing adaptive scaffolds that effectively support student needs.

Given the developing interest in this area, this workshop sought papers on: (1) theoretical frameworks for designing scaffolding; (2) implementations of adaptive scaffolds; (3) cognitive, metacognitive and self-regulation models for designing scaffolds; and (4) formative assessments that support students' learning, performance, and learning-related behaviors. 14 papers have been accepted for this workshop: 8 as long papers that have each been allocated 8 pages, and 6 as short papers that have each been allocated 4 pages in the workshop proceedings.

A number of the accepted papers present games for learning science and math content as an open-ended learning environment where students have choice in constructing their own solutions to targeted problems. However, when the system detects non-optimal or incorrect behavior, it provides adaptive scaffolds to help the students discover and correct their incorrect solutions. Some of the papers discuss scaffolds in the form of representation schemes and selective tasks assigned to the student that aid their learning processes. Other papers use machine learning and data mining techniques to analyze student activity data and determine their learning behaviors and approaches to solving problems. A few papers adopt self-explanation as the framework for providing adaptive scaffolds, while others use Open Learner Modeling (OLM) as a mechanism for promoting student reflection, planning, and decision-making. One of the papers uses scaffolding to help students improve their metacognitive judgments. Another paper studies the effect of scaffolding as students work on invention activities related to data analysis. Finally, we also have a paper that discusses taxonomy of adaptive scaffolds in computer-based learning environments. We hope this set of papers leads to interesting and important discussions, and all participants can take away something that benefits their own work and advances the state of the art in this very important field of research.

In addition to the paper presentations and discussion, this workshop features other events:

1. A combined 90 minute hands-on activity and demonstration session where participants create levels to target and assess specific competencies in the Newton's Playground game (see http://www.gameassesslearn.org/newton/; the system has a level editor built into the game environment).
2. In the second half of the demonstration session, participants can demonstrate their creations.
3. A panel, where we compare and contrast approaches to scaffolding in traditional ITS problem solving environments and OELEs.

This workshop is the next in the series of Intelligent Support in Exploratory Environments (ISEE) Workshops that started in EC-TEL ’08 and has had representations in previous AIED, ITS and ICLS conferences. The last workshop was held at the Intelligent Tutoring Systems (ITS-2012) conference in Chania, Greece in June, 2012 (https://sites.google.com/a/lkl.ac.uk/isee/isee-its-12). Finally, we would like to acknowledge the contributions of all of the authors, without which this workshop would not have taken place. Many thanks to the program committee that helped review the submitted papers and provide valuable feedback to the authors. Last, but not the least, a special thanks to James Segedy, who helped put together the Workshop proceedings.

July 9, 2013

Gautam Biswas, Roger Azevedo, Valerie Shute, and Susan Bull

Program Committee

Workshop Co-Chairs

Gautam Biswas (Vanderbilt University): gautam.biswas@vanderbilt.edu
Roger Azevedo (McGill University): roger.azevedo@mcgill.ca
Valerie Shute (Florida State University): vshute@fsu.edu
Susan Bull (University of Birmingham): s.bull@bham.ac.uk

Workshop Committee

Vincent Aleven, Carnegie Mellon University

Bert Bredeweg, University of Amsterdam

Cristina Conati, University of British Columbia

Sergio Gutiérrez-Santos, London Knowledge Labs

Judy Kay, University of Sydney

Susanne Lajoie, McGill University

James Lester, North Carolina State University

Rose Luckin, London Knowledge Labs

Manolis Mavrikis, London Knowledge Labs

Bruce McLaren, Carnegie Mellon University

Ido Roll, University of British Columbia

James Segedy, Vanderbilt University

Philip Winne, Simon Fraser University

Table of Contents

Digital Games and Science Learning: Design Principles and Processes to Augment Commercial Game Design Conventions

Douglas B. Clark, Stephen Killingsworth, Mario Martinez-Garza, Grant Van Eaton, Gautam Biswas, John Kinnebrew, Pratim Sengupta, Kara Krinks, Deanne Adams, Haifeng Zhang, and James Hughes

Vanderbilt University
doug.clark@vanderbilt.edu, stephenkillingsworth@gmail.com, kwarizmi@gmail.com, grant.vaneaton@vanderbilt.edu, gautam.biswas@vanderbilt.edu, john.s.kinnebrew@vanderbilt.edu, pratim.sengupta@vanderbilt.edu, kara.krinks@gmail.com, deanne.adams@gmail.com, haifeng.zhang@vanderbilt.edu, jamesh53@gmail.com

Abstract. Digital games have the potential to make unique and powerful contributions to science education efforts. Much of that potential, however, remains unrealized, partly because powerful games for science learning need to synergistically augment commercial game design conventions and principles with design principles specific to the goals and nature of science learning and research on science learning. This paper builds on earlier frameworks outlining the affordances of commercial game design conventions for learning by proposing three design principles to help students explicitly articulate the intuitive science learning inherent in good game play in terms of formal science concepts and representations. We discuss these principles in the context of our recent and ongoing work in the SURGE projects. These projects investigate effective game mechanics to help students organize their tacit understandings about Newtonian mechanics into more formalized concepts.

Keywords: Digital learning environments, prediction, explanation, scaffolding, science education

1 Introduction

Digital games provide a promising medium for science education (Clark, Nelson, Sengupta, & D'Angelo, 2009; Honey & Hilton, 2010; NRC, 2009). In 2006, the Federation of American Scientists issued a widely publicized report stating their belief that games offer a powerful new tool to support education and encouraging private and governmental support for expanded research into complex gaming environments for learning. In 2009, a special issue of Science (Hines, Jasny, & Mervis, 2009) highlighted digital games in their survey of the promises and challenges of educational technology. Much of the initial debate over digital games for science education has focused on whether or not they support learning on science in general terms. This is obviously a simplistic question; well-designed games should produce better learning outcomes than games with unsound design. The NRC report on laboratory activities and simulations (Singer, Holton, & Schweingruber, 2005) supports this view, making clear that the design of physical and virtual learning activities, rather than simply the potential affordances of the medium, determines efficacy for learning. This paper outlines design principles focusing on helping students explicitly articulate the intuitive science learning inherent in good gameplay in terms of formal science concepts and representations.

2 SURGE I: Design and Rationale

SURGE was originally funded by an exploratory NSF DR-K12 grant between Vanderbilt University and Arizona State University (Clark & Nelson, 2008). The original design goal involved developing a game that would integrate formal physics representations and concepts with popular gameplay mechanics. We built SURGE I as a multi-platform game using the Unity3D game engine (unity3d.com). The SURGE I platform was intended to investigate design approaches for connecting students' "spontaneous concepts" (i.e., intuitions about kinematics and Newtonian mechanics) with formalized "instructed concepts." The design approaches integrate (1) disciplinary representations of Newtonian mechanics and explicit connections to its central concepts with (2) popular commercial game mechanics from games such as Mario Galaxy and Switchball that include marble motion. As a result, SURGE I and SURGE II are conceptually-integrated games for learning (Clark & Martinez-Garza, in press), rather than conceptually-embedded games. The science to be learned is thus integrated directly into the mechanics of navigating through the game world, rather than being embedded as an activity to be visited at some location in the game environment. The latter structure is typically present in many virtual worlds designed for science learning.

We focused heavily on popular game-play mechanics from appropriate game genres in the design of SURGE I. Core ideas from commercial game design conventions included (a) supporting engagement and approachable entry (Koster, 2004; Squire, 2011), (b) situating the player with a principled stance and perspective (McGonigal, 2011), (c) providing context and identification for the player with a role and narrative (Pelletier, 2008; Aarseth, 2007; Gee, 2007;), (d) monitoring and providing actionable feedback for the player (Annetta et al., 2009; Garris, Ahlers & Driskell, 2002; Kuo, 2007; Munz, Schumm, Wiesebrock & Allgower, 2007), and (e) using pacing and gatekeeping to guide the player through cycles of performance (Squire, 2006). An extended review of these commercial game ideas would be outside the focus of this paper; they are discussed in full detail in the cited works and other excellent analyses of the affordances of commercial game design for learning (e.g., Annetta, 2010; Gee, 2009; Klopfer, Osterweil, & Salen, 2009).

3 Baseline Student Performance in Original Surge I Design

Students playing versions of SURGE I demonstrated high engagement and significant learning gains on items based on the highly-regarded Force Concept Inventory (FCI), which is a widely known benchmark assessment for conceptual understanding of Newtonian dynamics at the undergraduate level (Hestenes & Halloun, 1995; Hestenes, Wells, & Swackhamer, 1992). A study with 208 seventh and eighth grade students in Taiwan and 72 seventh grade students in the United States (Clark, Nelson, Chang, D'Angelo, Slack, & Martinez-Garza, 2011), for example, showed significant pre-post gains, $t(250) = 2.0792$, p (one-tailed)$= 0.019$, with modest effect sizes. In Taiwan, 62% of the students liked or really liked playing SURGE, 32% thought it was okay, and only 6% did not like it. In the United States, 76% of the students liked or really liked playing SURGE, 21% thought it was okay, and only 3% did not like it. These percentages were similar across gender and previous game-playing experience. These findings mirrored our findings in multiple studies conducted with different populations including: (a) 155 U.S. undergraduate physics students (D'Angelo, 2010), (b) 69 U.S. Title I sixth grade students, (c) 72 U.S. undergraduate educational psychology students (Slack et al. 2010), and (d) 124 U.S. undergraduate educational psychology students (Slack 2011). Those studies showed similarly significant pre-post gains (one-tailed $p = .001$, $p = .02$, $p = .006$, and $p = .01$, respectively).

The downside, however, was that these gains and increasing mastery focused on intuitive understanding (which is what the FCI largely measures) rather than explicit understanding. Essentially, players could more accurately predict the results of various actions, impulses, and interactions (which improves performance in the game and on FCI questions), but players were not being supported in explicitly articulating their mental models and the connections from choices made in game play to formal disciplinary representations and concepts.

Thus these results demonstrated that the players were developing intuitive rather than formal understandings while playing a game built mainly on commercial design principles. This makes sense because the goal of commercial games involves helping players develop robust intuitive understanding that helps them enjoy increasing levels of mastery as they play the game, which naturally increases their engagement and desire to play more. If players are left confused and unable to learn to play the game, or if the learning process is overwhelming or poorly structured, players will disengage, making it very unlikely that they will recommend the game to others or purchase future versions of the game. Repeated designs of this type would naturally drive a game company into bankruptcy. Thus, strong evolutionary pressures in the gaming industry favor design conventions that support intuitive understanding. There is no immediate market need, however, for commercial games to support explicit articulation or connection to formal ideas. The intuitive understandings developed at the heart of commercial games generally are not intended to correspond with important understandings outside of those games.

The use and purposes of the knowledge obtained from gameplay in commercial digital games diverge in some important respects from the goals for science education. Commercial game design conventions thus need to be augmented to meet the educational goals for science education. For learners to achieve the goals of science education, they must be supported in explicitly integrating the intuitive understanding they develop through popular game-play mechanics with formal disciplinary concepts and representations. This is a critical challenge for the design of games for science learning. How do we promote the integration of intuitive and formal learning without sacrificing the engaging intuitive learning encouraged by successful commercial gameplay?

Research in psychology, science education, and the learning sciences suggests a number of ways to support explicit articulation and integration, but the design principles developed through that research focus on contexts and mediums with different characteristics, affordances, and constraints than those of digital games. As result, in order to be synergistic rather than disruptive, these design principles from psychology, science education, and the learning sciences require adaptation and reinterpretation for the digital game medium. Two areas of research are of specific interest in our own work for leveraging explicit articulation in synergy with commercial game design conventions. These areas of research focus on enhancing (1) prediction within navigation interfaces, (2) self-explanation within game dialog.

4 SURGE II Design Approach: Prediction within Navigation Interfaces to Scaffold Model Articulation

Our SURGE II research explores the potential of leveraging the research on prediction and explanation from psychology and science education to engage students in reflecting more consciously and deliberately about the underlying physics models (e.g., Mazur, 1996; Grant, Johnson & Sanders, 1990; Scott, Asoko & Driver, 1991). Prediction and explanation can promote metacognition, learning, and reflection (e.g., Champagne, Klopfer & Gunstone, 1982) and conceptual change (Tao & Gunstone, 1999; Kearney, 2004; Kearney & Treagust, 2000). A growing body of research and scholarship on games and cognition emphasizes cycles of prediction, explanation, and refinement at the core of game-play processes (Salen & Zimmerman, 2004, Wright, 2006).

In terms of scaffolding prediction, SURGE II shifts mechanics to adapt to what we have learned from SURGE I. In SURGE II, players navigate their avatar through the play area to collect Fuzzies and treasures and deliver them to safe locations while avoiding obstacles and enemies (as in SURGE I). Rather than employing the real-time interfaces of the original SURGE grant (where pressing an "arrow key" resulted in immediate application of an impulse or constant thrust in the direction of the arrow key), the new versions incentivize prediction by requiring the player to spatially place all of the commands in advance. This feature has the advantage of requiring the player to make predictions about the results of each command in terms of the motion of the player's avatar, rather than simply interacting reactively. Furthermore, SURGE II reduces the total number of commands a player initiates in a given level (thereby increasing the salience and impact of each individual command) to encourage players to think more carefully about the outcomes and implications of each action.

Our research with the new predictive interface to date has been promising. In our current study, 96 students played SURGE over three days. Learning outcomes were measured with an 11-item multiple-choice test of Newtonian kinematics modeled after the Force Concept Inventory and the Tennessee Comprehensive Assessment Program (TCAP) high-stakes science test. The pre- and post-test scores were compared using a two-sample paired t-test. The test showed a mean gain in test scores, from $M = 3.48$ to $M = 4.51$, and this result was statistically significant ($t = 5.184$, $p < .001$). The effect size was medium (Cohen's $d = 0.57$). Furthermore, the game was broadly appealing to students, with 92% of the respondents saying they "liked it" or "really liked it." Moreover, 80% of students considered the game appealing for both boys and girls. The sample comprised a cross-section of students who almost never play video games (40% reported playing less than two hours a week) as well as students for whom video games are a daily or near daily activity (33% reported playing an hour per day or more). These increased effect sizes encourage pushing forward with our exploration of leveraging prediction in the navigation interfaces.

5 SURGE II Design Approach: Self-Explanation within Game Dialog to Scaffold Model Articulation.

While the increased emphasis on prediction in the navigation design seems productive, the learning it promotes still focuses on making if/then predictions in the context of the consequences of different actions. We are, therefore, also exploring approaches for integrating explanation functionality into the dialog to leverage the increased intuitive grasp of the physics involved. Few games provide coherent structures for externalizing and reflecting on game-play; more often, such articulation and reflection occur outside the game, through discussion among players or participation in online forums (Gee, 2007; Squire, 2005; Steinkuehler & Duncan, 2008). We are now working to develop supports for this articulation and reflection by encouraging explanation and self-explanation in the dialog between the players and the characters within the game.

Research on self-explanation by Chi and others provides insight into the value of explanation for learning (e.g., Chi, Bassok, Lewis, Reimann, & Glaser 1989; Roy & Chi, 2005; Chi & VanLehn, in press). A recent review of research on students' self-explanation reports that self-explanation results in average learning gains of 22% for learning from text, 44% for learning from diagrams, and 20% for learning from multimedia presentations (Roy & Chi, 2005). Encouragingly, research by Bielaczyc et al. (1995) shows that instruction that stresses generating explanations improves performance even after the prompts that drive the explanations are discontinued. Mayer and Johnson (2010) have conducted preliminary work in embedding self-explanation in a game-like environment with encouraging results, including gains on transfer tasks. This emphasis on explanation is mirrored in research on science education. Work by White and Frederickson (1998, 2000), for example, demonstrates the value of asking students to reflect on their learning during inquiry with physics simulations.

Our design plan involves leveraging game dialog, which is a very popular aspect of conventional game design. Interestingly, while many aspects of commercial game design are currently very sophisticated, dialog in commercial games tends to involve relatively simple "multiple-choice" dialog trees that are not difficult to create. In fact, dialog in games is an area where educational games could take the lead. In SURGE II, after a player has completed a set of missions in the core game, a computer-controlled character in the game contacts the player and asks for help in mounting a similar rescue mission. The plan is for the resulting dialog tree to scaffold the player, requiring him or her to construct a solution for the character and to convince the character to try the solution by explaining how it fits a larger pattern of phenomena related to Newton's three laws of motion. Our goal is to present these invitations for dialog as puzzles that are engaging in their own right (Clark & Martinez-Garza, in press; Clark, Martinez-Garza, Biswas, Luecht, & Sengupta, in press). We will conduct our first studies of this approach later this year and will continue to explore its affordances for explicit articulation.

6 Bibliography

1. Aarseth, E. (2007). I Fought the Law: Transgressive Play and The Implied Player. Proceedings of DiGRA 2005 Conference: Situated Play.
2. Annetta, L. A. (2010). The “I's” have it: A framework for serious educational game design. Review of General Psychology, 14(2), 105.
3. Annetta, L. A., Minogue, J., Holmes, S. Y., & Cheng, M.-T. (2009). Investigating the impact of video games on high school students' engagement and learning about genetics. Computers and Education, 53(1), 74-85.
4. Bielaczyc, K., Pirolli, P., & Brown, A. L. (1995). Training in self-explanation and self-regulation strategies: Investigating the effects of knowledge acquisition activities on problem solving. Cognition and Instruction, 13(2), 221-252.
5. Champagne, A. B., Klopfer, L. E., & Gunstone, R. F. (1982). Cognitive research and the design of science instruction. Educational Psychologist, 17(1), 31.
6. Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. Cognitive Science, 13(2), 145–182.
7. Chi, M. T. H., & VanLehn, K. A. (in press). The content of physics self-explanations. Journal ofthe Learning Sciences.
8. Clark, D. B., & Nelson, B. (2008). Scaffolding Understanding by Redesigning Games for Education (SURGE). Exploratory DR-K12 grant funded by the U.S. National Science Foundation, 2008-2012
9. Clark, D. B., Nelson, B., Sengupta, P., D'Angelo, C. M. (2009). Rethinking Science Learning Through Digital Games and Simulations: Genres, Examples, and Evidence. Paper commissioned for the National Research Council Workshop on Games and Simulations. Washington, D.C.
10. Clark, D. B., Nelson, B. C., Chang, H.-Y., Martinez-Garza, M., Slack, K., & D'Angelo, C. M. (2011). Exploring Newtonian mechanics in a conceptually-integrated digital game: Comparison of learning and affective outcomes for students in Taiwan and the United States. Computers & Education, 57(3), 2178-2195. doi:16/j.compedu.2011.05.007

11. Clark, D. B., & Martinez-Garza, M. (in press). Prediction and explanation as design mechanics in conceptually-integrated digital games to help players articulate the tacit understandings they build through gameplay. C. Steinkuhler, K. Squire, & S. Barab (Eds.), Games, learning, and society: Learning and meaning in the digital age. Cambridge: Cambridge University Press.
12. Clark, D. B., Martinez-Garza, M., Biswas, G., Luecht, R. M., & Sengupta, P. (in press). Driving Assessment Of Students' Explanations in Game Dialog Using Computer-Adaptive Testing and Hidden Markov Modeling. In D. Ifenthaler, D. Eseryel, & G. Xun (Eds.). Game-based Learning: Foundations, Innovations, and Perspectives. New York: Springer.
13. D'Angelo, C.M. (2010). Scaffolding vector representations for student learning inside a physics game. Unpublished doctoral dissertation. Arizona State University.
14. Garris, R., Ahlers, R., & Driskell, J. E. (2002). Games, Motivation, and Learning: A Research and Practice Model. Simulation & Gaming, 33(4), 441 -467. doi:10.1177/1046878102238607
15. Gee, J. P. (2007). What Video Games Have to Teach Us About Learning and Literacy. Second Edition: Revised and Updated Edition (2nd ed.). Palgrave Macmillan.
16. Gee, J. P. (2009). Deep learning properties of good digital games: How far can they go. Serious Games: Mechanisms and Effects. Taylor & Francis Group, Routledge.
17. Grant, P., Johnson, L., Sanders, Y., & Science Teachers' Association of Victoria. (1990). Better links : teaching strategies in the science classroom. Melbourne: Science Teachers' Association of Victoria.
18. Hestenes, D., & Halloun, I. (1995). Interpreting the Force Concept Inventory: A Response to March 1995 Critique by Huffman and Heller. Physics Teacher, 33(8), 502–506.
19. Hestenes, D., Wells, M., & Swackhamer, G. (1992). Force Concept Inventory. Physics Teacher, 30(3), 141–158.
20. Hines, P. J., Jasny, B. R., & Merris, J. (2009). Adding a T to the three R's. Science, 323, 53.
21. Honey, M. A., & Hilton, M. (Eds.). (2010). Learning Science Through Computer Games and Simulations. National Research Council. Washington, DC: National Academy Press.
22. Kearney, M. (2004). Classroom Use of Multimedia-Supported Predict–Observe–Explain Tasks in a Social Constructivist Learning Environment. Research in Science Education, 34(4), 427–453.
23. Kearney, M., & Treagust, D. F. (2000). An investigation of the classroom use of prediction-observation-explanation computer tasks designed to elicit and promote discussion of students' conceptions of force and motion. Presented at the National Association for Research in Science Teaching, New Orleans, USA.
24. Klopfer, E., Osterweil, S., & Salen, K. (2009). Moving Learning Games Forward. Cambridge, MA: The Education Arcade.
25. Koster, R. (2004). A Theory of Fun for Game Design (1st ed.). Paraglyph Press
26. Kuo, M.-J. (2007). How does an online game based learning environment promote students' intrinsic motivation for learning natural science and how does it affect their learning outcomes? Digital Game and Intelligent Toy Enhanced Learning, 2007. DIGITEL '07. The First IEEE International Workshop on (pp. 135-142). Presented at the Digital Game and Intelligent Toy Enhanced Learning, 2007. DIGITEL '07. The First IEEE International Workshop on. doi:10.1109/DIGITEL.2007.28
27. Mayer, R. E., & Johnson, C. I. (2010). Adding instructional features that promote learning in a game-like environment. Journal of Educational Computing Research, 42(3), 241–265.
28. Mazur, E. (1996). Peer Instruction: A User's Manual (Pap/Dskt.). Benjamin Cummings.

29. McGonigal, J. (2011). Reality Is Broken: Why Games Make Us Better and How They Can Change the World. New York, NY: Penguin Press.
30. Munz, U., Schumm, P., Wiesebrock, A., & Allgower, F. (2007). Motivation and Learning Progress Through Educational Games. Industrial Electronics, IEEE Transactions on, 54(6), 3141-3144. doi:10.1109/TIE.2007.907030
31. National Research Council. (2009). National Research Council Workshop on Games and Simulations. October 6-7, 2009, Washington, D.C.
32. Pelletier, C. (2008). Gaming in Context: How Young People Construct Their Gendered Identities in Playing and Making Games. In Y. B. Kafai, C. Heeter, J. Denner, & J. Y. Sun (Eds.), Beyond Barbie and Mortal Kombat: new perspectives on gender and gaming. Cambridge, Mass: The MIT Press.
33. Roy, M., & Chi, M. T. H. (2005). The self-explanation principle in multimedia learning. In R. E. Mayer (Ed.), The Cambridge handbook of multimedia learning (pp. 271-286). New York: Cambridge University Press.
34. Salen, K., & Zimmerman, E. (2004). Rules of play: game design fundamentals. 2004.
35. Scott, P. H., Asoko, H. M., & Driver, R. H. (1991). Teaching for conceptual change: A review of strategies. Connecting Research in Physics Education with Teacher Education, 71–78.
36. Singer, S. Hilton, M.L., & Schweingruber, H.A. (Eds.) (2005). America's lab report: investigations in high school science. Washington, DC: National Academies Press.
37. Slack, K., Nelson, B., Clark, D. B., & Martinez-Garza, M. (2011). Model-Based Thinking in the Scaffolding Understanding by Redesigning Games for Education (SURGE) Project. Poster presented as part of a structured poster session at the American Educational Research Association (AERA) 2011 meeting. New Orleans, LA.
38. Slack, K. Nelson, B., Clark, D. B., Martinez-Garza, M. (2010). Influence of visual cues on learning and in-game performance in an educational physics game environment. Paper presented at the Association for Educational Communications and Technology (AECT) 2010 meeting. Anaheim, California.
39. Squire, K. (2005). Changing the game: What happens when video games enter the classroom. Innovate: journal of online education, 1(6), 25–49.
40. Squire, K. (2006). From content to context: Videogames as designed experience. Educational Researcher, 35(8), 19.
41. Squire, K. (2011). Video Games and Learning: Teaching and Participating Culture in the Digital Age. New York: Teachers College Press.
42. Squire, K. D., DeVane, B., & Durga, S. (2008). Designing centers of expertise for academic learning through video games. Theory Into Practice, 47(3), 240–251.
43. Steinkuehler, D., & Duncan, S. (2008). Scientific habits of mind in virtual worlds. Journal of Science Education and Technology, 17(6), 530-543.
44. Tao, P.-K., & Gunstone, R. F. (1999). The process of conceptual change in force and motion during computer-supported physics instruction. Journal of Research in Science Teaching, 36(7), 859–882.
45. White, B., & Frederiksen, J. (1998). Inquiry, modeling, and metacognition: Making science accessible to all students. Cognition and Instruction, 16(1), 3-118.
46. White, B. Y., & Frederiksen, J. R. (2000). Metacognitive facilitation: An approach to making scientific inquiry accessible to all. In J. A. Minstrell & E. H. Van Zee (Eds.), Inquiring into Inquiry Learning and Teaching in Science (pp. 331-370). American Association for the Advancement of Science.
47. Wright, W. (2006). Dream machines. Wired 14(04).

Understanding Users' Interaction Behavior with an Intelligent Educational Game: Prime Climb

Alireza Davoodi[1], Samad Kardan[1], Cristina Conati[1]
[1] Computer Science Department, The University of British Columbia,
ICICS/CS Building, 201-2366 Main Mall, Vancouver, B.C., Canada V6T 1Z4,
{davoodi, skardan, conati}@cs.ubc.ca

Abstract. This paper presents work on applying clustering and association rule mining techniques to mine users' behavior in interacting with an intelligent educational game, Prime Climb. Through such behavior discovery, frequent patterns of interaction which characterize different groups of students with similar interaction styles are identified. The relation between the extracted patterns and the average domain knowledge of students in each group is investigated. The results show that the students with significantly higher prior knowledge about the domain behave differently from those with lower prior knowledge as they play the game and that pattern could be identified early during the interactions.

Keywords: Intelligent Educational Games, Behavior Discovery, Association Rule Mining, Open Ended Learning, Scaffolding

1 Introduction

Open-Ended Learning Environments (OELEs) support student-centered learning and allow learners to follow an exploratory interaction behavior to construct their own models of concepts and revise their beliefs subsequent to receiving immediate feedback on their actions [1]. Previous studies have shown that students could not benefit much from an open-ended learning environment if not receiving proper feedback [2]. Among learning environments, educational games are designed to foster motivation and engagement which are shown to be influential in learning [3]. To this end, educational games such as Crystal Island provide exploratory learning environments and encourage autonomous interaction with the game [4]. While such freedom in interaction is required to maintain engagement in the game, it also provides learners with the possibility of showing different interaction patterns. The interaction patterns might be indicative of certain characteristics and understanding such patterns can provide valuable information about the students.

Adaptive OELEs have been designed to answer the need for understanding and intelligently supporting varying learning styles, capabilities, and preferences in individuals in developing their skills. An adaptive educational system maintains a model of student's learning and leverages the student's interactions with the system to provide tailored scaffolding. Many educational systems apply data mining approaches on the logs of students' recorded interactions to extract behavioral patterns and extract high-level information about students [5-7]. Along this line of research, we concentrate on understanding how students interact with Prime Climb (PC), an adaptive educational game (edu-game) and whether there is a connection between behavior patterns of students and their attributes such as prior knowledge. The ultimate goal in an adaptive educational game such as PC is to help a higher number of students learn the desired skills through interacting with the game. Achieving such an objective requires a pedagogical agent which maintains an accurate understanding of individual differences among users and provides more tailored interventions, with the aim of guiding the learners in the right learning direction. For instance, if a pedagogical agent is capable of identifying a group of students with high domain knowledge, it is possible to leverage such information to construct a more accurate user model and intervention mechanism. The user's interaction behaviours can also be provided to developers to improve the design of educational systems [8].

Behavioral discovery has been vastly used in educational systems, but there is limited application in educational games such as Prime Climb, in which educational concepts are embedded and presented in the game scenarios and narratives with minimum explicit technical notation (for instance mathematical notations in PC) to more genuinely support game aspects of the system. In Prime Climb, students do not explicitly practice approaches to number factorization but implicitly follow a self-regulated learning approach [9] to explore and understand the methods and practice them. This paper describes the first step toward leveraging students' behavioral patterns into building a more effective adaptive edu-game. The ultimate goal is devising mechanisms for extracting abstract high-level patterns from raw interaction data and leveraging such understanding for real-time identification of interaction styles to enhance user modeling and intervention mechanism in an edu-game like PC.

Behavior discovery has been recently applied in different educational systems. Kardan et al. [6] leveraged behavior discovery to propose a general framework for distinguishing users' interaction styles in exploratory learning environments. Keshtkar et al. [10] describe an approach to distinguishing players and mentors roles in a multi-chat environment within the epistemic game Urban Science. In another related work, Mccuaig et al. [5] discuss using interaction behaviors to distinguish students who will fail or pass a course in a Learning Management System (LMS). A sequence mining approach has been also used in differentiating behavior patterns in students' interacting with Betty's Brain, a learning-by-teaching environment [7].

Although behavior discovery has been recently applied to many educational systems, there is very limited work on behavior mining in an open ended intelligent educational game like Prime Climb in which learning through playing the game is intended. Additionally, most of the previous works use the entire interaction data to make inferences about the users. In this work, we present the results of behavior mining not only on a big portion of interaction data but also on a truncated data set, which will provide the possibility of constructing an online classifier for early detection of varying patterns of interactions.

2 Prime Climb an Intelligent Edu-game

Prime Climb (PC) is an intelligent educational game for students in grades 5 and 6 to practice number factorization skills. Prime Climb is equipped with an intelligent pedagogical agent which maintains a probabilistic model of the student's knowledge on number factorization skills.

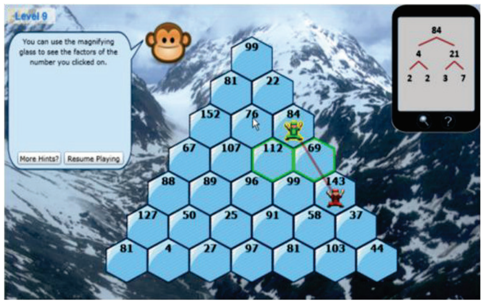


Fig. 1 Prime Climb

The pedagogical agent leverages the probabilistic model to provide an adaptive scaffolding mechanism. If model's assessment about the student's knowledge on a skill falls below a certain threshold, a hint is presented to the player. The hints are given in incremental level of details. In PC, the player and his/her partner climb a series of 11 mountains of numbers by pairing up the numbers which do not share a common factor. There are two main interactions of a player with PC:

Making Movements**:** A player makes one or more movements at each time, by clicking on numbered hexagons on the mountains. PC provides immediate feedbacks on correctness of movements. If a player makes a wrong movement, s/he falls down.

Using Magnifying Glass Tool**:** The magnifying glass (MG) tool is always available for the user to benefit from. The MG is used to show the factor tree of a number on the mountains; it is located in the top right corner of the game (Fig. 1).

4 Behavior Discovery in Prime Climb

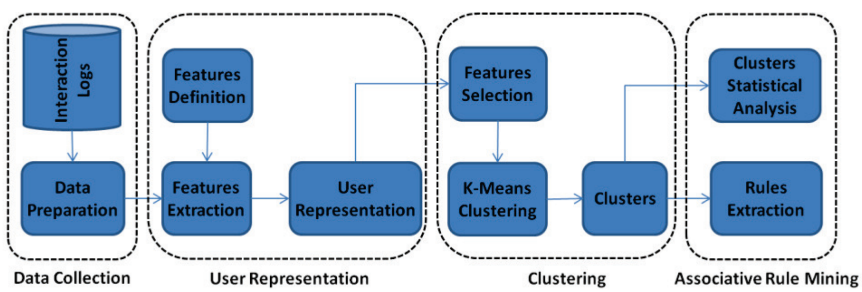


Fig. 2: Behavior discovery methodology in Prime Climb

4.1 Data Collection

Data collection is first component of the behavior discovery methodology in PC shown in Fig. 2. We collected interaction logs of 45 students who played PC voluntarily. Prime Climb consists of 11 levels (mountains), and not all students could manage to reach the last level. Out of the 45 students, 43 completed 9 or more levels. The remaining 2 students who completed fewer levels were excluded from further analyses to ensure that all students in analysis had completed a minimum of 9 levels. For the remaining 43 students, the interaction data for the first 9 mountains was used in the feature extraction process.

4.2 User Representation

Features Definition: Each user is represented by a vector of features. Based on the 2 main groups of interaction previously mentioned (movements and MG), two types of features are defined: (1) Movements features based on statistical measures on movements students made on the mountains and (2) MG features: based on statistical measures on students' usages of the MG tool. Table 1 shows some of these features:

Table 1 Some features used for behavior discovery

Movement Features
[Sum/Mean/STD] of number of [correct/wrong] movements made by a student across mountains
[Sum/Mean/STD] of time on [correct/wrong] movements made by a student across mountains
[Mean/STD] of length of sequences of [correct/wrong] moves made by a student
[Mean/STD] of time spent per sequence of [correct/wrong] moves made by a student
Magnifying Glass (MG) Features
[Sum/Mean/STD] of MG usage
Mean number of [correct/wrong] movements per each MG usage
STD of number of [correct/wrong] movements per each MG usage

Feature Set Definition: Each feature is a measure computed based on user's interactions with one or more mountains. There are two types of feature:
Mountain-Generic Features (m – n), (m >= 1 and n <= 9)**:** Calculated based on the users' interactions with mountains *m* to *n*, inclusively. For instance, the feature, correct-movements (1–9), represents the total number of correct movements made by the user on mountains 1 to 9.
Mountain-Specific Features (k), (1 <= k <= 9)**:** Calculated based on interactions with mountain *k*. For instance, correct-movements (7), represents the total number of correct movements made by the user on mountain 7.
In this paper, we present the behavior discovery results on the two feature sets:
Mountain-Generic Movement(1–9) Set**:** Contains mountain-generic features (1–9) which are related to movement actions the student makes.
Mountains-Generic+Specific-MG+Movement(1–2) Set**:** Contains mountain-generic MG features (1–2), mountain-generic movement features (1–2), mountain-specific MG features (1) and (2), and mountain-specific movements features(1) and (2).

4.3 Clustering

Feature Selection: Prior to performing clustering, feature selection is applied to filter out irrelevant features [11].
Clustering: The optimal number of clusters is determined as the lowest number suggested by C-index, Calinski and Harabasz[12] and Silhouette [13] measures of clustering validity. Once all the students are represented by vectors of selected features, the GA *K*-means (*K*-means for short) clustering algorithm [6], which is a modified version of GA *K*-means [14], is applied to cluster the users into an optimal number of clusters.

4.4 Rule Mining: Higher Prior Knowledge *vs.* Lower Prior Knowledge

Next, the Hotspot algorithm [15] is used to extract the rules for each discovered cluster. Also, we analyzed whether the resulting clusters are significantly different on a measure called *cluster's prior knowledge*, which is defined as follows:
Cluster's Prior knowledge: The cluster's prior knowledge gives the average level of factorization skills of the cluster's members prior to playing the game and is defined as the average of raw pre-test scores of the cluster's members. The following formula is used to calculate the cluster's prior knowledge:

$$\textbf{Cluster's prior knowledge} = \frac{\sum_{\textbf{student}\in\textbf{cluster}} \textbf{pre\_test}(\textbf{student})}{\textbf{Cluster's size}}. \quad (1)$$

where $\text{pre\_test}(\text{student})$ is the student's pre-test score. Before playing the game, a student takes a pre-test on number factorization skills. The maximum score a student can get is 15. The average pre-test score across the 43 students is 11.7, and the standard deviation is 3.29.
Behavior Discovery on Mountain-Generic-Movement(1–9) set: In this feature set, each student is represented by a vector of mountain-generic movement features(1–9). As a result of the features' selection mechanism, 18 features were selected out of the original 30 features. The optimal number of clusters was found to be 2, and the *K*-means method was used to cluster the set of students into 2 groups. The result of a *t*-test showed that there is a statistically significant difference between the prior knowledge of cluster 1 of students (higher prior knowledge (HPK) group) (M = 13.0 , SD = 2.0) and cluster 2 of students (lower prior knowledge (LPK) group) (M = 11.3 , SD = 3.45), p=.03 and Cohen's d= 0.53. Next, the Hotspot association rule mining algorithm was applied on the clusters to extract the associative rules. Table 2 shows the rules extracted for each cluster.
Understanding the Rule Mining Results:
Rules: Each bulleted item in following tables shows an extracted rule. For example, "Mean-Time-on-Movements=Higher" is an extracted rule which applies to at least 25% of the members of cluster 1. (In this study, the threshold of 25% is applied for all rules extracted by the Hotspot algorithm). This rule shows that the values for the feature "Mean-Time-On-Movements(1–9)" across the cluster's members belong to the "Higher" Bin.
Bins: In this study, the Hotspot algorithm considers two bins for values of each feature: (1) Lower bin and (2) Higher bin. Each bin shows a range of values of the features such that the lower bin represents the lower range of values and the higher bin represents the upper range of values for the feature. The cut-off point for splitting a range of values for a feature into two ranges (lower and upper) is calculated specifically for the feature in each extracted rule by the Hotspot algorithm. The lower and higher bins are indicated by the words "Lower" and "Higher" in front of the features in the following tables.
Rule's Support: The other important information is the rule's support shown in square brackets in front of the extracted rules in the following tables. For instance, [6/6=100%] in front of the first rule for the cluster 1 in Table 2 shows that there are in total 6 (in denominator) out of 43 students on which the extracted rule applies and all of these students belong to cluster 1 (6 in the numerator of the fraction). In addition, it can be concluded that this extracted rule applies to 60% (6/10) of the cluster 1 (note that the size of cluster 1 is 10).

Table 2: Extracted Rules for Mountains-Generic-Movement(1–9)

Rules for Cluster 1[HPK]: (Size: 10/43 = 23.26%)
• Mean-Time-on-Movements(1-9) = Higher, [6/6=100%] • Mean-Time-Spent-On-Correct-Movements-On-Mountains(1-9) = Higher, ([5/5=100%])
Rules for Cluster 2[LPK]: (Size: 33/43 = 76.74%)
• Mean-Time-On-Movements(1-9) = Lower, [33/37=89.19%] ○ STD-Time-On-Wrong-Correct-Moves(1-9) = Lower, [33/35=94.29%] • Mean-Time-On-Consecutive-Wrong-Movements(1-9) = Lower, [31/35=88.57%] ○ STD-Time-On-Movements(1-9) = Lower, [31/33=93.94%] ○ STD-Time-On-Correct-Movements(1-9) = Lower, [31/33=93.94]

Discussion and Interpretation: The extracted rules show that the students belonging to the HPK cluster (cluster 1) spent more time on movements and correct movements across 9 mountains. This could indicate that the students with higher prior knowledge were more involved in the game and spent more time before making a movement. Since the time spent on making a correct movement is higher for this group of students, it might mean that a correct move by this group of students is less likely to be due to a lucky guess as compared with the total population. In contrast, the group of students with lower prior knowledge spent less time on making movements as well as making wrong movements. This could be an indication of less involvement in the game by the lower prior knowledge group. It could show that a correct movement by this group of students is more likely due to a guess. In addition, there are some other frequent patterns of interaction for the group of students with lower prior knowledge. These patterns show a lower standard deviation on time spent on making movements and correct movements. This indicates that this group of students showed a consistent pattern of lack of engagement in the game. Therefore, we can conclude that the students with higher prior knowledge showed more engagement in the game than students with lower prior knowledge.

Behavior Discovery on Mountain-Generic+Specific-MG+Movement(1–2) set: This feature set does not employ interaction data from all 9 mountains; instead, only the data from the first 2 mountains is included. Such feature set is mainly valuable for constructing an online classifier to classify students based on their interaction with the game during the game play. The ultimate aim is leveraging such a feature set to step toward building a more accurate individualized student model and intervention mechanism as the student makes progress in the game. For instance, if the classifier can identify a student as a lower/higher knowledgeable student, it could leverage the information for early adjustment of the adaptive intervention mechanism. Similarly *K*-means was applied to cluster the students represented by the Mountains-Generic+Specific-MG+ Movements (1–2) Set. The optimal number of clusters was calculated to be 2, and 25 features out of 51 original ones were selected, as a result of applying the features selection mechanism. The cluster's *prior knowledge* was calculated for each of the discovered clusters and compared using a *t*-test. The result of the *t*-test showed a statistically significant difference between cluster *1's prior knowledge* (M = 12.45 , SD = 2.66) and cluster 2's prior knowledge (M = 9.22 , SD = 3.93), p = 0.02, Cohen's d = 1.08. Next, association rule mining was applied on the 2 clusters, as shown in Table 3.

***Interpretation and Discussion*:** As shown in Table 3, the results of behavioral discovery on the Mountains-Generic+Specific-MG+Movements(1-2) set is not consistent with the results of behavior discovery on the Mountains-Generic-Movements(1-9) set. Behavior discovery on interaction data from the first two mountains shows that students with higher prior knowledge (M = 12.45 , SD = 2.66) constitute around 79% of the all students and spend less time on making movements. It was previously shown in Table 2 that the students in the HPK cluster constituted approximately 23% of all students and spent more time on making movements when interaction data from all 9 mountains was included. Despite this, we expect that as the students progress in the game, the students with higher prior knowledge would behave differently from the other students and separate themselves from the others. To verify this, we also extracted frequent patterns when more interaction data from upper mountains is included in the clustering and rule mining. When the interaction data from the first 3 mountains is included in patterns mining, 2 clusters are identified which are not significantly different on their prior knowledge. When interaction data from the first four mountains is included, we observe patterns similar to those identified using the interaction data from all 9 mountains as shown in Table 3-right. The result of the t-test shows a statistically significant difference between cluster 1's prior knowledge (M = 13.28, SD = 1.58) and cluster 2's prior knowledge (M = 11.39 , SD = 3.4), p = 0.02, Cohen's d = 0.60. Also, approximately 16% of students belong to the HPK cluster, and 84% belong to the LPK group. This result is very similar to the results when data from all 9 mountains is included. Similar patterns are observed when more interaction data from upper mountains is included in the analysis.

Table 3: Extracted Rules for Mountains-Generic+Specific-MG+ Movements(1-2) [left] and MG+Movements(1-4) [right]

Rules for Cluster 1[HPK] (Size: 33/42=78.57%)
• Mean-Time-On-Movements(1)=Lower, [30/31 =96.77%] • Mean-Time-On-Movements(1-2) = Lower, [29/30 = 96.67%]
Rules for Cluster 2[LPK] (Size: 9/42=21.43%)
• Mean-Time-Spent-On-Mountain(1-2) = Higher, [7/7=100%] • Total-Time-On-Mountain(1) = Higher, [5/5=100%]

Rules for Cluster 1[HPK] (Size: 7/43=16.28%)
• Mean-Time-On-Movements(4) = Higher, [5/5 = 100%] • Mean-Time-On-Correct-Movements(3) = Higher, [3/3 = 100%]
Rules for Cluster 2[LPK] (Size: 36/43=83.72%)
• Mean-Time-On-Correct-Movements(1-4) = Lower, [35/35 = 100%] • Mean-Time-On-Movements(1-4)=Lower, [34/34 = 100%]

5 Conclusions and Next Steps

This paper discusses behavior discovery in Prime Climb (PC). To this end, different sets of features were defined. The features were extracted from interaction of students with PC in the form of making movements from one numbered hexagon to another numbered hexagon and usages of the MG tool. To identify frequent patterns of interaction, first, a feature selection mechanism was applied to select more relevant features from the set of all features. Then a K-means clustering was applied to cluster the students into an optimal number of clusters and the Hotspot algorithm of association rule mining was applied on the clusters to extract frequent interaction patterns. Finally, the prior knowledge of the clusters were compared. When interaction data from all 9 mountains was included in behavior discovery, it was found that the students with higher prior knowledge were more engaged in the game and spent more time on making movements. In contrast, the students with lower prior knowledge spent less time on making movements, indicating that they were less involved in the game. Behavior discovery also was conducted on truncated sets of features in which only a fraction of interaction data was included. The results showed that using the interaction data from the first two mountains resulted in groups of students that are statistically different on their prior knowledge.

The scaffolding mechanism in PC relies on the student model so we expect improvements in the model to result in more tailored interventions and guidance. Current PC uses the same student model for all students. Following the results of the presented study, we plan to adjust the model based on the characteristics of each discovered group of students. In addition, an online classifier will be built which identifies frequent patterns of interaction in the students, classifies them into different groups in real time, and leverages such information to build a more personalized user model and adaptive intervention mechanism in PC.

References

1. Papert, S. Mindstorms (2nd ed.), New York: Basic Books inc. (1993) _
2. L. Alfieri, P. Brooks, N. Aldrich, and H. Tenenbaum, "Does Discovery-Based Instruction Enhance Learning," Journal of Education Psychology, vol. 103, no. 1, pp. 1-18, (2011)
3. Lee, J., Luchini, K., Michael, B., Norris, C., Solloway, E., 2004, More than just fun and games: Assessing the value of educational video games in the classroom. Proceedings of ACM SIGCHI 2004, Vienna, Austria, pp. 1375–1378 (2004)
4. Sabourin, J., Rowe, J., Mott, B., & Lester, J. Exploring Affect and Inquiry in Open-Ended Game-based Learning Environments. (2011)
5. Mccuaig J., Baldwin, J.: Identifying Successful Learners from Interaction Behaviour. EDM 2012, Chania, Greece. pp. 160–163 (2012).
6. Kardan, S., and Conati. C.: A Framework for Capturing Distinguishing User Interaction Behaviours in Novel Interfaces. EDM, Eindhoven, the Netherlands, pp. 159–168 (2011)
7. Kinnebrew, J. S., Biswas G.: Identifying Learning Behaviors by Contextualizing Differential Sequence Mining with Action Features and Performance Evolution, EDM 2012, Chania, Greece. pp. 57–64 (2012).
8. Hunt, E., Madhyastha, T.: 2005: Data Mining Patterns of Thought. In: Proceedings of the AAAI Workshop on Educational Data Mining. (2005).
9. Paris, S., Paris, A.: Classroom Applications of Research on Self-Regulated Learning. Educ. Psychol. 36(2), 89–101 (2001)
10. Keshtkar, F., Morgan, B., Graesser, A.: Automated Detection of Mentors and Players in an Educational Game. EDM 2012, Chania, Greece. pp. 212–213 (2012)
11. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. 3, 1157–1182 (2003).
12. Milligan, G.W., Cooper, M.C.: An Examination of Procedures for Determining the Number of Clusters in a Data Set. Psychometrika, 50(2), 159–179 (1985).
13. Rousseeuw, P. J. :Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis". *Computational and Applied Mathematics* 20: 53–65. doi:10.1016/0377-0427(87)90125-7, (1987)
14. Kim, K. Ahn, H.: A Recommender System using GA *K*-means Clustering in an Online Shopping Market. Expert Syst. Appl. 34(2), 1200–1209 (2008).
15. Hall, M., Eibi, F., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data mining Software: An Update. SIGKDD Explor. 11(1), 10–18 (2009)

Designing Digital Objects to Scaffold Learning

Grant Van Eaton, Douglas B. Clark, David Beutel

Vanderbilt University, Peabody College
grant.vaneaton@vanderbilt.edu, doug.clark@vanderbilt.edu,
dmbeutel@gmail.com

Abstract. Digital objects in learning games provide opportunities to scaffold teacher and student learning toward deeper epistemological understanding of the concepts they represent. Representations encapsulated in digital objects, however, have the potential to misrepresent the concepts they stand in place of. Using student and teacher interview data after playing a physics learning game, analysis of the role of representations in students' epistemological development led to two design recommendations. When designing digital objects to effectively scaffold concepts, designers should pay attention to the ways in which learning environments explore the nature of core concepts represented by digital objects and explicitly model the meaning of the representations in the learning environment.

Keywords: Digital learning environments, representation, scaffolding, epistemology, science education

1 Introduction

In their review of the literature on digital games and simulations for science education, Clark, et al [1] propose a shift in research agenda away from an exploratory phase that furnishes mere proofs of concept and instead calls on researchers to focus on ascertaining the design principles that best support learning and conceptual change. Design principles in digital learning environments necessarily rely on the use of representations that interact with players in order to model core concepts. These representations then have the power to scaffold the learning trajectory of both teachers and students as they play the game. Representations, however, have the ability to take on a life of their own as a teacher or student appropriates them as tools for learning. Using interview data collected from a four-day classroom implementation of the SURGE: EPIGAME physics learning game, this paper will explore two questions central to the interplay between design, representation, and epistemology:

- How do representations in the SURGE learning environment interact with teachers and students?
- How do these representations scaffold the development of teachers' and students' epistemology of force?

1.1 Theoretical Framework

When thinking about how to use representations to scaffold concepts in a digital learning environment, Ball and Cohen's [2] educative curriculum framework provides an orientation that positions the learning environment to scaffold learning not only for students, but also their teachers. Using learning games to develop deeper content knowledge in teachers, however, will only be effective insofar as 1) the representations in the learning environment properly embody the focus concept(s) and 2) if the correct scaffolds are in place to bridge teachers' intuitive understanding of their content with the concepts represented in the game.

1.2 Representations in the Learning Environment

In order to discuss the potential for learning games to educate students, and the importance of representations to accomplish this task, this analysis will focus on a key representation in the SURGE: EPIGAME learning environment: force. In SURGE, players must navigate a spaceship around obstacles while staying on a set path. This is accomplished by issuing commands to the ship as to the magnitude and direction the ship should fire forces to achieve the desired path. Within the game, these representations are represented by force tiles placed on a timeline delineated in one-second increments.

As representations in the game, force tiles are intended to represent a command given to the ship to fire a force of a specific magnitude and direction at a certain time. This representation is not the actual force being applied, but rather a command to the ship to fire the desired force. Force tiles are placed within the timeline at the bottom of the simulation space, representing when the ship should issue the command to fire the force indicated on the force tile. The timeline is thus intended to represent and visualize the amount of time between commands to fire forces.

2 Impact of Representations on Scaffolding Learning

Lehrer and Schauble [3] have shown that representations edit concepts insofar as they reduce or enhance the information they contain. In the best case scenario, these reductions and enhancements effectively scaffold student and teacher understanding toward the concept embodied in the representation. These representations, however, also have the potential to misrepresent the concept to such an extent that, despite the best design intentions, students and teachers emerge from interaction with the representation holding a fundamentally different concept than intended by designer.

2.1 Force

Throughout student interviews, force tiles take on independent ontological status as actors in the game's simulation space, contrary to the intent of the designers. One student repeatedly talks of 'sending' a force from the timeline into the simulation space in order to do work, even gesturing from force tiles in the timeline to the point in the simulation space in which the force is applied:

Student: Like, where it sends... where you send a 60 Newton force over here to get to this point, and then you'd send another 60 Newton force to stop it *[student gestures from 60 Newton force on timeline to the spot where the force is applied in the simulation space]* ... and then a 20 Newton force... *[repeats gesture]* and then a 20 Newton force to stop it and go up... *[repeats gesture]*

In the student's explanation, he student sends a 20 Newton force "to stop" the ship. In the student's mind, the force tile does not represent a mere command for the ship to apply force and decelerate, but rather the force tile object itself travels into the simulation space to oppose the movement of the ship.

This distinction is important with regard to the student's developing epistemology of force. Within the framework of the force tile merely representing a command of the ship to apply force, the action of the ship carrying out the force tile's command represents a change in velocity to decelerate the ship, Newton's second law of motion. The student's conception of the force tile being 'sent' into the simulation space to 'oppose' the ship, however, gives agency to the force tile to travel into the simulation space and push backward on the ship in order to stop it, an enactment Newton's third law of motion. This unintended consequence is directly related to the design of the force representation.

The student's teacher, perhaps unsurprisingly, also echoes his student's epistemological misconception. Following gameplay, the student's teacher was given an example level from the game and asked to identify each of Newton's laws in the level:

Teacher: Newton's second... of course, when I change from at rest to in motion I've applied a force. So [the ship] starts moving from left to right. When I stopped [the ship] here I had to put an unbalanced force on it to go up to down.

Teacher: Newton's third law... opposites. When I stopped the ship I had to apply an opposite force of the same force amount to make my ship stop.

In these two statements, the teacher's epistemology of force becomes evident: unbalanced forces (Newton's second law) start the motion of the ship and opposing forces (Newton's third law) stop the ship. Parsing the teacher's response, the verb 'to apply' takes center stage. In his second law formulation, the teacher "applied a force" and in his third law formulation, the teacher also "had to apply an opposite force" in order to achieve the outcome he desired in the simulation space. Within the semantic frame of application, force is no longer applied by the ship, but by the teacher. What and where this force is, however, remains elusive. It is conceivable, based on the formulation of Newton's third law to 'stop the ship', that the ability to apply force in the simulation environment is a property of the force tile, which pushes on the ship to cause it to stop. As the teacher seeks to answer the question 'What is force?', the representations of the learning game lead to the conclusion that force is a property of an acting object opposing another acting object, scaffolded by the representation of the force tile opposing the ship.

3 Redesign Suggestions for Scaffolding Learning

As a result of the effects of representations on scaffolding epistemological formation evidenced in the student interview, two considerations for future design of scaffolding in digital learning environments emerge.

3.1 Exploration of Core Concepts

Confusion emerges on the part of the student as to the nature of force. Integrating opportunities within the game to explore the question "what is force?" could potentially clarify for students what the force tiles represent, allow for the representation to better scaffold understanding of force and motion, and further reinforce canonical understanding of Newton's laws. In the absence of such an exploration, students are free to ascribe their own properties to the objects, 'sending' them to do work that they are actually incapable of doing.

3.2 Explicit Modeling of Representations

Beyond exploration, however, teachers and students must have the nature of representations in gaming environments explicitly modeled to ensure properties of the object are correctly ascribed. In the SURGE example, a simple statement that the force tiles are not, in fact, independent objects that travel to the simulation space and push on the ship, but rather are simply commands given to the ship to fire its rockets, could potentially alleviate the confusion as to the tile's agency in the simulation space.

References

1. Clark, D., Nelson, B., Sengupta, P., & D'Angelo, C. (2009). Rethinking science learning through digital games and simulations: Genres, examples, and evidence. In Learning science: Computer games, simulations, and education workshop sponsored by the National Academy of Sciences, Washington, DC.
2. Ball, D. L., & Cohen, D. K. (1996). Reform by the book: What is: Or might be: The role of curriculum materials in teacher learning and instructional reform? *Educational Researcher*, *25*(9), 6-14.
3. Lehrer, R., & Schauble, L. (2002). Symbolic communication in mathematics and science: Co-constituting inscription and thought. In E. Amsel & J. Byrnes (Eds.), The development of symbolic communication (pp. 167-192). Mahwah, NJ: Erlbaum.

Fostering Diagnostic Accuracy in a Medical Intelligent Tutoring System

Reza Feyzi-Behnagh, Roger Azevedo, Elizabeth Legowski, Kayse Reitmeyer, Eugene Tseytlin, and Rebecca Crowley

Department of Educational and Counselling Psychology, McGill University, Montreal, Canada
Department of Biomedical Informatics and Pathology, University of Pittsburgh, PA, USA
reza.feyzibehnagh@mail.mcgill.ca; roger.azevedo@mcgill.ca;
{legoex, reitmeyerkl, tseytline, crowleyrs}@upmc.edu

Abstract. Diagnostic classification is an important part of clinical care, which is often the main determinant of treatment and prognosis. Clinicians' under- or over-confidence in their performance on diagnostic tasks can result in diagnostic errors which can lead to delay in appropriate treatment and unnecessary increase in the cost of medical care. This paper presents a version of SlideTutor aiming to reduce pathologists' and dermatopathologists' bias in diagnostic decision-making. This is accomplished by frequently prompting them to make metacognitive judgments of confidence, presenting them with the expert diagnostic solution path for each case, and de-biasing them by making them conscious of their metacognitive biases. This paper describes and summarizes the functionalities of SlideTutor, its cognitive training, tutoring phase, expert feedback, metacognitive intervention, and the open learner model.

1 Introduction and Background

Intelligent tutoring systems (ITSs) are adaptive and personalized instructional systems designed to mimic the well-known advantages of human one-on-one tutoring over other types of instructional methods [e.g., 1]. ITSs are capable of accelerating and enhancing the training of novices by providing adaptive and individualized scaffolding and feedback based on a complex interaction between several modules representing the domain knowledge as well as learner knowledge acquisition and development of expertise. The adaptive scaffolding and feedback in ITSs are targeted at improving student learning and fostering skills, such as making accurate metacognitive judgments [see 2]. In contexts where the teacher has limited time to spend on presenting content, teaching problem solving skills, and providing tailored feedback to individual students, ITSs can prove extremely helpful by providing adaptive individualized instruction to learners, organize content, and point out their errors for as much time and as many iterations as the learner requires [3].

ITSs can prove beneficial in training of highly specialized clinicians, such as pathologists. Training of specialized clinicians is very difficult in traditional training contexts for several reasons, including insufficient exposure to infrequently encountered cases, and the increased workloads of mentors which limit the time for training the next generation of practitioners and increase the potential for clinical errors among less-experienced practitioners. Training of pathologists typically requires five or more years, which includes both residency training (3-5 years) and advanced fellowship (1-3 years). In the context of training pathologists, ITSs could help alleviate many of the above-mentioned problems by providing a safe environment where residents can practice whenever they have time and as frequently as needed, and receive individualized feedback and guidance without inadvertently harming patients in the process. More specifically, ITSs can scaffold residents' accuracy of diagnoses, thereby alleviating their overconfidence or under-confidence in their performance on diagnostic tasks. Overconfidence would cause the clinician to conclude the diagnosis too quickly, therefore neglecting to fully consider alternative hypotheses and all the evidence in the case, which can result in diagnostic errors [4]. On the other hand, under-confidence might lead them to order unnecessary or inappropriate additional testing and use consultative services, which increases the risk of iatrogonic complications (i.e., complications caused by medical treatment or diagnostic procedures), delays treatment, and unnecessarily increases the costs of medical care [5].

In order to alleviate the problem of under- or overconfidence in residents' diagnostic performance (i.e., poor calibration of judgment and performance), scaffolding needs to be provided to improve the accuracy of their metacognitive judgments (i.e., Feeling of Knowing, FOK) and eliminate any diagnostic bias. FOK is defined as the learner's certainty of his/her actual performance [6]. ITSs can play a significant role in assisting pathologists in making more accurate metacognitive judgments about their diagnostic decision-making and performance, and as a result make more accurate diagnoses.

One of the important methods of scaffolding and improving learners' metacognitive skills and performance is the use of open learner models (OLMs) in ITSs. A student model is an important part of an ITS which observes learner behavior and builds an individualized qualitative representation of her/his cognitive and metacognitive skills and gets updated in real-time during learners' interaction with the ITS [7]. Learner models are usually embedded in the ITS architecture and are not visible to the students, however, several researchers [e.g., 8] have investigated the benefits of allowing learners to access their learner model (OLM). Research has indicated that the mere displaying of visualizations of OLMs in ITS interfaces raises the awareness of the learners, allowing them to reflect on different aspects of their learning and problem solving. Besides all the advantages of using OLMs in interactive ITSs, according to [9], no study has investigated the use of OLMs for displaying metacognitive processes (e.g., metacognitive judgments of correctness of performance). In spite of the great potential and possibilities offered by the use of medical ITSs, few of these systems have been fully developed [e.g., 9] and only a fewer have been empirically evaluated [e.g., 10].

In this paper, we describe an adapted version of SlideTutor, an ITS which scaffolds pathology residents' accuracy of metacognitive judgments using different metacognitive interventions and an OLM for presenting metacognitive accuracy. The paper does not include our evaluation of the effectiveness of the implemented modules.

2 Description of the Medical ITS: SlideTutor

The SlideTutor intelligent tutoring system (http://slidetutor.upmc.edu) was modified for use in this study. The computational methods and the architecture of the original system have been previously published [11]. For the current study, the system uses a modular architecture implemented in the Java programming. SlideTutor provides users with cases to be solved under supervision by the system. Cases incorporate virtual slides, which are gigabyte size image files created from traditional glass slides by concatenating multiple images from a high resolution robotic microscope. Virtual slides are annotated using a custom built editing environment to produce case representations of discrete findings and their locations. A separate Ontology Web Language (OWL) based expert knowledge base consists of a comprehensive set of evidence-diagnosis relationship for the entire domain of study. A reasoning module uses a decision tree approach to construct a dynamic solution graph (DSG), representing the current state of the problem and all acceptable next steps including the best-next-step. As for the interface, participants use a graphical user interface (Fig. 1) to examine and diagnose the cases. Participants can pan and zoom in the virtual slide, locate findings using the mouse, and select from lists of findings and qualifiers, such as size and type, from a tree-like representation. Once findings are specified, they appear as evidence nodes in the diagrammatic reasoning palette (Fig 1). Afterwards, participants assert hypotheses using a separate tree-based menu, which eventually appear as nodes in the diagrammatic reasoning palette. Support links can then be drawn between evidence and hypothesis nodes to specify relationships between the two. Finally, one or more hypotheses may be dragged to the diagnosis window, and selected as the final diagnosis(es) before proceeding to the next case.

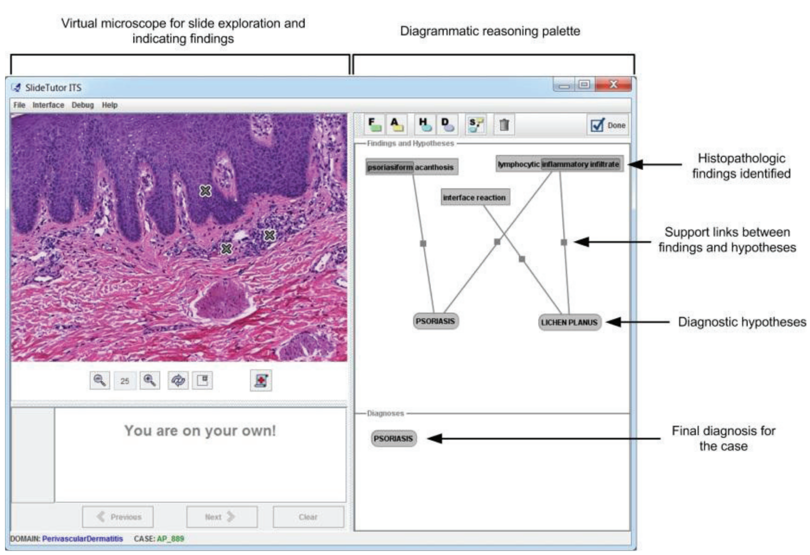


Fig. 1. SlideTutor interface

2.1 The Dynamic Book

An interactive knowledge browser has been developed (called the Dynamic Book) that shows feature-diagnosis relationships as well as glossary information on all features and diagnoses in the selected domain of dermatopathology (i.e., perivascular diseases) (Fig. 2). A description of the domain and the cases is presented in the next section. A total of sixty-two diagnoses and fifty-seven findings are presented in this interface. Six of the diagnoses comprising six patterns were used in the tutoring phase of the study. By clicking on each one of the diagnoses, an image is presented in the interface showing an example of how the disease presents on a patient's skin. A description of the diagnosis was also presented under the image. Additionally, a list of potentially associated findings is presented to the right of the image and diagnosis description. A zoomed-in virtual slide image accompanied each of the findings in the list, where the presentation of the finding is indicated by an arrow. A description of the particular finding together with a list of potentially associated diagnoses is also presented. In order to guide the exploration of participants during the Dynamic Book phase towards important parts of the book, they are provided with a list of tasks to work through which pertained to a mix of patterns they would encounter in the tutoring phase and ones they would not.

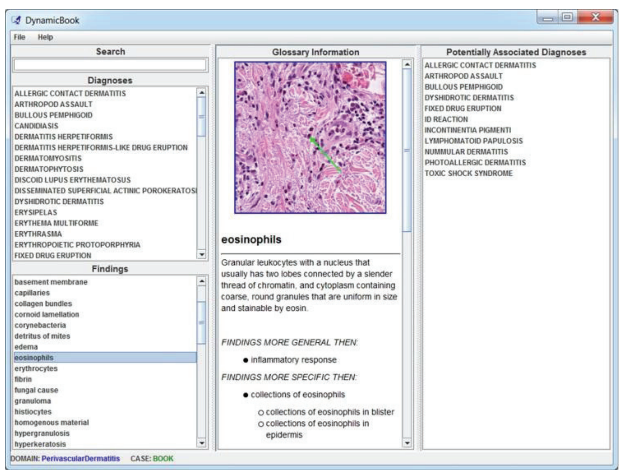


Fig. 2. Dynamic book interface

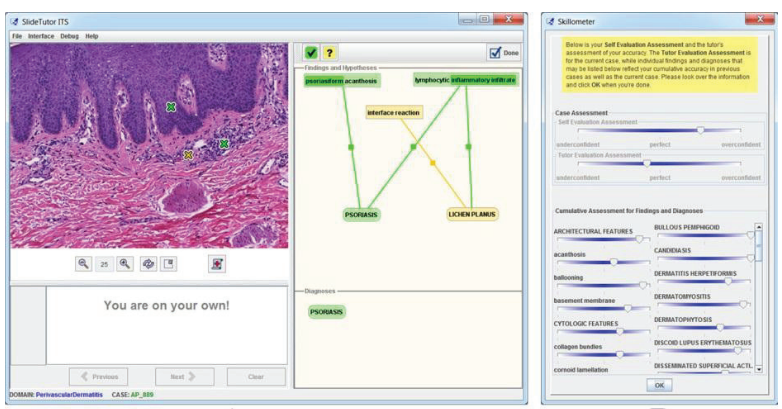
2.2. Pathology Cases

The Perivascular Dermatitis domain was selected for the current SlideTutor study because the domain is well-tested, includes patterns (i.e., a combination of evidence identified in a particular case) with multiple cases, and more cases are available than other domains. Also, Perivascular Dermatitis is a large domain and it is unlikely that participants would have complete knowledge of this diagnostic area. 20 cases were used for the tutoring phase. Cases were obtained from the University of Pittsburgh

Medical Center (UPMC) slide archive and from private slide collections. Diagnoses were checked and confirmed by a dermatopathologist prior to inclusion in the system repository. For each case, a knowledge engineer and an expert dermatopathologist collaborated in defining all present and absent findings, their locations on the slide (case annotation), and relationships among findings and diagnoses (knowledge-base development). Each diagnosis included a set of one or more diseases that matched the histopathologic pattern.

2.3 The Coloring Book and Metacognitive Judgments

For the intervention condition, once participants complete identifying findings, hypotheses, and diagnoses for a case, they progress to an interface called the Coloring Book (Fig. 3A). In this interface, they indicate if they are sure or unsure of the items they identified for the case (i.e., FOK judgments) by clicking on them and coloring them as either green (sure) or yellow (unsure). Next, they are presented with a window with a slider where they indicate how accurate they think their self-assessments in the coloring book were (ranging from underconfident to overconfident). Afterwards, they are presented with correct findings, hypotheses, and diagnoses for the respective case (colored in green) and incorrectly identified items as red. After reflecting on their performance and the feedback from the system, they are presented with a window juxtaposing the sliders for their self-assessment of their FOK judgments and the evaluation of the tutor based on their performance and their FOK judgments (the open learner model: OLM) (Fig 3B). At the bottom of the window, one or more individual findings or diagnoses may be listed, which reflects the participant's cumulative accuracy in previous cases as well as the current case for the particular finding or diagnosis. At the end, they are asked to make another metacognitive judgment and state whether they would feel confident solving similar cases, to which they respond on a 6-point Likert scale ranging from "not confident" to "very confident". This concludes the case, and progresses them to the next case.



A B

Fig. 3. Coloring book interface (A) and the OLM (B)

3 Study Timeline

As part of the design of the study and interface of the ITS, the study phases and timeline were determined as follows (Fig. 4). An approximate total time of four hours was allocated as the participant session time. At the beginning and after signing the informed consent form, the participants were administered a test (pre-pre-test) of their prior knowledge of the domain targeted by the current version of SlideTutor (i.e., Perivascular diseases). Next, they spent 30 minutes acquiring cognitive knowledge of the domain while accomplishing a task given to them by the experimenter (Dynamic Book phase). Afterwards, another test of cognitive knowledge of the domain was administered (pre-test). Once the test was completed, they proceeded to the tutor training and tutor use phase (in intervention or control condition) where they solved 20 cases and indicated their confidence in their responses and were shown an OLM (intervention condition), or solved the cases and progressed with no feedback from the system (control condition). At the end, a post-test was administered to gauge their knowledge gains during interactions with the tutor. A detailed description of the ITS, the tests, dynamic book, and the tutoring interventions is presented below.

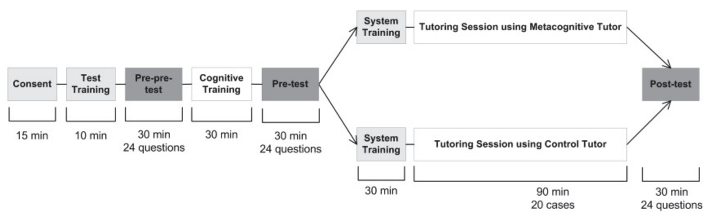


Fig. 4. Study timeline

4 Measures

4.1 Cognitive Measures

In order to measure the prior cognitive knowledge of the domain at the beginning of the tutoring session, cognitive gains after the cognitive learning phase, and the knowledge gains after the tutoring session, three 24-item tests were administered. Three versions of each test were created, and the test order was randomized per session to control for order effects. Each test comprised of 24 questions, and the questions were a mix of tutored and untutored items. Tutored items were about the material that was presented in the cases seen with the tutoring system, while untutored items were about material that was not covered by the tutoring system. Three question types were used in the tests: finding, diagnosis, and differentiate questions. Finding questions consisted of a static microscopic image with an arrow pointing at a feature to be identified. Diagnosis questions consisted of a list of findings, and participants had to provide the diagnosis(es) that match the findings. Differentiate questions consisted of two diagnoses, and participants had to provide a feature that can be used to differentiate the two. After responding to each question, participants were asked to rate if they were sure or unsure of their responses using radio buttons (FOK metacognitive judgment).

4.2 Metacognitive Measures

Feeling of knowing (FOK) metacognitive judgment measures were collected on all test items in the three cognitive knowledge tests and on all findings, hypotheses, and diagnoses identified in cases in the tutoring phase. The FOK measures were collected as binary values: sure vs. unsure. The data from metacognitive ratings on test questions were only used for analyses after the study was completed. However, the metacognitive judgment ratings for items identified in cases in the tutoring phase in the Coloring Book layout (see section 2.3) were used for calculation of a measure of over- or under-confidence called Bias, which was presented to the participant after solving the case and indicated their confidence in the items they identified in the case (in the OLM: see section 2.3). The bias score is calculated by subtracting the relative performance on all items (total correct items divided by all items) from the proportion of items judged as known (total sure items divided by all items) [12]. Figure 5 indicates how bias scores are calculated. Positive bias scores indicate over-confidence and negative scores indicate under-confidence. When performance perfectly matches the rated confidence level, the bias score equals zero. In other words, the bias score indicates the direction and degree of lack of fit between confidence and performance [13]. The bias score for each case was presented to the participant in the form of a slider ranging from under-confident to perfect to over-confident with a cursor indicating the participant's bias score.

		Performance: Correct	Performance: Incorrect
Feeling of Knowing	Sure	True Positive a	False Positive B
	Unsure	False Negative c	True Negative D

Bias = Confidence – Judgment

$$= \frac{a+b}{a+b+c+d} - \frac{a+c}{a+b+c+d}$$

Fig. 5. FOK contingency table and the calculation of bias

5 Conclusion

We described the functionalities of a version of SlideTutor aimed at reducing the metacognitive bias of pathologists and dermatologists while diagnostic decision-making by deploying metacognitive interventions and using an open learner model to aid participants in reflecting on their diagnostic performance. Open learner models have not been used in the previous studies for displaying the metacognitive performance of participants [8], and the current iteration of SlideTutor is novel in this regard. The Dynamic Book interface used for the cognitive learning phase provided participants with an environment to conduct a targeted search and knowledge acquisition (targeted at completing the task assigned by the experimenter). As mentioned above, since the domain chosen for this version of SlideTutor is a very large domain, a cognitive learning phase was deemed necessary in order to provide the opportunity for acquisition of some cognitive knowledge and freely explore the glossary of diagnoses and findings.

Acknowledgment
The authors gratefully acknowledge the support of this research by the National Library of Medicine through grant number 5R01LM007891.

References

1. Koedinger, K., Aleven, V., Roll, I., & Baker, R. (2009). In vivo experiments on whether supporting metacognition in intelligent tutoring systems yields robust learning. In A. Graesser, J. Dunlosky, D. Hacker (Eds.), *Handbook of metacognition in education*. 383-413. Mahwah, NJ: Erlbaum.
2. Azevedo, R., & Hadwin, A. F. (2005). Scaffolding self-regulated learning and metacognition: Implications for the design of computer-based scaffolds. Instructional Science, 33, 367-379.
3. VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educational Psychologist, 46(4), 197-221.
4. Graber, M. L., Franklin, N., & Gordon, R. (2005). Diagnostic error in internal medicine. Archives of Internal Medicine, 165 (13), 1493-1499.
5. Berner, E. S. & Graber, M. L. (2008). Overconfidence as a cause of diagnostic error in medicine. The American Journal of Medicine, 121 (5A), S2-S23.
6. Metcalf, J., & Dunlosky, J. (2008). Metamemory. In H. Roediger (Ed.), Cognitive psychology of memory (Vol. 2, pp. 349-362). Oxford: Elsevier.
7. Bull, S. (2004). Supporting Learning with Open Learner Models. Proceedings of 4th Hellenic Conference with International Participation: Information and Communication Technologies in Education, Athens, Greece. Keynote.
8. Bull, S., & Kay, J. (in press). Open learner models as drivers for metacognitive processes. In R. Azevedo & V. Aleven (Eds.). International handbook of metacognition and learning technologies. Amsterdam, The Netherlands: Springer.
9. Azevedo, R., & Lajoie, S. (1998). The cognitive basis for the design of a mammography interpretation tutor. International Journal of Artificial Intelligence in Education, 9, 32-44.
10. El Saadawi, G. M., Tseytlin, E., Legowski, E., Jukic, D., Castine, M., Fine, J., . . . Crowley, R. S. (2008). A natural language intelligent tutoring system for training pathologists: implementation and evaluation. Advances in Health Sciences Education: Theory and Practice, 13, 709-722.
11. Crowley, R. S., & Medvedeva, O. (2006). An intelligent tutoring system for visual classification problem solving. Artificial Intelligence in Medicine, 36 (1), 85-117.
12. Kelemen, W. L., Frost, P. J., & Weaver III, C. A. (2000). Individual differences in metacognition: Evidence against a general metacognitive ability. Memory & Cognition, 28(1), 92-107.
13. Schraw, G. (2009). A conceptual analysis of five measures of metacognitive monitoring. Metacognition and Learning, 4(1), 33-45.

Teacher Perspectives on the Potential for Scaffolding with an Open Learner Model and a Robotic Tutor

Aidan Jones, Susan Bull and Ginevra Castellano

Electronic, Electrical and Computer Engineering, University of Birmingham, UK
axj100@bham.ac.uk, s.bull@bham.ac.uk, g.castellano@bham.ac.uk

Abstract. This paper considers the potential for scaffolding learning in open-ended learning environments using a robotic tutor and an open learner model. While we expect this approach to be more broadly applicable, we here illustrate with a map-reading activity in geography and/or environmental sciences. The paper presents issues raised in open-ended teacher interviews, which suggest real possibilities for incorporating a robotic tutor together with an open learner model in the classroom.

Keywords: affect detection, open learner model, scaffolding, social robotics

1 Introduction

Open learner models (OLM) externalise the learner model in a way that is interpretable by the user, e.g.: skill meters [16], concept maps [19], treemaps [14]. One of the aims of opening the learner model to the learner is to help promote reflection on the part of the learner; to facilitate their planning and decision-making; and raise their awareness of their understanding or their developing skills [3]. Thus, the OLM can be seen as a form of scaffolding for cognitive and metacognitive processes, with a particular focus on supporting and developing self-regulation. This focus is very much in line with previous considerations of tools offering scaffolding (see e.g. [1]). This approach to supporting the learner can be very light or can be more closely guided, depending on the level of detail of the modelling and the visualisation of the model, as well as the goals of the interaction and the user's current learning needs.

Most learner models that are inspectable by the learner have focussed on knowledge-related attributes. However, despite it being a difficult task, there is growing interest in detecting and responding to affective states (e.g. [6]; [24]; [25]), and increasingly with a goal of adaptive scaffolding to support individual differences [10]. A taxonomy of "academic emotions, which are directly related to academic learning, classroom instruction or achievement", has been identified [17]: the positive activating emotions of enjoyment, hope, and pride; the positive deactivating emotion of relief; the negative activating emotions of anger, anxiety, shame; and the negative deactivating emotions of hopelessness and boredom.

OLMs can offer an additional mechanism by which learner model data about affective states can be confirmed and/or clarified. In addition to visualisation of the learner model, the term 'open learner modelling' encompasses methods that allow users to contribute to, edit, or negotiate the contents of the learner model [3]. While we do not wish to require or rely on self-report about emotions and affective states, if a learner is frustrated by feedback that has been generated in part based on inaccurate or incomplete affect detection, a simple method to advise a learning environment of this could be of substantial benefit. Thus, while providing an OLM of the more traditional knowledge/skills representations, we recommend also allowing the learner to access the representations regarding their affective state (e.g. inferred through sensors [24], semantic and contextual cues [25], or based on a video corpus of affective expressions [7]). This may bring new issues to the problems of affect modelling (e.g. if the learner model indicates an affective state that the learner disagrees with, might this make them angry, demotivated or frustrated?) Nevertheless, as well as offering an opportunity to modify or influence the representation of affect, it may also help increase learner trust in the learner model, as the user will be able to identify why certain aspects of feedback or scaffolding are tailored in the manner that they are, and have the opportunity to address or challenge any discrepancies. In this paper we take the starting point of benefits previously demonstrated for OLMs (e.g. [12]; [16]), and consider their use in a more open-ended context, and with affect modelling.

2 Scaffolding with an Open Learner Model

As argued above, OLMs can be considered as ways to help scaffold learning and the learning process, and may have particular potential in open-ended tasks and environments. With the increasing focus on professional competency frameworks and the inevitable extension of the competency perspective to educational contexts (e.g. for language [8], for STEM literacy [2], for geography [21]), there comes even greater scope for future use of open-ended learning environments, and corresponding challenges for scaffolding learning in such situations. Competency frameworks have already been applied in a generic OLM context, with examples for language [4] and meeting facilitation [20]. We propose that such approaches be further developed to meet the requirements of the changing educational focus, curricula, and assessment.

We illustrate here with a geography and/or environmental science map-based activity, where tools may be used to discover information from a map, to measure distance and area, to view terrain or entities on the map such as buildings, cities and countries. The learner may identify features, follow directions in a trail, explore the area, or determine the best location for some purpose (e.g. where to situate a new visitor centre). Such activities can range from specific to very open-ended, and a range of competencies may be demonstrable (e.g. map-reading, map sketching, mapping, geographical argumentation, ethical judgement (see [21]).) This relates closely to the England and Wales National Curriculum for Geography [9] key processes, e.g.:

"Pupils should be able to:

- use atlases, globes, maps at a range of scales, photographs, satellite images and other geographical data;
- ask geographical questions, thinking critically, constructively and creatively;

- analyse and evaluate evidence, presenting findings to draw and justify conclusions;
- solve problems and make decisions to develop analytical skills and creative thinking about geographical issues."

However, the nature of this type of open-ended activity may also lead to different affective states across and within individuals. In the next section we consider the opportunities for improving scaffolding using OLMs that include representations of affective states, supported by an empathic robotic tutor.

3 Support from a Robotic Tutor

Opening up a system's representations of a learner's affective state could, as indicated above, further influence learner affect. To mitigate a possibly negative reaction that could impact motivation, we recommend taking a social robotics approach. Artificial tutors may incorporate their understanding of the learner's emotional state in their pedagogical strategies and interventions [5]. The presence of a 2D or 3D character has revealed some positive learning effects, especially in engagement [15]; and recall has been shown to be higher with a robotic teacher when adaptive cues have been given based on EEG measurements of engagement [22]. Studies that compared virtual representations of characters with robots showed a preference for robotic embodiment with reference to social presence [13], enjoyment [18] and performance [11]. Thus, we suggest this to be a useful avenue to explore for scaffolding learning particularly when affective states are also modelled. For example, Figure 1 shows the Nao Robot and its ability to point or gesture towards items on a tabletop, which include visualisations of the learner model. Since many of the activities we envisage are map-based, we will use an interactive map approach on a touch table in this example.

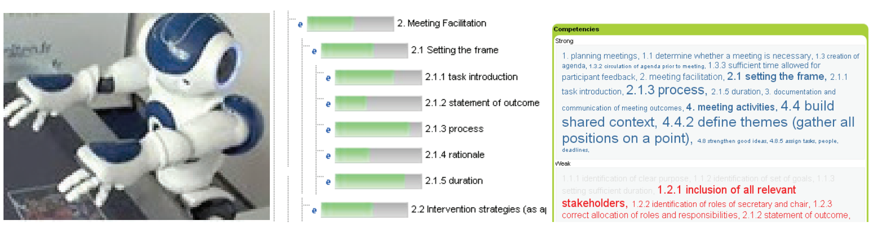


Fig. 1. The Nao Robot and a competency-based open learner model (skill meters and word cloud shown, from the Next-TELL open learner model [20])

Examples of general interactions and scaffolding between the learner and the robot include: offering assistance by guiding the learner through instructions; asking questions (to prompt reflection); gestures (to illustrate or focus attention, or indicate shared focus); offering affective support if learners' actions are not optimal (telling them not to worry and try again); drawing attention back to task if a learner becomes distracted; mirroring affective state when this is positive, and bringing awareness to affective state if it is negative. This aims to foster a perception of the robot as empathic (see e.g. [7]).

In addition to the learner model visualisations on the tabletop, the robot can itself express the model content by giving a summary of relevant knowledge or competencies, perhaps at the start of a session to show that it remembers the learner, but also during a session to give the learner a sense of achievement and to prompt them to think about their learning and how they might use the learner model information. As with adaptive scaffolding in general, interaction about the learner model will be tailored as appropriate to the individual, as will other scaffolding behaviours from the robot.

When using the OLM to investigate its representations of their affective state, the learner will already be accustomed to the robot's shared understanding of their competencies. Therefore, when it then comes to reviewing affective states in the model, the robot's ability to invite or allow discussion or adjustment to the affective model contents can build on the relationship that the learner has with the robot, with reference to their understanding or competencies. This approach will build on previous findings using a chatbot, that child-system negotiation of the knowledge-focussed data in an OLM resulted in significant improvements in children's learning without additional tutoring [12]. In that case negotiation involved student or system challenges and discussion about the child's beliefs (representations in the learner model) with the aim of prompting reflection and increasing the accuracy of the learner model by taking students' opinions about their learning into consideration. In our current work we propose also encouraging the learner to think about their affective state, how this may influence their learning, and how they might regulate their affect. In effect, this is an approach to help learners self-scaffold during the transition from more tightly to less tightly guided interaction. The first step towards this goal involves obtaining teacher viewpoints on the potential of this approach in the classroom. This is considered in the next section.

4 Teacher Interviews

Following from the arguments above that suggest possibilities for scaffolding in open-ended tasks using an OLM together with an empathic robot, teacher interviews were undertaken to determine the likelihood of uptake of this approach in contexts where the required technologies are already in place.

4.1 Participants, Materials and Methods

Seven participants took part in open interviews (4 teachers, 2 teaching assistants, 1 trainee). The aims of the study were described, highlighting emphatic tutoring and interaction, and personalising robotic tutoring to the learner's needs. In a semi-formal interview, specific questions relevant to scaffolding and OLMs included:

- What role would a system like this play? (To ascertain teachers' views on how the robot could effectively 'fit' into the classroom and classroom practice.)
- If you had a robot that could monitor how a child is progressing, how would you like that robot to interact with the child? (To provide information for the design of the learning scenarios and robot interactions.)
- Would it be beneficial to set the level of difficulty and how do you do this at the moment? (To gauge the extent of teachers' likely acceptance of a coarse-grained personalisation approach with a robotic tutor.)

- How do you detect when a student is having difficulties and how do you help the learner overcome the difficulties? (To ascertain how teachers detect when a learner is facing difficulties in this kind of open-ended activity, and whether they may be receptive to more fine-grained adaptation with the robotic tutor.)

Written notes were made by the researcher. Comments were then categorised to help design subsequent formal interviews before building the prototype environment.

4.2 Results

Table 1 summarises the number of teachers expressing each of the points addressed below, following the comment categorisations, with representative viewpoints then discussed further. Several teachers were very interested in the fact that they could use the system to encourage independent learning, as this is becoming a key objective for teachers. To address the varied needs of students, at the moment the teachers might give out different question sheets to different students. Typically the teachers change the difficulty of an activity by changing the language style, the number of prompts, breaking down the activity into smaller steps, and the amount of scaffolding provided. The most difficult questions or problems may be very open-ended, and require the learner to argue a point in their own words, or the teacher may apply extra constraints such as working within a budget. All teachers were keen that the system to be trialled should be able to respond to the individual, stretching the most able while also ensuring suitable personalisation for the less advanced students.

Table 1. Teacher comments categorised

Comment	No. teachers
Encourage independent learning	3
Personalisation / adaptivity	7
More open-ended activities	7
Prompt metacognitive behaviours	7
Affect detection	7
Use of progress bars	2
Incorporation of robot into classroom	7

In addition, all teachers stated that they would like the learning activities they undertake to easily move beyond basic map reading skills to activities where the learner needs to make comparisons, decisions and arguments. Comparisons in this space could be to compare high and low CO_2 production, population density, and similar. Decisions and arguments could be made on tasks which involve, for example, deciding on the most appropriate location for a visitor centre or flood defence: the learner must make an argument in favour or against an action. Thus, the teachers are looking for ways to incorporate more open-ended activities into the classroom interaction. All also wanted to encourage reflection and metacognitive behaviours, for example, by saying “Have a think”, “Did you consider...?”. They also thought that the robot could usefully point out that there is no really wrong answer in some of the activities.

All teachers already detect whether a learner is having difficulties, from their behaviour. For example: the teacher can tell if the learner is not listening, not paying attention or not understanding. This information can come from their facial expression, where they are looking, whether they are fidgeting, how they respond to instructions, whether they are actively asking for help verbally or by raising their hand, or if they are chatting or disrupting other children. The teachers can also identify whether a learner is attempting a task in a sub-optimal way.

Two teachers suggested that progress bars may be beneficial. They stressed that real time assessment would be desirable, and if a learner faced difficulties, these need to be caught promptly and acted upon as appropriate by the system or the teacher.

There were no concerns from any of the teachers about fitting the robot into the classroom activities, particularly if the lesson plan actively included the robot (e.g. as a station in a station rotation lesson where a number of learners in a class would have a turn with the robot). The teachers were interested in monitoring the learner's progress from a console, enabling the teacher to intervene if the learner stopped making progress, particularly useful if there were multiple learners interacting with multiple artificial tutors. They also thought that the simple fact that there was a robot would make any task seem novel and more engaging.

4.3 Discussion

Because the interviews were open, not all points were discussed in each interview. The lower level of comments in some areas therefore may not indicate disagreement, but rather that these issues were not raised during the interview.

The possibility for the robot to adapt to individuals, as requested by all participants, is exactly the kind of approach enabled by a learner model. For this reason the learner model is anticipated to be acceptable to teachers in this robot-tabletop context. All teachers also wished to use open-ended tasks such as described above, to match the requirements of the England and Wales National Curriculum for Geography [9]. This is, therefore, another indication of likely acceptance. Furthermore, because teachers are already identifying student engagement and other affective states, the modelling of affect and use of a physical robot is an approach that they will understand: while they may not be able to discuss knowledge and competencies individually to the extent they wish, a robotic tutor can help in this task while maintaining an approximation of the empathic approach a teacher would use. The fact that two teachers suggested progress bars indicates that these participants wish to have a view of learning visible on the tabletop, in line with OLM. In addition, the OLM should facilitate the kind of metacognitive behaviours considered important by all teachers. The request for being able to monitor learners is also in line with OLM also being a tool to support teachers [20]. This goes beyond many learning analytics visualisations (e.g. dashboards [23]), to focus on understanding, competencies, and now also affect.

An important immediate concern is practical deployment in the existing learning context and curriculum. All teachers could see how the robot and touch table could be integrated into the classroom, and could identify benefits for doing so. Thus we argue that there is a role for empathic robots and OLMs in scaffolding open-ended learning.

5 Summary and Conclusions

This paper has argued the benefits of using an OLM as a means to lightly scaffold learners in open-ended learning contexts where the development of self-regulation skills and metacognitive behaviours are considered important. This is becoming increasingly central with the competency focus adopted in many subjects and countries. Affective modelling is considered beneficial in such contexts, given the potential frustrations of the open-ended nature of activities, and the provision of a means to discuss and possibly correct the system representations of affect is suggested. Because of the advantages of robotic tutors, an empathic robot approach is proposed. The teacher interviews confirmed the feasibility of introducing this solution to real classrooms that have the appropriate technologies.

Acknowledgements

This work was partially supported by the European Commission (EC) and was funded by the EU FP7 ICT-317923 project EMOTE. The authors are solely responsible for the content of this publication. It does not represent the opinion of the EC, and the EC is not responsible for any use that might be made of data appearing therein. We thank the participating teachers from Haybridge High School and Stourport High School, Worcestershire, UK.

References

1. Azevedo, R. Using Hypermedia as a Metacognitive Tool for Enhancing Student Learning? The Role of Self-Regulated Learning, Educational Psychologist 40(4), 199-209 (2005)
2. Bybee, R.W.: Advancing STEM Education: A 2020 Vision, Technology and Engineering Teacher 70(1), 30-35 (2010)
3. Bull, S. & Kay, J.: Open Learner Models, in R. Nkambou, J. Bordeau & R. Miziguchi (eds), Advances in Intelligent Tutoring Systems, Springer, 318-338 (2010)
4. Bull, S., Wasson, B., Kickmeier-Rust, M., Johnson, M.D., Moe, E., Hansen, C., Meissl-Egghart, G. & Hammermuller, K.: Assessing English as a Second Language: From Classroom Data to a Competence-Based Open Learner Model, International Conference on Computers in Education (2012)
5. Burleson, W. Affective Learning Companions: Strategies for Empathetic Agents with Real-Time Multimodal Affective Sensing to Foster Meta-Cognitive and Meta-Affective Approaches to Learning, Motivation, and Perseverance, PhD Thesis, MIT (2006)
6. Calvo, R.A. & D'Mello, S.: Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications, IEEE Transactions on Affective Computing 1(1), 18-37 (2010)
7. Castellano, G., Leite, I., Pereira, A., Martinho, C., Paiva, A. & McOwan, P/.W.: Multimodal Affect Modelling and Recognition for Empathic Robot Companions, International Journal of Humanoid Robotics 10(1) (2013)
8. Council of Europe.: The Common European Framework of Reference for Languages, http://www.coe.int/t/dg4/linguistic/Cadre1_en.asp. Accessed 18 March 2013.

9. Department for Education. Secondary National Curriculum until 2014 (Geography), http://www.education.gov.uk/schools/teachingandlearning/curriculum/secondary/b00199536/geography. Accessed 29 April 2013.
10. Desmarais, M.C. & Baker, R.J.D.: A review of Recent Advances in Learner and Skill Modeling in Intelligent Learning Environments, User Modeling and User-Adapted Interaction 22, 9-38 (2012)
11. Hoffmann, L. & Krämer, N.C. How Should an Artificial Entity be Embodied? Comparing the Effects of a Physically Present Robot and its Virtual Representation, Proceedings of Workshop on Social Robotic Telepresence, HRI (2011)
12. Kerly, A. & Bull, S.: Children's Interactions with Inspectable and Negotiated Learner Models, in B.P. Woolf, E. Aimeur, R. Nkambou & S. Lajoie (eds), Intelligent Tutoring Systems, Springer-Verlag, Berlin Heidelberg, 132-141 (2008)
13. Kidd, C. D.: Sociable Robots: The Role of Presence and Task in Human-Robot Interaction (Doctoral dissertation, Massachusetts Institute of Technology) (2003)
14. Kump, B., Seifert, C., Beham, G., Lindstaedt, S.N. & Ley, T. Seeing What the System Thinks You Know Visualizing Evidence in an Open Learner Model, LAK, ACM. (2012)
15. McQuiggan, S.W. & Lester, J.C. Modeling and Evaluating Empathy in Embodied Companion Agents, International Journal of Human-Computer Studie, 65, 348–360 (2007)
16. Mitrovic, A., Martin, B.: Evaluating the Effect of Open Student Models on SelfAssessment. International Journal of Artificial Intelligence in Education 17(2), 121--144 (2007)
17. Pekrun, R., Goetz, T., Titz, W. & Perry, R.: Academic Emotions in Students' Self- Regulated Learning and Achievement: A Program of Qualitative and Quantitative Research, Educational Psychologist 37(2), 91-105 (2002)
18. Pereira, A., Martinho, Leite, C.I. & Paiva, A. iCat the Chess Player: the influence of embodiment in the enjoyment of a game, Proceedings of 7th International Joint Conference on AAMAS , Estoril, Portugal, 1253-1256 (2008)
19. Perez-Marin, D., Alfonseca, E., Rodriguez, P., Pascual-Neito, I.: A Study on the Possibility of Automatically Estimating the Confidence Value of Students' Knowledge in Generated Conceptual Models, Journal of Computers 2(5), 17-26 (2007)
20. Reimann, P., Bull, S. & Ganesan, P.: Supporting the Development of 21st Century Skills: Student Facilitation of Meetings and Data for Teachers, in R. Vatrapu, W. Halb & S. Bull (eds), Proceedings of the Workshop Towards Theory and Practice of Teaching Analytics, EC-TEL 2012, CEUR Workshop Proceedings http://ceur-ws.org/Vol-894 (2012)
21. Rempfler, & Uphues, R.: System Competence in Geography Education Development of Competence Models, Diagnosing Pupils' Achievement, European Journal of Geography 3(1), 6-22 (2012)
22. Szafir, D. & Mutlu, B. Pay Attention! Designing Adaptive Agents that Monitor and Improve User Engagement, Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems ACM, 11-20 (2012)
23. Verbert, K., Duval, E., Klerkx, J., Govaerts, S., Santos, J.L.: Learning Analytics Dashboard Applilations, American Behavioral Scientist, early online version Feb 28, DOI: 10.1177/0002764213479363, (2013)
24. Woolf, B.P., Arroyo, I., Cooper, D., Burleson, W. & Muldner, K.: Affective Tutors: Automatic Detection of and Response to Student Emotion, in R. Nkambou, J. Boureau & R. Mizoguchi (eds), Advances in Intelligent Tutoring Systems, Springer-Verlag, Berlin Heidelberg, 207-227 (2010)
25. Zhang, L.: Exploration of Affect Detection Using Semantic Cues in Virtual Improvisation, in S.A. Cerri, W.J. Clancey, G. Papadourakis & K. Panourgia (eds), Intelligent Tutoring Systems, Springer-Verlag, Berlin Heidelberg, 33-39 (2012)

Metacognitive Tutoring for Scientific Modeling

David A. Joyner, Ashok K. Goel, and David M. Majerich

Design & Intelligence Laboratory, School of Interactive Computing
Georgia Institute of Technology, Atlanta, Georgia, USA
david.joyner@gatech.edu; ashok.goel@cc.gatech.edu;
dmajerich6@mail.gatech.edu

Abstract. In this paper, we present a set of metacognitive tutors for teaching scientific inquiry-driven modeling. We describe the MeTA architecture in which the tutors are implemented and experiences with an initial pilot study.

Keywords: metacognitive tutors, intelligent tutoring systems, scientific modeling, middle school science.

1 Introduction

Supporting metacognition has been identified as one of the most important principles of instructional design [4]. In recent years, interventions using a variety of metacognitive skills have been studied. Aleven et al. examine the use of a metacognitive tutor for help seeking within a cognitive tutor for geometry [1]. Some systems, such as MetaTutor, focus on teaching students self-assessment skills to identify knowledge gaps or monitor their own progress [3,10]. Betty's Brain can teach students metacognitive skills by having them request that Betty engage in those skills herself [11]. These projects have shown the success of tutoring interventions based on developing metacognitive skills.

Inquiry-based learning has long been pursued as a desirable approach to classroom curriculum design [6], and significant efforts have been made to incorporate authentic scientific modeling and inquiry into science education, such as in projects like Thinker Tools [13]. This paper presents our early efforts to construct a metacognitive tutoring system specifically aimed at teaching these skills within an open-ended learning environment named MILA (for Modeling & Inquiry Learning Application).

2 Tutoring Scientific Inquiry-Driven Modeling in MILA

MILA (**M**odeling & **I**nquiry **L**earning **A**pplication) is an interactive learning environment for supporting learning about ecosystems in middle school science. Students use MILA to construct Component-Mechanism-Phenomenon models of complex ecological phenomena. Component-Mechanism-Phenomenon models are adaptations of Structure-Behavior-Function models [7,12], and MILA evolves from our earlier work on learning Structure-Behavior-Function models of ecosystems [8,12].

To support students' modeling and inquiry while engaging with MILA, we constructed a metacognitive tutoring system consisting of four separate metacognitive tutoring agents playing four different functional roles: a Guide, a Critic, a Mentor, and an Interviewer. Broadly, these tutors were constructed according to lessons and guidelines transferred from other initiatives in metacognitive tutoring [2,10]. Students interact with tutors by clicking tutors' avatars in the tutor box. Upon clicking, the tutor's window appears and gives the student any feedback it has available, as shown in Figure 2. Reactive tutors checks their Mappings when the student clicks in order to respond to students' help-seeking behaviors [1]. A proactive tutor actively monitor students' progress and interrupt the students to provide their feedback or ask their question in order to facilitate just-in-time error correction [10].

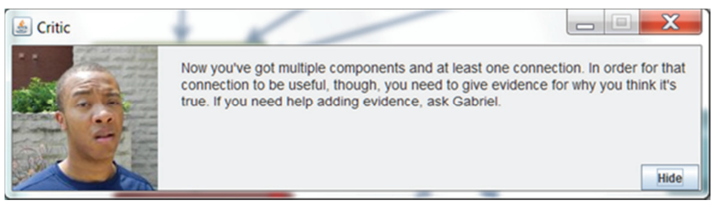


Fig. 2. An example of one of the four tutors, the Critic. All tutors appear in dialog boxes such as this one. In addition to text feedback, tutors may ask students to answer questions or offer students questions they might want answered.

The Guide serves to answer students' questions, and thus is a reactive tutor. She is developed to anticipate what questions students may want to ask based on the current lesson, the students' current model, software, and tutor interactions and then offer those questions when called. For example, early in the unit, the Guide anticipates questions that largely focus on interaction with the software itself. Later, she expects and offers questions based on students' current models or recent model construction process.

The Critic analyzes students' models, validating students' models against a set of defined model criteria. He is a reactive tutor who only checks models when students are looking for feedback, demonstrating the knowledge gaps of which students should be aware in model construction and providing guidance on how to fill those knowledge gaps, as well as avoid them in the future.

The Mentor leverages the notion of cognitive apprenticeship [5]. He is a proactive tutor who observes students' interaction with the software and demonstrates new or difficult concepts. In practice, the main role of the Mentor has been to set expectations and learning goals, addressing Roll et al.'s eighth design principle: communicate the metacognitive learning goals to the students [10].

Completing the set of four tutors is the Interviewer. The Interviewer asks students to answer questions in natural language. The Interviewer serves the metacognitive goal of encouraging students to self-reflect on their process by prompting students to elucidate their decision-making.

3 The Architecture of MeTA

This set of metacognitive tutors for teaching inquiry-driven modeling has been constructed in an experimental architecture titled MeTA, for Metacognitive Tutoring Architecture. At a basic level, the MeTA architecture builds on the characterization of an intelligent agent as a function f that maps a history of percepts P* into an action A; f: $P^* \rightarrow A$. This section describes MeTA at a software architecture level, consisting of percepts, actions, and mappings between them.

Percepts are defined information the tutor can sense in the learning environment. We have used six categories of percepts for constructing our tutors, including history software and tutor interaction and a current model of student behavior. Actions, in turn, are output complements to the input percepts. Whereas percepts tell tutors for what to look for, actions tell them how to respond. We have used six different categories of actions, including textual feedback, soliciting further information, and altering an underlying model of student behavior. Mappings pair up sets of Percepts with sets of Actions. When every Percept in a given Mapping is observed, the tutor responds with the associated Actions. In many ways, individual tutors can be seen as prioritized lists of Mappings.

4 Initial Deployment & Results from MeTA in MILA

MILA was used in a two-week camp in Summer 2012 with 16 middle school students. The phenomenon that students were charged with explaining was the actual, sudden death of thousands of fish in a nearby lake. To investigate this problem, students took field trips to the lake, participated in physical science and biology exercises, and engaged with MILA in groups of two or three. MILA provided facilities for stating the problem, proposing multiple hypotheses, modeling those hypotheses, consulting static simulations, and researching online hypermedia and data sources. Given that this was the first use of MeTA tutors in a classroom, data gathering and analysis was treated as an exploratory study; the goal, in line with design-based research, was to observe the strengths and weaknesses to better understand how to create effective metacognitive tutors in the future. We found two primary guidelines that are informing our ongoing revisions to the tutoring systems. First, our experience deploying tutors that play multiple functional roles within the software directed our attention to the different ways in which students interact with different roles and types of feedback; this has been similarly touched on elsewhere in research on metacognitive tutoring [3,9]. This has led to the revision of these tutors for new interventions to better differentiate their functional roles and expand the range of types of feedback available. Secondly, we observed the need to address the challenge outlined in Roll et al. 2007 [10] regarding applying one of Anderson et al. 1995's original design guidelines [2] to the metacognitive tutoring domain. This principle – "Facilitate successive approximations of the target skill" – addresses the need to differentiate and address the student's current level of efficacy with the target skill, changing the way in which the skill is addressed as student efficacy changes. Ongoing revisions to the tutors outlined here attempt to equip the system with the ability to infer and address successive approximations of the target skill.

5 Acknowledgments

We are grateful to Kristina Strickland (Little Mill Middle School), Andrew Burch & Chris Thompson (CEISMC), Helen Ellis (Georgia Aquarium), Mary Moerlins (Piedmont Park), Dr. Megan Duffy, Dr. Marc Weissburg, Dr. Jeanette Yen, Nicolas Papin, and Cherish Weiler for their help. This work was partially supported by Proctor & Gamble. We are thankful to Dr. Gautam Biswas for his encouragement and support.

6 References

1. Aleven, V., McLaren, B., Roll, I., & Koedinger, K. (2004). Toward tutoring help seeking. *Intelligent Tutoring Systems, 19*, 105-154.
2. Anderson, J., Corbett, A., Koedinger, K., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences 4.* 167-207.
3. Azevedo, R., Witherspoon, A., Chauncey, A., Burkett, C., & Fike A. (2009). MetaTutor: A MetaCognitive tool for enhancing self-regulated learning. In *Procs. 23rd AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems.*.
4. Bransford, J., Brown, A., & Cocking, R. (2000). *How people learn: Brain, mind, experience, and school*. Washington, D.C.: National Academy Press.
5. Collins, A., Brown, J., & Holum, A. (1991). Cognitive Apprenticeship: Making Thinking Visible. *American Educator 15*(3).
6. Edelson, D., Gordin, D., & Pea, R. (1999). Addressing the challenges of inquiry-based learning through technology and curriculum design. *Journal of the Learning Sciences*, *8*(3-4), 391-450.
7. Goel, A, Rugaber, S, Vattam, S (2009) Structure, behavior & function of complex systems: the SBF modeling language. *AIEDAM,* 23:23-35.
8. Goel, A., Rugaber, S., Joyner, D., Vattam, S., Hmelo-Silver, C., Jordan, R., Sinha, S., Honwad, S., & Eberbach, C. (2013) Learning Functional Models of Complex Systems: A Reflection on the ACT project on Ecosystem Learning in Middle School Science. In *International Handbook on Meta-Cognition and Self-Regulated Learning*, R. Azevedo & V. Aleven (editors), pp. 545-560, Berlin: Springer.
9. Graesser, A., VanLehn, K., Rosé, C., Jordan, P., & Harter, D. (2001). Intelligent tutoring systems with conversational dialogue. *AI Magazine*, *22*(4), 39.
10. Roll, I., Aleven, V., McLaren, B., & Koedinger, K. (2007). Designing for metacognition—applying cognitive tutor principles to the tutoring of help seeking. *Metacognition in Learning 2*(2).
11. Schwarz, D., Chase, C., Chin, D., Oppezzo, M., Kwong, H., Okita, S., Biswas, G., Roscoe, R., Jeong, H., & Wagster, J. (2009) Interactive Metacognition: Monitoring and Regulating a Teachable Agent. In D.Jacker, J. Dunlosky, & A. Graesser (eds.), *Handbook of Metacognition in Education*.
12. Vattam, S., Goel, A., Rugaber, S., Hmelo-Silver, C., Jordan, R., Gray, S, & Sinha, S. (2011) Understanding Complex Natural Systems by Articulating Structure-Behavior-Function Models. *Journal of Educational Technology & Society, 14*(1): 66-81.
13. White, B. & Frederiksen, J. (1998). Inquiry, modeling, and metacognition: Making science accessible to all students. *Cognition and Instruction, 16*(1), 3-117.

Evaluation of a Data Mining Approach to Providing Adaptive Support in an Open-Ended Learning Environment: A Pilot Study

Samad Kardan[1] and Cristina Conati[1]

[1]Department of Computer Science, University of British Columbia
2366 Main Mall, Vancouver, BC, V6T1Z4, Canada

{skardan, conati}@cs.ubc.ca

Abstract. This paper describes the initial evaluation results for providing adaptive support based on effective/detrimental interaction patterns discovered by applying data mining on user interaction data for an Interactive Simulation. Previously, we presented the process of building a classifier user model for the AIspace CSP applet, an open-ended interactive simulation which helps with learning how to solve constraint satisfaction problems. In a later work, we presented a methodology for providing adaptive interventions based on the class association rules that form our classifier user model. In this work, we discuss how to use the generated adaptation rules for delivering adaptive support in the form of hints. The initial qualitative evaluation of the resulting support mechanism, as well as a quantitative evaluation using eye tracking and action logs, show that the interventions were well-received by users.

Keywords: Adaptive Interventions, Interactive Simulations, Eye Tracking

1 Introduction

Interactive Simulations (IS hereafter) are increasingly used as learning tools, where they present an open-ended and exploratory environment to support learning in many different disciplines. These ISs are designed to foster exploratory learning by giving students the opportunity to practically and proactively experiment with concrete examples of concepts and processes they have learned theoretically. However, it has been shown that if the students are left to experiment and explore without any additional support, many will show suboptimal interaction behaviors (e.g., [1]) and may not learn well from this form of interaction (e.g., [2]). These students can benefit from having additional support in the form of scaffolding while interacting with this type of Open-Ended Learning Environments (OELEs) (e.g., [3]). The Constraint Satisfaction Problem (CSP) Applet is one of the collection of interactive tools for learning common Artificial Intelligence algorithms, called AIspace [4]. The CSP applet is an Interactive Simulation designed to help students deepen their understanding of solving constraint satisfaction problems. We intend to add adaptive support to the CSP applet to help students use the applet effectively for learning. Implementing adaptive interventions requires adding two components to an OELE: (1) a **user model** that determines if and when to intervene, with additional information on which interventions are appropriate at the time; and (2) an **intervention mechanism** that delivers different interventions based on the assessment of the student model.

Due to the open-ended nature of the interactions with ISs, providing intelligent support is challenging because many different possible behaviors should be taken into account and most often it is not known a priori which behaviors are effective and which ones are not. All this makes developing a successful intelligent support mechanism time consuming [5]. To address these challenges in a timely and generalizable manner, we employ Educational Data Mining [6] methodologies. Our goal is to find relevant patterns in user interaction data in an IS (e.g. the CSP applet) that leads to different levels of user performance. Then, build a user model based on these patterns and finally, use these patterns to extract adaptation rules for delivering relevant adaptive interventions.

To achieve this goal, first we developed a user modeling framework that utilizes user clustering and class association rules mining to identify relevant user types/behaviors from interface actions [7]. Then, we devised a methodology for using the discovered association rules to generate adaptation rules which are then transformed to adaptive interventions [8]. This paper describes the initial evaluation of adaptive interventions that are implemented following our proposed process.

The rest of the paper is organized as follows: First, we briefly describe the CSP applet, the user modeling framework used for extracting user behaviors (i.e., the class association rules), and the methodology for generating adaptation rules based on these behaviors. Then, we discuss the different dimensions for providing interventions based on these adaptation rules. Finally, we present the results of a pilot study with a new version of the CSP applet that implements the proposed support mechanism.

2 The AIspace CSP applet

A CSP consists of a set of variables, variable domains and a set of constraints on legal variable-value assignments. Solving a CSP requires finding an assignment that satisfies all constraints. The CSP applet illustrates the Arc Consistency 3 (AC-3) algorithm for solving CSPs represented as networks of variable nodes and constraint arcs. AC-3 iteratively makes individual arcs consistent by removing variable domain values inconsistent with a given constraint, until all arcs have been considered and the network is consistent. Then, if there remains a variable with more than one domain value, a procedure called domain splitting is applied to that variable in order to split the CSP into disjoint cases so that AC-3 can recursively solve each case.

The CSP applet demonstrates the AC-3 algorithm dynamics via interactive visualizations on graphs using color and highlighting, and graphical state changes are reinforced through textual messages. The applet provides several mechanisms for the interactive execution of the AC-3 algorithm on a set of available CSPs. These mechanisms are accessible through the toolbar, or through direct manipulation of graph elements. The user can perform seven different actions: (1) Fine Step: use the fine step button to see how AC-3 goes through its three basic steps (selecting an arc, testing it for consistency, removing domain values to make the arc consistent); (2) Direct Arc Click: directly click on an arc to apply all these steps at once; (3) Auto AC:

automatically fine step on all arcs one by one using the auto arc consistency button; (4) Stop: pause auto arc consistency; (5) Domain Split: select a variable to split on, and specify a subset of its values for further application of AC-3 (see pop-up box in the bottom right of Fig. 1); (6) Backtrack: recover alternative sub-networks during domain splitting; (7) Reset: return the graph to its initial status.

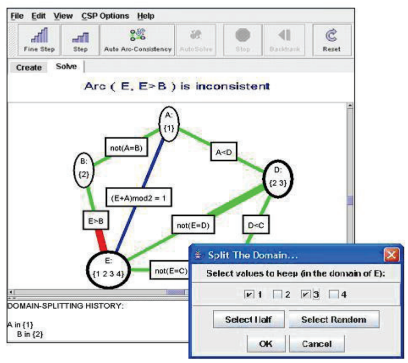


Fig. 1. CSP applet with example CSP problem

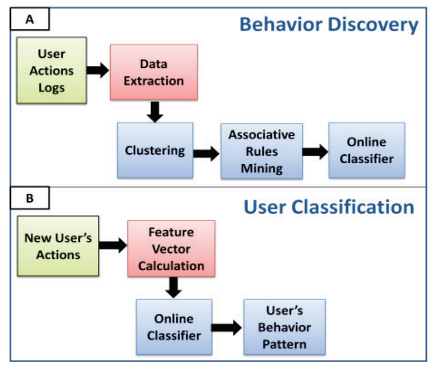


Fig. 2. General User Modeling Approach.

3 Mining Behavior Patterns

In this section we briefly describe the two main phases of our approach to building a classifier user model from interaction data first described in [7]: Behavior Discovery (Fig. 2A) and User Classification (Fig. 2B). In *Behavior Discovery*, raw unlabeled data from interaction logs is preprocessed into feature vectors representing individual users in terms of their interface actions. These vectors are the input to an unsupervised clustering algorithm (i.e., k-means with a modified initialization step, see [7]) that groups them according to their similarities. The resulting clusters represent users who interact similarly with the interface. These clusters are then analyzed to identify if/how they relate to learning. Afterwards, association rule mining is applied on each cluster to extract the common behavior patterns in the form of class association rules for each performance level. A Class Association rule is a rule in the form of X→ *c*, where X is a set of feature-value pairs and *c* is the predicted class label (i.e., the cluster) for the data points where X applies (see Table 1).

Our goal is to use these detected behaviors and information regarding their effectiveness as a guide for intelligent adaptive support during the interaction. Thus, in the *User Classification* phase (Fig. 2B), class association rules extracted in the Behavior Discovery phase are used to build an online classifier user model. This classifier is used to assess the performance of a new user based on her interactions.

In [7], we reported the result of applying our framework on the action logs collected from a study with 65 users using the CSP applet. For this dataset, the Behavior Discovery resulted in two clusters of users that achieved significantly different learning performance levels (high vs. low). We will refer to them as High Learning Gain (HLG) and Low Learning Gain (LLG) groups respectively. Also, the online classifier achieved an accuracy of over 80% in classifying new users as HLG or LLG by observing only the first 25 percent of their interactions.

In addition to assigning a label to the user, the user model also returns the observed rules that caused that classification decision. In [8], we described our proposed methodology for building an intervention mechanism based on the discovered behavior patterns which is briefly described in the next section.

4 Extracting Adaptation rules from Discovered Patterns

The class association rules generated in the Behavior Discovery phase represent the interaction behaviors of LLG and HLG. All of these rules are used in the classifier user model to determine the performance of a new user, and identify a set of behaviors that are either conducive or detrimental to learning. Ideally, one would want to design adaptive interventions that discourage all the detrimental behaviors, and encourage all the good ones. For instance, consider the following rule for the LLG:

Rule4: If Direct Arc Click frequency = Lowest **and** Direct Arc Click Pause Average = Lowest → Cluster LLG

This rule indicates that if the frequency of Direct Arc Click (DAC) action is lower than a threshold (the mechanism to set this threshold is described in [7]) and the average pause time between a DAC and the next action is also lower than a certain threshold then the user is considered a LLG. Here, we want to prevent this from happening and there are two possible interventions (***intervention items*** from now on) that can be delivered to address this rule: (1) Encouraging/enforcing the user to perform DAC more often; (2) Encouraging/enforcing the user to pause longer after DAC actions (possibly thinking about the DAC outcomes).

There may be several rules like the one above that are applicable at a given time. The number of rules, may pose a challenge considering factors such as the cost of implementation and effectiveness of the resulting intervention items, thus filtering the rules is necessary (see [8] for a detailed discussion). For each intervention item, we compute a score calculated as the sum of the weights of the rules which recommend that item within a given cluster (these weights indicate the importance of each rule in classifying a user [7]) and use this as an importance factor for that item. Then we apply a filtering strategy that keeps the most prominent behaviors and ignores the weaker ones while taking the diversity of the intervention items and their cost of implementation into account (see [8] for details). For our current study, we use 6 intervention items as selected by our filtering strategy, highlighted in Table 1.

Table 1. A selection of representative rules for HLG and LLG clusters in the CSP dataset

Rules for HLG cluster:
Rule1: Direct Arc Click frequency = Highest
Rule5: Domain Split frequency = Highest **and** Auto AC frequency = Lowest
└ **Rule8:** Domain Split frequency = Highest **and** Auto AC frequency = Lowest **and** Fine Step Pause Average = Highest **and** Reset frequency = Lowest
Rules for LLG cluster:
Rule1: Direct Arc Click Pause Average = Lowest
Rule3: Direct Arc Click frequency = Lowest

When delivering the implemented interventions to a user, there can be more than one rule satisfied at a certain time leading to multiple items being recommended to that user. If the items are semantically correlated (as determined by the system designer), there is an opportunity to combine two items into one hint. For instance, based on the light blue items in Table 1, a hint can recommend using Direct Arc Click instead of Auto AC, because Direct Arc Click is a finer-grained version of Auto AC, with added user involvement (semantically correlated items have the same color in Table 1). However, non-related items will need separate hint messages and we decided to deliver only one hint at a time to prevent users from possibly getting overwhelmed. Therefore, in each step we choose the intervention item with highest score, calculated similar to above but only for the satisfied rules that recommend that item.

Adaptation rules can be categorized into two main groups, (1) Preventive interventions that discourage bad behavior as detected by the rules for LLG cluster, e.g.: "IF user is classified as a LLG and is using Direct Arc Click very infrequently (less than a threshold), then give a hint to promote this action"; and (2) Prescriptive interventions that encourage the effective behaviors described by the rules for HLG cluster. In this case, we want these rules to be satisfied. This means that if a student labeled as LLG shows any behavior in contrast with these rules then the corresponding intervention will be delivered to her, e.g.: "IF the user label is LLG, then if *Direct Arc Click frequency* is lower than *x* and *Auto AC frequency* is higher than *y* then "prompt user to use Direct Arc Click instead of Auto AC".

The advantage of preventive interventions is that we already know these behaviors result in bad performance so we can confidently prevent users from following such patterns. Prescriptive interventions are less reliable because it is not clear if/how behaviors that were effective for some learners could be beneficial for others.

5 Designing adaptive interventions

There are different forms of adaptive interventions that can be used to implement a specific adaptation goal (in our case, helping students use and learn most effectively from the CSP applet). Similar to most of the educational environments that provide adaptive support, we provide explicit advice via textual hints, and provide this advice incrementally. However, our focus on the interface actions when extracting the user interaction behaviors enables us to make interface changes as another way of delivering interventions. Thus, we provide a first level of advice with a textual hint that suggests or discourages a target behavior, followed when needed by a textual hint that reiterates the same advice, accompanied by a related interface adaptation (e.g., highlighting or deactivating relevant interface items).

Delivering adaptive interventions also require deciding whether the interventions should be subtle or forceful. Subtle interventions are in the form of suggestions that can be easily ignored by the user (e.g. a text message shown in a hint box at the corner of the screen). Forceful interventions make the user follow the related advice by reducing or eliminating user's options for the next action (e.g. deactivating all the items on the toolbar to force the user to pause before taking next action).

The current adaptive version of the CSP applet uses the subtle approach. The main drawback of this approach is that the recommendations may not be attended to by users or the user might decide not to follow them. However, this approach has the very desirable advantage of being less intrusive than the forceful approach. Therefore, from a usability point of view, it makes sense to try and see whether subtle adaptive interventions can already significantly improve the effectiveness of the CSP applet.

The detailed procedure of delivering the subtle incremental interventions described above is as follows: (1) for each intervention there is a text message presented in format of a hint that appears in a hint box at the upper left corner of the applet's main panel (level-1 hint). The hint box will blink once, each time a new message is displayed. (2) After receiving the hint, the student is given a time window to change her behavior. (3) If after the time window, the preconditions for that intervention are still satisfied the intervention will be provided again. In this case in addition to a text message, corresponding interface element(s) for that intervention will be highlighted until the user chooses her next action (level-2 hint). Figure 3 shows a level-2 intervention suggesting a decrease in use of *Auto AC* vs. an increase in use of *Direct Arc Click*. In addition to a text message the arcs that can be clicked are also highlighted.

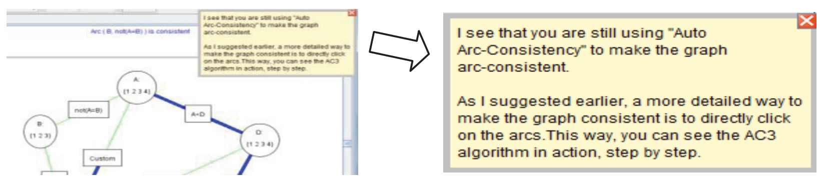


Fig. 3. A hint suggesting the use of Direct Arc Click action with the interface highlights (left); and the content of the hint box (right).

6 Evaluation

We ran a pilot study in a Wizard-of-Oz setting (i.e., experimenter would trigger the interventions based on a set adaptation rules) to evaluate the intervention mechanism described above for three factors: visibility, intrusiveness, and follow rate of the interventions. The data was collected from 6 computer science students. Each participant: (1) studied a textbook chapter on the AC-3 algorithm; (2) wrote a pre-test on the concepts covered in the chapter; (3) used the CSP applet to study two CSPs, while her gaze was tracked with a Tobii T120 eye-tracker; and (4) took a post-test analogous to the pre-test [9]. At the end of the experiment, a qualitative evaluation of interventions was done using a post-hoc questionnaire and a follow-up interview.

Figure 4 summarizes the opinion of our 6 participants about the text hint messages collected by the post-hoc questionnaire. The participants did not find the hint messages intrusive or annoying. They found the messages easy to notice and useful in the process of interaction. Moreover, most of the participants reported following the instructions provided in the hints. The rest of this section will present quantitative results derived from action logs and eye gaze data collected during the interaction.

Regarding visibility of the hints, out of 27 hints provided in total, 25 of them were attended to by the participants. One of two omitted hints was a level-1 hint given to participant 4 (P4), while she did not notice this hint, the subsequent level-2 of the same hint (with interface highlights) managed to grab her attention. The second case was a level-2 hint given to P6, where he decided not to follow a level-1 hint prior to this hint and was given a level 2 hint. In this case, the highlighting reminded him of the recommended action (Direct Arc Click) from the level-1 hint, thus he followed the hint without having to look at the hint box. These two cases, highlight the importance of the 2-level hinting strategy reinforced by interface changes.

Figure 5 illustrates the number of hints shown, attended to and followed by each participant. Out of 27 hints given, 20 were followed by the participants (74% follow rate). Students, who show many detrimental behaviors, will get more hints. Such students are the target group that we want to help learn better from their interaction with the CSP applet. Therefore, P2 and P4 are of especial interest. Both of these participants reported finding the interventions relevant and useful. However, P4 did not follow every hint, and generally only followed the recommendations when repeated in the form of a level-2 hint. This is reflected in her self assessment of how often she followed the hints as well (Table 2).

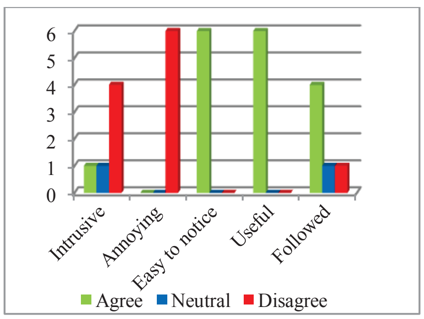


Fig. 4. Reception of the text hints by participants

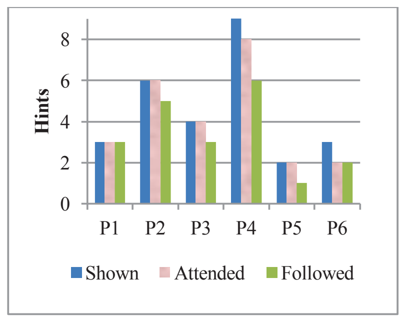


Fig. 5. Number of hints shown, attended and followed for each participant

We also analyzed the average reading time of the hint messages for each participant, overall and for the hints they dismissed/followed (Table 2). We can observe an individual element in reading time between participants which can be further investigated as a guide for user adaptive reaction time for hints. Another trend is that users who received more hints also spent less time reading them. This is expected as these users are the ones with sub-optimal interaction behaviours and this again shows the importance of the 2-level progressive hinting strategy which gets more intrusive the second time a hint is provided.

Table 2. Hint rate, self rated following of hints, and average reading time for each participant

	P1	P2	P3	P4	P5	P6
Followed Hints - Self-rated (1-5)	4	4	4	2	4	3
Avg. Reading Time (ms)	2814	1642	1547	925	2639.5	9460
Avg. Reading Time: Followed (ms)	2814	1530.6	1663	937.5	3464	8975
Avg. Reading Time: Dismissed (ms)	-	2199	1199	887.5	1815	9945
# Hints given	3	6	4	9	2	3

7 Conclusion and future work

In this paper, we presented the final step of the process for adding adaptive interventions to an OELE called AIspace CSP applet. This process started with mining behavior patterns in the form of association rules from a dataset of collected user interface actions [7]. Then, continued with extracting adaptation rules from the discovered behaviors [8]. The final step was to deliver the adaptive interventions defined based on the adaptation rules via an intervention mechanism. We identified the *form* and *forcefulness* of delivering the interventions as two aspects of this step and described our 2-level subtle method of delivering interventions using both text messages and interface changes. The very encouraging initial results of our pilot study regarding reception of the interventions by the users, shows a great potential for the Adaptive version of the CSP applet which provides personalized support. A second pilot study is scheduled to test the user model and the improvements made to the applet based on our findings in the first pilot study. We plan to run a full scale study afterwards.

References

1. Ploetzner, R., Lippitsch, S., Galmbacher, M., Heuer, D., Scherrer, S.: Students' difficulties in learning from dynamic visualisations and how they may be overcome. Computers in Human Behavior. 25, 56–65 (2009).
2. Shute, V.J.: A comparison of learning environments: All that glitters. Computers as cognitive tools. pp. 47–73. Hillsdale, NJ, England: Lawrence Erlbaum Associates, Inc (1993).
3. De Jong, T.: Technological Advances in Inquiry Learning. Science. 312, 532–533 (2006).
4. Amershi, S., Carenini, G., Conati, C., Mackworth, A.K., Poole, D.: Pedagogy and usability in interactive algorithm visualizations: Designing and evaluating CIspace. Interacting with Computers. 20, 64–96 (2008).
5. Cocea, M., Gutierrez-Santos, S., Magoulas, G.D.: Challenges for intelligent support in exploratory learning: the case of ShapeBuilder. Proceedings of the International Workshop on Intelligent Support for Exploratory Environments at ECTEL 2008. , Maastricht, The Netherlands (2008).
6. Romero, C., Ventura, S.: Educational Data Mining: A Review of the State of the Art. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews. 40, 601–618 (2010).
7. Kardan, S., Conati, C.: A Framework for Capturing Distinguishing User Interaction Behaviours in Novel Interfaces. In: Pechenizkiy, M., Calders, T., Conati, C., Ventura, S., Romero, C., and Stamper, J. (eds.) Proceedings of the 4th International Conference on Educational Data Mining. pp. 159–168. , Eindhoven, the Netherlands (2011).
8. Kardan, S., Conati, C.: Providing Adaptive Support in an Exploratory Learning Environment by Mining User Interaction Data. Proceedings of the 5th International Workshop on Intelligent Support for Exploratory Environments (ISEE 2012), in conjunction with the 11th International Conference on Intelligent Tutoring Systems (ITS 2012). , Chania - Greece (2012).
9. Kardan, S., Conati, C.: Exploring Gaze Data for Determining User Learning with an Interactive Simulation. In: Masthoff, J., Mobasher, B., Desmarais, M., and Nkambou, R. (eds.) User Modeling, Adaptation, and Personalization. pp. 126–138. Springer Berlin / Heidelberg (2012).

Adaptive Multi-Agent Architecture to Track Students' Self-Regulated Learning

Babak Khosravifar[1], Roger Azevedo[1], Reza Feyzi-Behnagh[1], Michelle Taub[1], Gautam Biswas[2], and John S. Kinnebrew[2]

[1]McGill University, The Laboratory for the Study of Metacognition and Advanced Learning Technologies, Montreal, Canada
[2]Vanderbilt University, The Teachable Agents Group at Vanderbilt University, Nashville, USA
(babak.khosravifar,roger.azevedo)@mcgill.ca,
(reza.feyzibehnagh,michelle.taub)@mail.mcgill.ca,
(gautam.biswas,john.s.kinnebrew)@vanderbilt.edu

Abstract. Intelligent Tutoring Systems (ITS) can be designed to improve learning and performance through Pedagogical Agents (PAs) that are designed to foster self-regulated learning through interactions and exchange of information with human learners. PAs are intelligent and follow rational behaviors, but to adaptively track students' progress, they need to be systematically and specifically designed. However, in order to follow a common goal, different self-regulatory systems have been designed that use PAs, but fail to provide an adaptive multi-agent architecture which provides such feature that agents adaptively track students' scaffolding. In this paper, we introduce a multi-agent framework designed for an agent-based ITS. We also define the agent architecture, multi-agent framework and communication mechanism.
Keyword. Pedagogical Agents, Self-Regulated Learning, Multi-Agent Systems, Agent Communication Mechanism.

1 Introduction

Increasing adaptivity is being devoted to frameworks involving intelligent components that receive (or search for) data and dynamically update their internal engine to efficiently acquire and integrate information. This adaptivity is becoming a crucial feature in ITSs that provide scaffolding for students to effectively self-regulate their learning. There are various ITSs [1–4, 6], which are used to conduct educational research. But in this paper, we only concentrate on agent-based ITSs [1, 3, 4, 6] where PAs continuously interact with students and objectively provide guidance to facilitate the process of learning and use of effective SRL processes. We concentrate on this category of ITSs because agents are intelligent components that could be equipped with adaptive applications and dynamically track student behaviour, based on the scaffolding they are receiving.

Current ITSs are not entirely adaptive to students' knowledge acquisition during learning in real-time. This may be because in most agent-based ITSs [1, 4, 6], agents are developed to interact with students to facilitate their navigation through parts of the system and provide adaptive scaffolds and feedback to facilitate their learning. This is done using rule-based (predefined) decision maker modules that pick a specific action, which can be either feedback to the students or some sort of communication with the system. The action selection mechanism has been thoroughly defined and enables agents to effectively react to students' progress based on predefined scenarios. In such ITSs, agents generally have a narrower focus on specific performance features/outcomes that illustrate acquisition of knowledge in the target domain.

To address the aforementioned adaptivity problem, PAs need to maintain decision making procedures [5] that continuously interact with the student (in the form of direct interaction and recording the data about that interaction) and dynamically analyze the collected data to update the scaffolding model that the agent builds as it assesses students' progress. By analyzing collected data, agents are able to better interact with students since they are aware of students' detailed work and progress in learning. In this paper, we focus on a multi-agent framework designed for an agent-based ITS that is being designed to analyze a much wider array of student behavior, activities, responses to agents, and performance in order to better understand many aspects of both students' understanding of domain knowledge and underlying self-regulatory abilities.

2 Multi-Agent Architecture

The proposed multi-agent architecture is a simulation environment designed to model and scaffold learners' SRL processes as they learn a biology topic. This environment is focused on further understanding of students' deployment of SRL processes by providing a computer-based learning environment with Pedagogical Agents (PAs) that model and track students' progress while learning complex science topics. In the proposed muti-agent architecture, there are three PAs that directly interact with students:

Peer agent, that interacts the most with the student and obtains basic information (like his/her knowledge level) from the student. In fact, the peer agent is the one that builds the student model and dynamically updates the model with respect to students' activities and deployment of SRL processes;

SRL agent, that tracks students' progress towards using effective SRL processes. This agent is in charge of guiding the student in accomplishing the learning goal and effectively finalizing the process of learning about the complex topic. The SRL agent also provides relative data (computed knowledge level) that influence the peer agent's further interaction with the student.

Science agent, that is in charge of helping and scaffolding the student to understand the science content. This agent informs the other two agents when the student is having difficulties with the content, choosing relevant page sequences, reading the content at an optimal time, and evaluating his/her goals.

The three introduced agents directly interact with students and are known by students as their interactive partners. These agents also interact with each

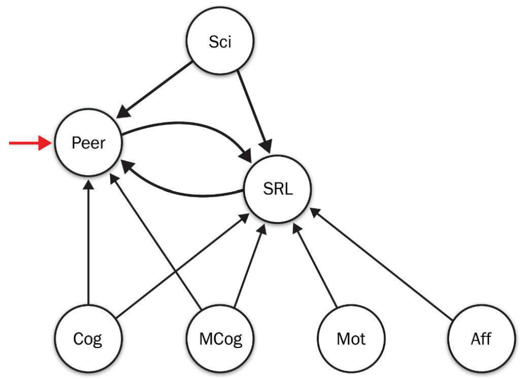


Fig. 1. Multi-agent framework.

other to better guide the student to accomplish the goal of learning about the complex science topic. To adaptively track and model students' scaffolds, there are various data types regarding students' use of SRL processes that need to be collected and analyzed in order to maintain adaptive scaffolding and provide effective guidance to the student. In the proposed framework, we assign four hidden agents, each of which are associated to a category that captures related data, analyses the data and provides relative reports in the form of messages to other involving agents. These massive data is categorized into four groups:

Cognitive agent, that provides details regarding students' learning-related parameters, including their content reading process, highlighting, note taking, and all other cognitive processes;

Metacognitive agent, that provides details regarding students' performance-related parameters, such as scores on various quizzes, accuracy of judgment of learning, and all other metacognitive processes;

Motivational agent, that provides details regarding students' task difficulty, attributions, self-efficacy;

Affective agent, that provides details regarding students' motivations while interacting with the system.

The whole architecture enhances the performance of data collection, and analyzes agents' decision making. Moreover, the multi-agent architecture provides modular functionalities that makes it simpler to test, analyze, and integrate in the system. Figure 1 illustrates the multi-agent architecture together with the involved agents. Hidden agents are rational intelligent components that are capable of analyzing data related to a specific architecture and a pre-defined logic. PAs are rational and are developed with goals related to educational purposes, such as, to optimize learning for students. The core of an agent architecture is its data processing engine that analyses the data that is collected from the surrounding environment and provides an action that best fits its goal-directed purpose. In the proposed architecture, PAs also run data analyses and react to the environment via a selected action by the student. We focus here on the obtained data that help (whether one of the three PAs or the four hidden agents) to analyze and better understand environmental changes, specifically students' decisions and actions.

In the proposed architecture, hidden agents continuously communicate to capture students' activities while interacting with the system and therefore provide accurate information, evidence, and reasoning to the three interactive agents who can then adaptively provide feedback and scaffolds to the students. In the proposed architecture, the main role of these four agents is to collect data regarding cognitive, metacognitive, motivational, and affective SRL processes. These massive data are continuously collected, analyzed and updated to adaptively track their learning progress and adaptations based on the scaffolding they are being provided with.

3 Conclusion

This paper introduces an adaptive multi-agent framework designed for intelligent tutoring systems. This framework could be used in agent-based learning environments where pedagogical agents coordinate with one another to facilitate SRL processes in learners [3]. The main objective is to enable PAs to effectively track students' progress while interacting with the system throughout the learning session. In future research, we intend to propose different mechanisms to develop adaptive multi-agent communication and decision making to represent an optimally efficient learning environment to facilitate the acquisition, internationalization, application, and transfer of self-regulatory processes.

References

1. Azevedo, R., Witherspoon, A., Chauncey, A., Burkett, C., and Fike, A. (2009). MetaTutor: A meta- cognitive tool for enhancing self-regulated learning. In R. Pirrone, R. Azevedo, G. Biswas, (eds.), Proceedings of the AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems (pp. 14-19). Menlo Park: Association for the Advancement of Artificial Intelligence (AAAI) Press.
2. Azevedo, R., Behnagh, R., Duffy, M., Harley, J., and Trevors, G. (2012). Metacognition and self-regulated learning in student-centered leaning environments. In D. Jonassen and S. Land (Eds.), Theoretical foundations of student-center learning environments (2nd ed.)(pp. 171-197). New York: Routledge.
3. Biswas, G., Jeong, H., Kinnebrew, J., Sulcer, B., and Roscoe, R. (2010). Measuring self-regulated learning skills through social interactions in a Teachable Agent Environment. Research and Practice in Technology-Enhanced Learning, 5(2), 123-152.
4. Graesser, A. C., and McNamara, D. S. (2010). Self-regulated learning in learning environments with pedagogical agents that interact in natural language. Educational Psychologist, 45, 234-244.
5. Ross, S., Chaib-draa B., and Pineau J. (2008). Bayesian reinforcement learning in continuous POMDPs with application to robot navigation. International Conference on Robotics and Automation (ICRA), pp. 2845-2851. Jochen Triesch: IEEE.
6. Woolf, B. (2009). Building intelligent interactive tutors: Student-centered strategies for revolutionizing E-Learning. San Francisco, CA: Morgan Kaufmann.

A Differential Temporal Interestingness Measure for Identifying the Learning Behavior Effects of Scaffolding

John S. Kinnebrew, Daniel L.C. Mack, and Gautam Biswas

Department of EECS and ISIS, Vanderbilt University, Nashville TN 37212, USA,
john.s.kinnebrew@vanderbilt.edu

Abstract. Effective design and improvement of scaffolding in complex and open-ended learning environments, requires the ability to assess the effectiveness of a variety of scaffolding options, not only in terms of overall performance and learning, but also in terms of more subtle effects on students' behavior and understanding. In this paper, we present a novel data mining technique that aids the analysis of scaffolding and students' learning behaviors by identifying activity patterns that distinguish groups of students (e.g., groups that received different scaffolding and feedback during an extended, complex learning activities) by differences in both total behavior pattern usage and evolution of pattern usage over time. We demonstrate the utility of this technique through application to student activity data from a recent experiment with the Betty's Brain learning environment and four different scaffolding conditions.

Keywords: learning behaviors, interestingness measure, sequence mining, information gain

1 Introduction

In order to more effectively teach and promote skills required in the modern world of near-ubiquitous computing and internet connectivity, computer-based learning environments have become more complex and open-ended. This complexity also drives a need for dynamic and adaptive scaffolding that can support students in understanding how to employ and learn with these environments and tools. However, in order to effectively design and improve such scaffolding, we must first be able to assess the effectiveness of a variety of scaffolding options, not only in terms of overall performance and learning, but also in terms of more subtle effects on students' behavior and understanding. In this paper, we present a novel data mining technique that aids the analysis of how students' learning behaviors and strategies are employed with differing frequency over the course of learning or problem-solving activities as the result of different scaffolds and feedback that can be provided in a learning environment.

Identifying sequential patterns in learning activity data can be useful for discovering and understanding student learning behaviors. Researchers have applied sequence mining techniques to a variety of educational data in order to better understand learning. The primary sequential pattern mining task is to discover sequential patterns of items that are found in many of the sequences in a given dataset [1, 2]. For example, Perera *et al.* ([3]) use sequential pattern mining to provide mirroring and feedback tools to support effective teamwork among students collaborating on software development using an open source professional development environment called TRAC. Other researchers have also employed sequential pattern mining to identify differences among student groups or generate student models to customize learning to individual students [4–6].

Once these behavior patterns are mined, researchers must interpret and analyze the resulting patterns to identify a relevant subset of important patterns that provide a basis for generating actionable insights about how students learn, solve problems, and interact with the environment. Researchers have developed a variety of measures to utilize properties other than the default of pattern frequency to rank mined patterns [7]. These measures are often referred to as "interestingness measures" and have been applied data mining tasks like sequence mining and association rule mining [8]. To better analyze student learning and behavior, interestingness measures have been used for tasks such as ranking mined association rules (e.g., [9]).

Investigation of the frequency with which a pattern occurs over time can reveal additional information for pattern interpretation and may help identify more important patterns, which occur only at certain times or become more/less frequent, rather than patterns with frequent, but uniform, occurrence over time. In this paper, we present a novel approach, combining sequence mining and an information-theoretic measure for ranking behavior patterns that distinguish groups of sequences (e.g., groups of students in different experimental conditions) by differences in both total pattern usage and the evolution of pattern usage over time. To effectively analyze these patterns and quickly identify trends in the evolution of pattern usage, we employ a related visualization in the form of heat maps.

2 Identifying Interesting Differences in Pattern Usage

In this section, we present the Differential Temporal Interestingness of Patterns in Sequences (D-TIPS) technique, and its novel interestingness measure, for identifying and visualizing patterns that are employed differentially over time among groups of students (e.g., groups that receive different scaffolding in an open-ended learning environment). The first step in analyzing learning activity sequences is to define and extract the actions that make up those sequences from interaction traces logged by the environment. The definition of actions in these sequences for Betty's Brain data is discussed further in Section 3. Given a set of sequences corresponding to the series of actions performed by each student, the D-TIPS technique consists of four primary steps:

1. Generate candidate patterns that are common to the majority of students in at least one group by combining the sets of patterns identified through ap-

plying sequential pattern mining separately to each group's learning activity sequences (with a frequency threshold of 50%).

2. Calculate a temporal footprint for each candidate pattern by mapping it back to locations where it occurs in the activity sequences. Specifically, each sequence is divided into n consecutive slices, such that each contains $\frac{100}{n}\%$ of the student's actions in the full sequence, where n is the chosen number of bins defining the temporal granularity of the comparisons. Corresponding slices for a group (e.g., the first slice from each sequence in the group, the second slice from each, and so on) are then grouped into bins and each action in the slices is marked to indicate whether or not it is the beginning of a pattern match in its original sequence. This set of binned and marked actions defines the temporal footprint of the pattern for the group.
3. Provide a ranking of the candidate patterns using an information-theoretic interestingness measure (described in more detail below) applied to the temporal footprint of each pattern.
4. For the highly-ranked patterns, visualize their temporal footprints using heat maps to identify differences in usage trends and spikes across student groups. Specifically, we employ a two-dimensional heat map where the y-axis is student group and the x-axis is time discretized by temporal bin. In a single row (i.e., for a specific student group), each cell's count is the percentage of total pattern occurrence (with respect to the student group) within the corresponding temporal bin. The use of *percentages* of pattern occurrence allows analysis of temporal variation normalized by the total frequency of the pattern in the group, which will tend to highlight different temporal trends in pattern usage across groups, even when total pattern occurrence differs significantly among groups.

In order to identify more interesting patterns by their difference in temporal usage across groups in step 3, the D-TIPS interestingness measure applies information gain (IG) with respect to pattern occurrence across the groups in each of the n corresponding bins of their temporal footprints. Information gain is defined as the difference in expected information entropy [10] between one state and another state where some additional information is known (e.g., the difference between a set of data points considered as a homogeneous group versus one split into multiple groups based on the value of some other feature or attribute). Information entropy is the amount of expected uncertainty found in a random variable, X, whose value can be called the *class* of the data point. IG when used in classifiers, such as decision trees, is applied to a dataset where each data point has multiple features in addition to its class. The IG of a given feature is then the reduction in expected uncertainty about the correct class of a data point when its feature value is known, or conversely the increase in information about the class of the data point. IG is calculated as the difference between the information entropy of the data without knowledge of the feature values and the conditional information entropy when the feature values are known.

Information gain is leveraged in classifiers to determine which features are most discriminatory because they provide the least amount of uncertainty among classes in the data. D-TIPS applies information gain to determine which patterns are the most interesting because knowledge of their occurrence and temporal location provides the least amount of uncertainty among the student groups. In D-TIPS, each action/data-point's class is its group, and the feature of each data point, for a given pattern, is the combination of whether the action begins an occurrence of the pattern *and* the number of the bin in which the action occurred. This information-theoretic definition of the D-TIPS measure provides two important properties: 1) given two patterns with the same total occurrences for corresponding groups, the pattern with the greater discrimination of groups by *differences in temporal location/bin among groups* will have a higher rank, and 2) given two patterns with the same relative temporal behaviors (i.e., the same proportion of total pattern occurrence in each bin) for corresponding groups, the pattern with the greater discrimination of groups by *differences in total occurrence among groups* will have a higher rank.

Therefore, the D-TIPS measure provides a way of recognizing differences among groups both by total pattern occurrence and by temporal behavior (e.g., decreasing usage versus increasing usage, or spikes in different bins). Further, when the same differences across groups occur for two patterns, the pattern with higher overall frequency will have the higher rank. Thus, D-TIPS tends to emphasize patterns with large relative differences among groups (by total occurrence and/or temporal behavior) even when they are not especially frequent in the overall dataset, while also emphasizing patterns with more moderate differences among groups when the frequency of the pattern in the overall dataset is high. Conversely, D-TIPS tends to deemphasize patterns that are homogeneous across groups (by both relative occurrence and temporal behavior) or that are especially rare in all groups.

3 Betty's Brain Data

The data we employ in the analysis in Section 4 consists of student interaction trace from the Betty's Brain [11] learning environment. In Betty's Brain, students read about a science process and teach a virtual agent about it by building a causal map. They are supported in this process by a mentor agent, who provides feedback and support for their learning activities. The data analyzed here was obtained in a recent study with 68 7^{th}-grade students taught by the same teacher in a middle Tennessee school. At the beginning of the study, students were introduced to the science topic (global climate change) during regular classroom instruction, provided an overview of causal relations and concept maps, and given hands-on training with the system. For the next four 60-minute class periods, students taught their agent about climate change and received feedback on content and learning strategies from the mentor agent.

The study tested the effectiveness of two support modules designed to scaffold students' understanding of cognitive and metacognitive processes important for success in Betty's Brain (details provided in [12]). The *knowledge construction* (KC) support module scaffolded students' understanding and suggested strategies on how to construct knowledge by identifying causal relations in the resources, and the *monitoring* (Mon) support module scaffolded students understanding and suggested strategies on how to monitor Betty's progress by using the quiz results to identify correct and incorrect causal links on Betty's map. Participants were divided into a control and three treatment groups. The knowledge construction (KC) group used a version of Betty's Brain that included the KC support module and a causal link tutorial that they could access at any time and were prompted to enter when the mentor determined they were having difficulty identifying causal links in the resources. The monitoring (Mon) group used a version of Betty's Brain that included the Mon support module and a tutorial about employing link annotations to keep track of links shown to be correct by quizzes. The full (Full) group used a version of Betty's Brain that included both support modules and tutorials. Finally, the control (Con) group used a version that included neither the tutorials nor the support modules.

In Betty's Brain, the students' learning and teaching tasks were organized around seven activities: (1) reading resource pages to gain information, (2) adding or removing causal links in the map to organize and teach causal information to Betty, (3) querying Betty to determine her understanding of the domain based on the causal map, (4) having Betty take quizzes that are generated and graded by the mentor to assess her current understanding and the correctness of links in the map, (5) asking Betty for explanations of which links she used to answer questions on the quiz or queries, (6) taking notes for later reference, and (7) annotating links to keep track of their correctness determined by quizzes and reading. Actions were further distinguished by context details, which for this analysis were the correctness of a link being edited and whether an action involved the same subtopic of the domain as at least one of the previous two actions. The definition of actions in Betty's Brain learning activity sequences are discussed further in [13].

4 Results

To illustrate and characterize the performance of the D-TIPS technique on educational data, we present selected results from its application to student learning activity data in the Betty's Brain classroom study described in Section 3. The D-TIPS analysis identified 560 activity patterns that occurred in at least half of the students in one or more of the four experimental conditions. Given the limited number of students in each condition, we chose to bin pattern occurrence values into fifths of the activity sequences for a broad analysis of their usage evolution over time. Table 1 presents 3 of the top 30 most differentially-interesting patterns identified by D-TIPS across the four scaffolding conditions. For comparison, the average occurrences per student and ranking by that value is also presented. Over half (18) of the 30 analyzed D-TIPS patterns had a rank past 50th by occurrence, with 13 of them ranking beyond 100th, indicating that they would be unlikely to have been considered without D-TIPS.

Table 1. Selected Patterns with D-TIPS and Occurrence Rankings

Pattern	D-TIPS Rank	Occurrence Rank	Avg Occurrence
[Quiz]	3	2	21.8
[Read] → [Note]	18	100	1.7
[Read] → [Read] → [Remove Link$^-$]	29	137	1.4

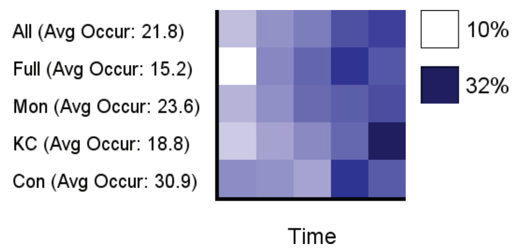


Fig. 1. [Quiz]

The first pattern in Table 1 illustrates a single action pattern that was ranked very high by both D-TIPS and overall occurrence. While individual student actions are often less interesting than longer patterns, they are still important to consider, especially when they also illustrate a tendency to be employed differentially across groups and over time. Figure 1 shows that all groups tended to use quizzes more frequently later in their work on the system. Since students' causal maps grew over time, monitoring and correction of the maps were more important later in their learning activities. There were some differences in usage trends over time among the different conditions, such as the steeper increasing trend for the KC and Full groups than the Monitoring group and the earlier peak in usage for the Full and Control groups. However, the overall occurrence by conditions differed markedly, with the Control group performing far more quiz actions than the others, and the Monitoring group performing more quiz actions than the KC and Full groups. While the Monitoring group's use of the quiz was expected to be high due to the focused monitoring support that relied heavily on the quiz, it is surprising that the Control group had the highest quiz usage. This might indicate that without either KC or monitoring support, the Control group struggled more and fell back on guessing and checking (with the quiz) strategies.

Figure 2 illustrates a knowledge construction behavior of reading and taking notes that was ranked highly by D-TIPS. Another difference among the groups, which added to the interestingness of this pattern under the D-TIPS analysis, is that the Control group tended to perform reading followed by note-taking primarily in the last fifth of their activities, as opposed to the first two fifths for the other groups. However, further analysis of the data attributed this primarily to only two of the Control group students, although the reason for this aberration is still unclear.

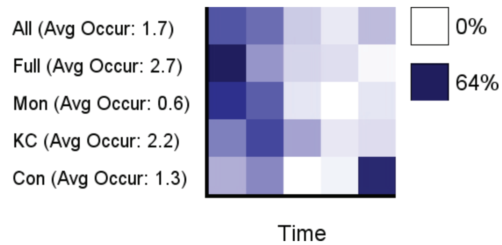


Fig. 2. [Read] → [Note]

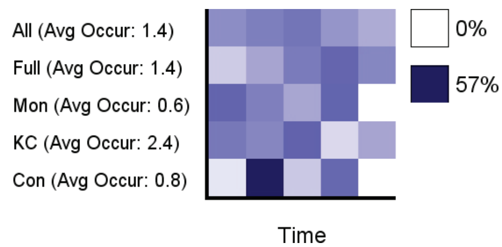


Fig. 3. [Read] → [Read] → [Remove Link^{-}]

The pattern illustrated in Figure 3 involves a sequence of (two) reading actions followed by removing an incorrect link. While there was no consistent temporal trend in the usage of this pattern, the Monitoring and Control groups exhibited this pattern less than once per student, while the KC group averaged 2.4 times per student. Although ranked lower by D-TIPS at 45th, the sub-pattern of a single read action followed by removing an incorrect link illustrates the same differences. This suggests that students with the KC feedback, relied more heavily on reading to identify incorrect links than either the Control and Monitoring groups, possibly because the Control group struggled more in general and the support in the Monitoring group focused students more on the use of quizzes to identify incorrect links.

5 Conclusion

While identification of high-frequency patterns is undoubtedly useful, finding patterns that have differing usage over time across a set of student groups is also important for analyzing the effects of scaffolding. In this paper, we presented the D-TIPS technique, which identifies patterns that differ in their usage among student groups by either total (group) occurrence or temporal behavior, even when they are not especially frequent in the overall dataset. Results from the use of this technique to mine Betty's Brain data illustrated the potential benefits and helped characterize differences between D-TIPS and a baseline occurrence ranking. D-TIPS identified patterns that illustrated potentially important differences in learning behavior among different scaffolding conditions that would have probably been overlooked by considering only pattern frequency. Future work will include autonomous identification of an effective number of bins for splitting a given set of activity sequences, as well as methods to individually characterize student groups by the patterns identified in D-TIPS.

6 Acknowledgments

This work has been supported by NSF-IIS Award #0904387 and IES Award #R305A120186.

References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proceedings of the Eleventh IEEE International Conference on Data Engineering (ICDE). (1995) 3–14
2. Zaki, M.: Sequence mining in categorical domains: incorporating constraints. In: Proceedings of the ninth international conference on Information and knowledge management, ACM (2000) 422–429
3. Perera, D., Kay, J., Koprinska, I., Yacef, K., Zaïane, O.: Clustering and sequential pattern mining of online collaborative learning data. IEEE Transactions on Knowledge and Data Engineering **21**(6) (2009) 759–772
4. Amershi, S., Conati, C.: Combining unsupervised and supervised classification to build user models for exploratory learning environments. Journal of Educational Data Mining **1**(1) (2009) 18–71
5. Martinez, R., Yacef, K., Kay, J., Al-Qaraghuli, A., Kharrufa, A.: Analysing frequent sequential patterns of collaborative learning activity around an interactive tabletop. In: Proceedings of the Fourth International Conference on Educational Data Mining, Eindhoven, Netherlands (2011)
6. Su, J.M., Tseng, S.S., Wang, W., Weng, J.F., Yang, J., Tsai, W.N.: Learning portfolio analysis and mining for scorm compliant environment. Journal of Educational Technology and Society **9**(1) (2006) 262–275
7. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: A survey. ACM Computing Surveys (CSUR) **38**(3) (2006) 9
8. Zhang, Y., Zhang, L., Nie, G., Shi, Y.: A survey of interestingness measures for association rules. In: Business Intelligence and Financial Engineering, 2009. BIFE'09. International Conference on, IEEE (2009) 460–463
9. Merceron, A., Yacef, K.: Interestingness measures for association rules in educational data. Educational Data Mining 2008 (2008) 57
10. Renyi, A.: On measures of entropy and information. In: Fourth Berkeley Symposium on Mathematical Statistics and Probability. (1961) 547–561
11. Biswas, G., Leelawong, K., Schwartz, D., Vye, N., Vanderbilt, T.: Learning by teaching: A new agent paradigm for educational software. Applied Artificial Intelligence **19**(3) (2005) 363–392
12. Biswas, G., Kinnebrew, J.S., Segedy, J.R.: Analyzing students' metacognitive strategies in open-ended learning environments. In: Proceedings of the 35th annual meeting of the Cognitive Science Society, Berlin, Germany (August 2013)
13. Kinnebrew, J.S., Loretz, K.M., Biswas, G.: A contextualized, differential sequence mining method to derive students' learning behavior patterns. Journal of Educational Data Mining (In Press, 2013)

Process and Outcome Benefits for Orienting Students to Analyze and Reflect on Available Data in Productive Failure Activities

Ido Roll, Natasha G. Holmes, James Day, Anthony H.K. Park, and D.A. Bonn

University of British Columbia, 6224 Agricultural Road, Vancouver, BC V6T 1Z1, Canada
{ido,nholmes,jday,bonn}@phas.ubc.ca

Abstract. Invention activities are Productive Failure activities in which students attempt to invent methods that capture deep properties of given data before being taught expert solutions. The current study evaluates the effect of scaffolding on the invention processes and outcomes, given that students are not expected to succeed in their inquiry and that all students receive subsequent instruction. Two Invention activities related to data analysis concepts were given to 130 undergraduate students in a first-year physics lab course using an interactive learning environment. Students in the Guided Invention condition were given prompts to analyze given data prior to inventing and reflect on their methods after inventing them. These students outperformed Unguided Invention students on delayed measures of transfer, but not on measures of conceptual or procedural knowledge. In addition, Guided Invention students were more likely to invent multiple methods, suggesting that they used better self-regulated learning strategies.

Keywords: Invention activities, productive failure, scaffolding, interactive learning environments, transfer.

1 Introduction

Invention activities are activities in which students generate solutions to novel problems prior to receiving instruction on the same topics. For example, students may be asked to generate methods that capture the variability of given data sets prior to being taught about mean deviation [1-3]. Invention activities facilitate Productive Failure in that students commonly fail to generate valid methods in these activities [4-5]. For example, students may use range or count the number of different values as a measure of variability, ignoring distribution and number of data points. However, the failure is often productive as students learn from the subsequent instruction and practice better than students who receive only instruction and practice, controlling for overall time on task [1,3-6].

Unlike other forms of Productive Failure, in Invention activities students are given carefully designed sets of data, called *contrasting cases*, to invent mathematical methods that capture deep properties of data [7-8]. For example, the contrasting cases in Figure 1 are given to students when asked to create a method for calculating a weighted average. The contrast between Carpenters A and C helps students notice and encode the roles of spread and magnitude. The contrast between A and D helps students notice the role of sample size.

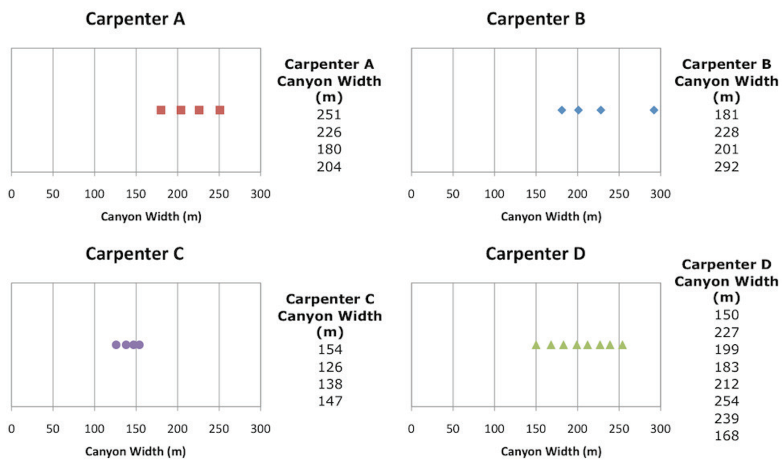


Figure 1 Contrasting cases emphasize the roles of magnitude, distribution, and sample-size in determining weighted average.

The invention process resembles an inquiry process in that students attempt to discover the underlying structure of data [9]. Thus, in the absence of additional support, it is of no surprise that students rarely invent valid methods. However, as described earlier, the invention process improves subsequent learning even in the absence of successful invention [1,2,6]. This raises an interesting question, which we address in this paper: Should the invention process be supported? One hypothesis suggests that supporting invention may lead to improved learning, as students may invent better methods. However, an alternative hypothesis suggests that failure is necessary for learning [10]. Thus, supporting students during their invention process may, in fact, hinder learning.

Scaffolding Invention Activities

One common form of support is scaffolding [11]. Specifically, scaffolding the inquiry process was shown to improve learning in discovery learning [12-13]. Within the context of Invention activities, similar scaffolding was shown to improve the invention process and its outcomes [3]. Within the scope of this study, we chose to focus on scaffolding two key phases that bracket the invention process: orientation and reflection.

Orientation. Invention Activities constrain the inquiry process by offering students contrasting cases to work with. However, simply having the contrasting cases may not be enough. We have previously found that many students working with Invention activities do not engage with the available contrasts when developing their methods [3]. Thus, following a prescriptive cognitive task analysis, we developed and validated prompts that help students orient themselves to the given data. This is done by instructing students to make pairwise comparisons between the contrasting cases with regard to the target concept. For example, students would be asked to compare carpenters A and D in figure 1 to determine which one did a better job of measuring the width of a bridge, see Figure 2. Since the two cases have roughly the same average and spread, students are confronted with the issue of sample size and need to determine whether and how the number of measurements may factor into the problem.

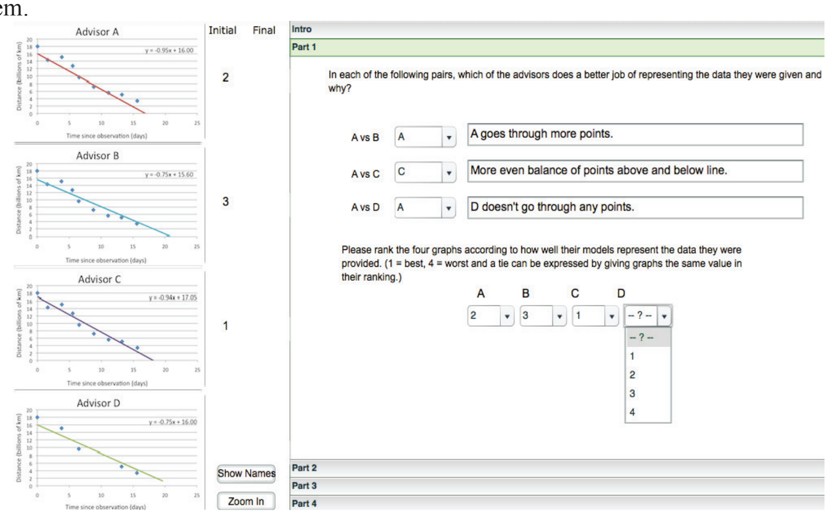


Figure 2. Ranking pairwise contrasts in the orientation scaffold.

Reflecting on the invented method. A second process that we chose to focus on is evaluation and reflection. In addition to being a key process in the scientific toolbox, the process of evaluation is beneficial, as it requires students to self-explain their correct or incorrect reasoning. In the context of Invention activities, once students develop their methods, the scaffolding asks them to explain how their invented methods take into account what they have learned during the pairwise comparisons. Students then apply their invented method to the contrasting cases, and then are asked to evaluate their method by comparing these results to their qualitative rankings as identified by them intuitively in the orientation phase.

Scaffolding students' orientation and reflection processes was found to improve students' invention behaviours and their invented methods on paper [3]. However, we are yet to evaluate the effect of the scaffolding on students' learning gains. The current study evaluates the effect of scaffolding during Invention activities on learning in two ways. First, we evaluate whether scaffolding improves the invention process itself. Given that evaluation and iteration are important inquiry skills, and that multiple invented methods are often associated with better learning in Productive Failure tasks [5], we evaluate the invention process by measuring the likelihood that groups invent more than a single solution. Second, we evaluate the effect of scaffolding on learning outcomes from the overall invention-instruction-practice process. We do so by comparing pre-to-post gains. Notably, these scaffold are static, unlike the view of scaffolding as an adaptive, negotiated process [14]. Understand when students require scaffolding in Productive Failure, and how to detect that using a student model, is outside the scope of the current work.

Method

We compared the Invention activities with and without scaffolding using a pre-to-post design. The *in-vivo* study took place in a first-year physics laboratory course at the University of British Columbia. 130 first-year students from four sections of the course participated in the study. The study was spread across a four-month term with the pre-test and two Invention activities given in three subsequent weeks at the beginning of the term. The final post-test was delivered at the end of the term, roughly two months after students had finished the second invention activity.

Students were randomly assigned to two groups, and different groups were assembled for the two activities. Students in the Unguided Invention (UI) condition worked with a convention invention activity, as defined in [1.2] (n = 65). Students in the Guided invention (GI) condition received the additional scaffolding, as described below (n = 65). Students were given approximately 30 minutes to work on the Invention activities. Each activity was followed by a short lecture on the target domain from the course instructor, which included a group discussion to direct students' attention to the important features of the data. Following the direct instruction, students worked on scientific experiments for roughly two more hours. These experiments provided opportunities for students to practice applying the expert solution from the Invention activities. Topics from the Invention activities were revisited or built on in subsequent weeks.

All students worked on the Invention activities using a dedicated interactive learning environment, the Invention Support Environment (ISE) [15]. Figure 3 shows the interface of ISE for the second activity used in this study, which focuses on evaluating goodness of fit for linear trendlines. The majority of the screen estates are dedicated to an accordion that breaks down the invention process:

- Introduction: background story and task
- Part 1: orientation. I this phase students analyze the contrasting cases qualitatively (available to GI students only).
- Part 2: generation. In this phase students invent a mathematical method to capture the deep property of the data. This is done using an equation editor (shown in Figure 3).
- Part 3: Students were guided to apply their method using a calculator or a spreadsheet software (e.g., MS Excel), and report back their values.
- Part 4: Students were asked to evaluate their methods based on their qualitative ranking (GI condition only).

The left side of the screen presents the contrasting cases to students. These stay available throughout the process. Students can zoom in on the contrasting cases and see the raw data by clicking on the Zoom In button. The centre of the screen shows students their initial and final ranking, when these are available (GI condition only).

The ISE is a skeleton that can deliver a variety of invention activities that share the same structure. It is used regularly by instructors in this course to deliver roughly 5-6 activities per term. A current version of ISE also includes instruction and opportunities for practice within the environment. Authoring new problems in ISE requires designers to give the text and data, but not to author new behaviours, as these are already built into ISE. ISE was built using the Cognitive Tutor Authoring Tools (CTAT) [16].

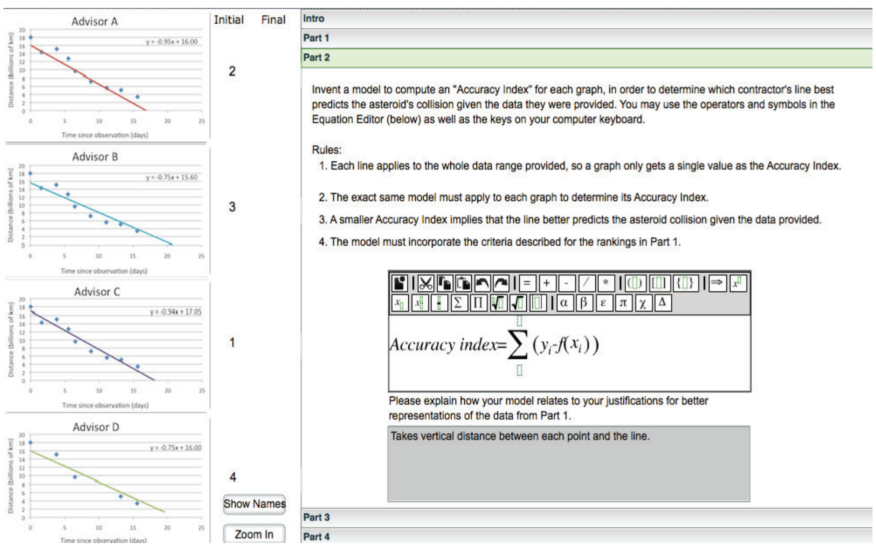


Figure 3: The Invention Support Environment

The two conditions differed with regard to support that students received before and after inventing their methods. The scaffolding that was given to students in the GI condition was modeled after the paper scaffoldings that were used in [3]. These scaffolding were designed to promote expert scientific behaviours that were identified in a prescriptive cognitive task analysis using similar invention activities:

The goal of the Orientation prompts was to get students familiar with the data prior to beginning to invent. Students were asked to compare pairs of contrasting cases and rank these according to the target feature. Students were then asked to briefly explain each of their rankings.

To encourage students to reflect on their invented methods, students were explicitly asked to self-explain their invented methods, referring back to their pairwise rankings. In addition, students were explicitly asked to evaluate their methods by comparing the results of their calculated values with their initial ranking during the orientation.

It should be noted that while the UI group did not have explicit prompts to perform these particular steps, they still had the opportunity to engage in them spontaneously. For example, the implementation process often leads naturally to reflection, as students recognize the shortcomings of their formulas, especially if the students spontaneously analyzed the contrasting cases first. Thus, the main difference between the conditions is the explicit prompting to carry out and reflect on each of the key stages. Table 1 summarizes the differences between conditions. Snapshots of the entire process can be found in Appendix B.

The pre- and post- tests included three types of questions on both invention topics. Procedural items asked students to calculate numeric answers by applying the formulas. Conceptual items asked students to apply the concepts without calculation to demonstrate understanding of the basic principles of the domains. Transfer items provided students with equations that were deliberately varied from the domain formulas and asked students to evaluate whether the formulas were reasonable ways to accomplish the same task. This requires a deep understanding of the deep features of the domain and their mathematical expressions in the equations [17]. Each type of assessment had two items, one on each topic.

Results

There was no effect for condition on pre-test: $t(127) = 0.18$, $p = 0.856$ (see Table 1). A paired t-test found significant learning from pre-test ($M = 0.47$, $SD = 0.24$) to post-test ($M = 0.61$, $SD = 21$) on items that were shared by both tests: $t(129) = 5.75$; $p <$ .0001.

Overall, 111 pairs of students worked on the two activities (56 pairs on the first activity and 55 pairs on the second). A logistic regression model found that groups in the GI condition were significantly more likely to create multiple methods, controlling for task, GI = 51% UI = 38%; $B = 1.13$, $SE(B) = 0.56$ $e^B = 3.091$, Z = 4.02, $p = 0.045$. The odds ratio (e^B) suggest that the odds to invent multiple methods is three times as high for GI students compared with UI students.

Table 1. Mean (SD) pre- and post-test scores on procedural, conceptual, and transfer items.

Item Type	Unguided Invention	Guided Invention
Pretest:	28% (31%)	33% (32%)
Posttest:		
- Procedural	46% (31%)	47% (28%)
- Conceptual	75% (28%)	74% (32%)
- Transfer	21% (29%)	33% (35%) *

* p < 0.05

An ANCOVA evaluating the effect of scaffolding on learning found no significant effect for condition on procedural, $F(2,127) = 0.02$, $p = 0.882$; or conceptual knowledge, $F(2,127) = 0.09$, $p = 0.761$. However, condition had a significant effect on transfer items, GI: M = 0.33, SD = 0.35; UI: M = 0.21, SD = 0.29: $F(2,127) = 4.81$; $p = 0.030$.

Discussion and Summary

The results presented above show that adding scaffolding to the invention process led to a higher rate of multiple methods during the invention process and to increased gains on a measure of transfer two months after the initial learning period. The scaffold had no effect on procedural and conceptual items. This is not surprising since the invention process itself usually has no benefits for these items compared with direct instruction and practice alone [1,2,17]. Thus, modifying the invention process similarly has no effect on performance on these items.

One key question to be answered is how the scaffolding resulted in the observed improvements. One likely answer suggests a two-fold process. By requiring students to compare pairs of contrasting cases, students notice more features, thus gaining a fuller understanding of the target domain. Using reflection prompts, the scaffolding improves students' meta-knowledge in that it highlights what is known (features) versus what is yet to be learned (the integrated method). Thus, orientation and reflection prompts help students obtain a fuller understanding of the domain, but not necessarily of any specific method. This may explain the observed effect on transfer, but not other, items.

The study further demonstrates that Productive Failure works not simply because support should be delayed. Instead, it is the transmission of domain knowledge that should be withheld, while other forms of support may be beneficial for learning even using the Productive Failure paradigm [6].

The study has several limitations. Most notably, due to the dynamic allocation of students to groups, we did not directly evaluate the relationship between quality of invention and quality of learning. Future work will have to address this issue, as well as focus on topics other than data analysis.

Notably, adding guidance during Invention activities helps learning even though students commonly fail to invent the expert solutions. Thus, not only that the failure to invent is, indeed, productive, but also, some failures are more productive than others. This study demonstrates how engaging students with good scientific practices helps them achieve a more productive failure.

Acknowledgements

This work was supported by the Pittsburgh Science of Learning Center, which is funded by the National Science Foundation, award number (#SBE-0836012), and by the University of British Columbia through the Carl Wieman Science Education Initiative.

References

1. Roll, I., Aleven, V., & Koedinger, K. R. (2009). Helping students know 'further'-increasing the flexibility of students' knowledge using symbolic invention tasks. In Proceedings of the 31st annual conference of the cognitive science society (pp. 1169-74).
2. Schwartz, D. L., & Martin, T. (2004). Inventing to prepare for future learning: the hidden efficiency of encouraging original student production in statistics instruction. Cognition and Instruction, 22(2), 129–184. Doi:10.1207/s1532690xci2202_1
3. Roll, I., Holmes, N., Day, J., & Bonn, D. (2012). Evaluating metacognitive scaffolding in guided Invention activities. *Instructional Science*, *40*(4), 691–710. doi:10.1007/s11251-012-9208-7
4. Kapur, M. (2008). Productive failure. *Cognition and Instruction*, *26*(3), 379–424. doi:10.1080/07370000802212669
5. Kapur, M., & Bielaczyc, K. (2012). Designing for productive failure. Journal of the Learning Sciences, 21(1), 45–83. doi:10.1080/10508406.2011.591717
6. Westermann, K., & Rummel, N. (2012). Delaying instruction: evidence from a study in a university relearning setting. *Instructional Science*, *40*(4), 673–689. doi:10.1007/s11251-012-9207-8
7. Schwartz, D. L., Chase, C. C., Oppezzo, M. A., & Chin, D. B. (2011). Practicing versus inventing with contrasting cases: The effects of telling first on learning and transfer. *Journal of Educational Psychology*, *103*(4), 759–775.
8. Roll, I., Aleven, V., & Koedinger, K. R. (2010). The invention lab: Using a hybrid of model tracing and constraint- based modeling to offer intelligent support in inquiry environments. In V. Aleven, J. Kay, & J. Mostow (Eds.), Proceedings of the 10th International Conference on Intelligent Tutoring Systems (pp. 115-24). Berlin: Springer Verlag.
9. de Jong, T., & van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research*, *68*(2), 179–201. doi:10.3102/00346543068002179
10. VanLehn, K. (1988). *Toward a theory of impasse-driven learning* (pp. 19-41). Springer US.
11. Koedinger, K. R., & Aleven, V. (2007). Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review*, *19*(3), 239-264.
12. de Jong, T. (2006). Scaffolds for scientific discovery learning. In J. Elen, R. E. Clark, & J. Lowyck (Eds.), *Handling complexity in learning environments: Theory and research* (pp. 107–128). Howard House: Emerald Group Publishing.
13. Hmelo-Silver, C. E., Duncan, R. G., & Chinn, C. A. (2007). Scaffolding and Achievement in Problem-Based and Inquiry Learning: A Response to Kirschner, Sweller, and Clark (2006). *Educational Psychologist*, *42*(2), 99–107. doi:10.1080/00461520701263368
14. Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. Journal of Child Psychology and Psychiatry, 17(2), 89-100. doi:10.1111/j.1469-7610.1976.tb00381.x
15. Holmes, N. G. (2011). The invention support environment: using metacognitive scaffolding and interactive learning environments to improve learning from invention (Master's dissertation, University of British Columbia).
16. Aleven, V., McLaren, B. M., Sewall, J., & Koedinger, K. R. (2009). A new paradigm for intelligent tutoring systems: Example-tracing tutors. *International Journal of Artificial Intelligence in Education*, *19*(2), 105-154.
17. Roll, I., Aleven, V., & Koedinger, K. R. (2011). Outcomes and mechanisms of transfer in Invention activities. In *Proceedings of the 33rd annual conference of the cognitive science society* (p. 2824-2829.

Embedded Scaffolding for Reading Comprehension in Open-Ended Narrative-Centered Learning Environments

Jonathan P. Rowe, Eleni V. Lobene, Bradford W. Mott, and James C. Lester

Department of Computer Science, North Carolina State University, Raleigh, NC 27695
{jprowe, eleni.lobene, bwmott, lester}@ncsu.edu

Abstract. Narrative-centered learning environments tightly integrate educational subject matter and interactive stories, where students serve as active participants in story-centric problem-solving scenarios. Embedding scaffolding within the storyline of a narrative-centered learning environment is a discreet approach to supporting students' learning processes without diminishing the motivational benefits of interactive narratives. This paper presents an implementation of story-embedded scaffolding in a narrative-centered learning environment, CRYSTAL ISLAND. CRYSTAL ISLAND's curricular focus has recently been expanded to include literacy education, with a focus on reading. Scaffolding takes the form of *concept matrices*, which are student-generated graphic organizers for complex informational texts that students read as part of CRYSTAL ISLAND's interactive narrative plot. Leveraging generative learning theory, we discuss directions for fading concept matrix-based scaffolding, and examine technical challenges and potential solutions.

Keywords: Narrative-centered learning environments, scaffolding, reading.

1 Introduction

There is growing evidence that narrative-centered learning environments, a class of game-based learning environments that embed educational content in interactive story scenarios, are an effective medium for fostering student learning and engagement [1–2]. A key benefit of narrative-centered learning environments is their capacity to discreetly support students' learning processes by tightly integrating educational and narrative elements. Guiding student problem solving in open-ended narrative-centered learning environments is particularly important, because students often have varying degrees of competency at solving ill-structured problems. Consequently, scaffolding in narrative-centered learning environments should meet at least two requirements: scaffolding should be dynamically tailored to individual students, and scaffolding should be naturalistically embedded within interactive narratives in order to sustain student engagement.

This paper proposes extensions to an open-ended narrative-centered learning environment, CRYSTAL ISLAND, that incorporate story-embedded scaffolding features for literacy education using generative graphic organizers. In CRYSTAL ISLAND, reading comprehension is critical for students gathering clues to solve a science

Fig 1. CRYSTAL ISLAND narrative-centered learning environment.

problem-solving mystery. Adaptively scaffolding students' reading processes is a promising direction for enhancing students' literacy skills, and has been the subject of considerable research by the intelligent tutoring systems community [3–4]. We describe how CRYSTAL ISLAND's plot and game mechanics currently incorporate story-embedded graphic organizers to scaffold students' reading comprehension processes, and outline future directions for intelligently diagnosing and fading this scaffolding.

2 CRYSTAL ISLAND for Literacy Education

Over the past several years, our lab has been developing CRYSTAL ISLAND (Fig. 1), a narrative-centered learning environment for middle school microbiology [1]. CRYSTAL ISLAND's curricular focus has recently been expanded to include literacy education based on Common Core State Standards. CRYSTAL ISLAND's narrative focuses on a spreading illness afflicting a research team on a remote island. Students act as medical detectives who must diagnose and treat the illness to save the team.

As part of CRYSTAL ISLAND's curricular focus on literacy, students encounter books and articles throughout the camp that contain complex informational texts about microbiology concepts (Fig. 2, left). Students read and analyze these texts, as well as complete associated concept matrices, to acquire knowledge to diagnose the illness. Concept matrices (Fig. 2, right) are graphic organizers, which students use to record key pieces of information encountered in the informational texts. The concept matrices are framed within the narrative as partially completed notes written by one of the research team's sick scientists. Students must discover and "complete" the notes based on content in the informational texts. The graphic organizers serve both as scaffolds for reading comprehension, as well as embedded assessments of

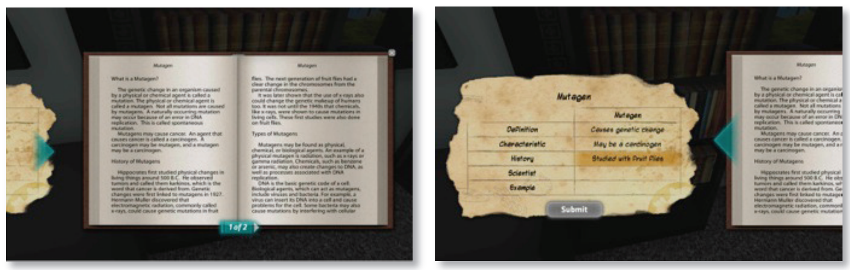


Fig 2. (Left) An informational text stylistically formatted like a virtual book, and (Right) a concept matrix stylistically formatted as a scrap of note paper.

students' reading comprehension skills. Completing a concept matrix involves clicking on each blank cell and selecting responses from drop-down menus. After a student has filled out a concept matrix, she can press an on-screen "Submit" button to receive immediate feedback on her responses.

3 Story-Embedded Scaffolding for Reading Comprehension

Graphic organizers, such as concept matrices, provide a natural mechanism for scaffolding reading comprehension skills in a non-obtrusive manner within narrative-centered learning environments. However, generative learning theory suggests that students will achieve improved learning gains if they create the concept matrices themselves. The current implementation of concept matrices in CRYSTAL ISLAND is highly structured. We plan to extend the current approach by intelligently reducing concept matrices' pre-specified structure as students improve their reading skills. Specifically, we propose fading the story-embedded scaffolding by transitioning from highly structured concept matrices to increasingly student-generated concept matrices.

Currently, whenever a student encounters a concept matrix in the story world, the matrix's layout (i.e., number of columns, number of rows) is fixed, the headings are pre-specified, and the set of possible answers for each cell are given. Fading the structure of story-embedded concept matrices can occur in at least three stages. First, one could remove the multiple-choice response menus for interior cells, instead requiring students to enter free-form text. This would require students to independently identify relationships between key terms and concepts from informational texts. Second, one could remove the pre-specified headers for each column and row, replacing them with either multiple-choice menus or free-form text entries. This would require students to independently identify the important themes in informational texts. Third, one could require students to specify the concept matrix layouts by selecting their number of columns and rows. This would require students to independently evaluate which, and how many, themes are most salient.

Effectively fading concept matrix-based scaffolding within CRYSTAL ISLAND raises notable technical challenges. The first challenge is identifying when to transition between successive levels of fading. This could be implemented as a fixed progression (e.g., if the student has encountered *N* concept matrices, fade by one level). Alternatively, fading decisions could be based on probabilistic student models—a common practice in ITSs—although assessing student knowledge from concept matrices presents its own challenges. One could also leverage reinforcement learning to induce optimal fading policies from an exploratory corpus of student interaction data, a technique that has shown success in tutorial dialogue modeling [5].

A second challenge is automatically assessing the quality of student-generated concept matrices. Automated assessment would require models of important concepts and themes from informational texts, as well as robust techniques for comparing informational text models to student-generated concept matrices, which may suffer from spelling errors, misconceptions, and incompleteness. Third, providing feedback tailored to individual students based on their self-generated concept matrices is difficult. Feedback could concern a broad range of subjects, such as corrections of factual errors, clarifications about important themes, or suggestions for alternate layouts, and it would need to cope with students' free-form written content.

Automated assessment and feedback raise interesting computational challenges, but intermediate solutions may exist. For example, it seems plausible that one could identify constraints that good concept matrices meet (e.g., included content terms, content of rows/columns), suggesting that constraint-based models [6] may show promise. While the computational challenges are substantial, tailoring and fading generative graphic organizers to scaffold reading comprehension in open-ended narrative-centered learning environments shows considerable promise for promoting both effective and engaging literacy learning experiences.

Acknowledgments. This research was supported by the National Science Foundation under Grant DRL-0822200. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. Additional support was provided by the Bill and Melinda Gates Foundation, the William and Flora Hewlett Foundation, and EDUCAUSE.

References

1. Rowe, J.P., Shores, L.R., Mott, B.W., Lester, J.C.: Integrating Learning, Problem Solving, and Engagement in Narrative-Centered Learning Environments. International Journal of Artificial Intelligence in Education. 21, 115–133 (2011)
2. Johnson, W.L.: Serious Use of a Serious Game for Language Learning. International Journal of Artificial Intelligence in Education. 20, 175–195 (2010)
3. Chen, W., Mostow, J., & Aist, G.S.: Recognizing Young Readers' Spoken Questions. International Journal of Artificial Intelligence in Education. 21 (2011)
4. Dela Rosa, K. & Eskenazi, M.: Self-Assessment of Motivation: Explicit and Implicit Indicators in L2 Vocabulary Learning. In: 15th International Conference on Artificial Intelligence in Education, pp. 296–303. Springer-Verlag, Berlin Heidelberg (2011)
5. Chi, M., VanLehn, K, Litman, D., Jordan, P.: Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical tactics. User Modeling and User Adapted Instruction, 21, 137–180 (2011)
6. Mitrovic, A.: Fifteen years of constraint-based tutors: What we have achieved and where we are going. User Modeling and User-Adapted Interaction. 22, 39–72 (2011)

Suggest-Assert-Modify: A Taxonomy of Adaptive Scaffolds in Computer-Based Learning Environments

James R. Segedy, Kirk M. Loretz, and Gautam Biswas

Institute of Software Integrated Systems, Department of Electrical Engineering and Computer Science, Vanderbilt University, 1025 16th Avenue South, Nashville, TN, 37212, U.S.A.
{james.segedy,kirk.m.loretz,gautam.biswas}@vanderbilt.edu

Abstract. Adaptive scaffolding in computer-based learning environments (CBLEs) continues to be an active area of research, with researchers framing the problem as determining the *what*, *when*, *how*, and *by whom or what* of adaptive scaffolding strategies. This paper presents our recent work in developing a taxonomy for adaptive scaffolds in CBLEs. The taxonomy, motivated by previous work in developing adaptive scaffolds, attempts to address the *how* of scaffolding by describing the tools and techniques available for scaffolding in CBLEs. We present the taxonomy, which describes adaptive scaffolds as one or more suggestions, assertions, and learning task modifications, and we discuss the utility of the taxonomy in describing adaptive scaffolding strategies.

Keywords: adaptive scaffolds, taxonomy, computer-based learning environments

1 Introduction

Research in computer-based learning environments (CBLEs) has long recognized the vital role of adaptivity in the success of a system's ability to independently foster learning in students [1]. Adaptive CBLEs regularly capture and analyze student activities in order to make decisions about how and when to scaffold learners [2]. These systems *take explicit actions* [3]; they may remind learners of relevant information, advise learners on how to proceed in their learning tasks, or modify the difficulty level of the learning activity itself.

The methods and tools used for scaffolding may vary widely based on the goal of instruction. For example, Chi and colleagues [4] presented 15 types of scaffolding actions identified in the research literature. These scaffolds include providing hints, fill-in-the-blank prompts, explanations, and correct answers, among others. Understanding these techniques, including when and why a particular scaffold may be more effective than another, remains an important area of research. Pea [5] framed the problem as defining the *what*, *why*, and *how* of scaffolding. *What* information should a scaffolding action focus on, w*hy* should a CBLE employ a scaffold, and *how* does the CBLE actually scaffold the learner (*i.e.*, what action does it take)? This framework was later revised by Azevedo & Jacobson [2] to focus on *what*, *when*, *how*, and *by whom or what*. The revised framework replaces the *why* question with a *when* question: *when* should a CBLE scaffold learners? It also introduces a new question: who or what should provide the scaffolds?

In this paper, we attempt to address the *how* question by presenting a novel taxonomy for classifying adaptive scaffolds in CBLEs. The taxonomy classifies adaptive scaffolds as a set of one or more *suggestions*, *assertions*, and learning task *modifications* (SAMs). Section 2 presents the background and motivation for the taxonomy; section 3 presents the taxonomy; and section 4 discusses future directions.

2 Previous Work in Classifying Adaptive Scaffolds

While some researchers in the field of educational technology have proposed methods for classifying and describing adaptive scaffolding approaches based on well-defined terms (*e.g.*, [6-7]), no comprehensive taxonomy of the tools and techniques available for scaffolding currently exists. Thus, the field now suffers from a lack of operational definitions, and several researchers refer to the scaffolds in their systems as "hints" or "feedback." Often, researchers define these scaffolds via examples. Bell & Davis [8], for instance, differentiate between three types of hints provided by a pedagogical agent named Mildred: activity hints, evidence hints, and claim hints. The provided descriptions of the hints are vague, and they are mainly illustrated with examples:

> The current instantiation of Mildred provides three types of hints - on activities, evidence, and claims. For example, in the "Critique Evidence" activity of All The News, an activity hint might say, "When you critique the evidence, you will think about: (1) the science ideas used in the evidence, (2) the methods used to create the evidence, and (3) how credible or believable the evidence is." Further activity hints for the Critique Evidence activity would provide definitions and examples of the critique criteria of science, methods, and credibility. Evidence hints are more specific, providing help in thinking about a particular piece of evidence. A hint for the "Bicyclists at Night" evidence (used in both All The News and How Far) is, "Why is the person in white [clothes] easier to see? What is happening to the light?" A student working on a critique of the Bicyclists at Night evidence could then receive converging evidence on both the act of critiquing and the specific evidence being critiqued. Likewise, claim hints help students think about a particular claim. For example, a claim hint about black "attracting heat" (as opposed to absorbing light) might say, "What would happen if there were a heat source in a dark room? Would someone wearing black get hotter than someone wearing white?" (p. 144)

Similarly, Jackson, Guess, & McNamara [9] present a CBLE, *iStart*, and describe the scaffolds provided by the system as "feedback" without defining the term, instead relying on examples:

> Merlin provides feedback for each explanation generated by the student. For example, he may prompt them to expand the explanation, ask the students to incorporate more information, or suggest that they link the explanation back to other parts of the text. Merlin sometimes takes the practice one step further and has students identify which strategies they used and where they were used. (p. 129)

Some researchers have developed more specific scaffold classifications. For example, Belland, Glazewski, & Richardson [10] propose four types of scaffolds: conceptual support, metacognitive support, procedural support, and strategic support. These support types are defined as help about "what to consider," "how to manage the learning process," "how to use tools," and "what strategies to use in approaching the problem," respectively. This classification differentiates scaffolds based on a single dimension: the type of information the scaffold is designed to support. However, because scaffolds are *actions*, an appropriate classification needs to consider both what information is supported and how it is supported.

In presenting a general framework for the design of Intelligent Tutoring Systems (ITSs), VanLehn [6] defines minimal feedback and three types of hints: point, teach, and bottom out. In ITSs, learners are presented with small multi-step problems in a well-defined domain (*e.g.*, physics). When students are having trouble correctly completing a problem step, the system usually intervenes to provide one of these types of scaffolds. Minimal feedback scaffolds indicate whether or not a learner's attempt at completing a problem step is correct or incorrect. Hints are provided in relation to a particular knowledge component (*e.g.*, a fact, definition, or procedure), and they are defined as follows:

> Pointing hints mention problem conditions that should remind the student of the knowledge component's relevance. Teaching hints describe the knowledge component briefly and show how to apply it. Bottom-out hints tell the student [how to apply the knowledge component to solve] the [current problem] step. (p. 242)

This scaffold classification, unlike the classification described in [10], does focus both on the information the scaffold is designed to support and the methods by which the information is supported. However, it is not general enough to classify a number of scaffolds that have been implemented in CBLEs. For example, several CBLEs provide scaffolds that suggest the use of a particular resource within the system rather than mentioning or explaining a knowledge component.

As a final example, Graesser & McNamara [7] describe the scaffolds implemented within a CBLE called *AutoTutor*, which teaches physics by posing questions and then holding natural language dialogues with learners as they attempt to answer those questions. During the course of these dialogues, *AutoTutor* may employ any of five types of dialogue moves: pumps, hints, prompts, correctness feedback, and assertions. *Pumps* ask the learner to continue elaborating on the answer they have started to offer. For example, *AutoTutor* might encourage a student to "keep going." *Hints* are questions that attempt to elicit a question-relevant proposition from the learner. For example, *AutoTutor* may ask students how Newton's second law of motion applies to the current question. *Prompts* are questions that ask the learner to provide explicit words or phrases that are important in answering the current question. For example, *AutoTutor* may present a partial definition of Newton's second law of motion and ask the learner to fill in the missing information. *Feedback* indicates whether the learner's answer is correct or incorrect, and *assertions* communicate entire propositions to learners when hints and prompts fail to elicit them.

In considering the presented scaffold classifications, some common themes emerge. First, several of the presented scaffolds operate by *providing a suggestion*. For example, pointing hints in ITSs direct attention to specific problem features, suggesting that learners consider those features; Merlin suggests that learners link their current explanation back to other parts of the text; and *AutoTutor* pumps learners, suggesting that they continue elaborating on their answer. Second, several of the presented scaffolds operate by *asserting information*. For example, teaching hints assert knowledge components and how to apply them; bottom-out hints assert how to solve the current problem step; and *AutoTutor's* assertions communicate question-relevant propositions to learners. Third, some scaffolds operate by *modifying the learning task*. For example, when *AutoTutor* asks the learner a question as part of delivering a hint, it is redirecting the learner's attention away from their former task (answering the original question) to a new task (answering a related question).

These observations have led us to develop a taxonomy that classifies adaptive scaffolds as one or more suggestions, assertions, and learning task modifications. This taxonomy is general and widely-applicable. Moreover, it provides a language for presenting and communicating scaffolding strategies.

3 The Suggest-Assert-Modify Taxonomy

The Suggest-Assert-Modify (SAM) taxonomy is illustrated in Figure 1. Suggestion scaffolds provide information to learners for the purpose of prompting them to engage in a specific behavior (*e.g.*, accessing a resource). By executing the recommended behavior, learners should encounter critical information that, if properly internalized, would allow them to make progress in accomplishing the learning task. The taxonomy classifies suggestions based on whether they target metacognitive activities (*e.g.*, planning or reflection) or cognitive knowledge integration activities. Knowledge integration is the process of analyzing and connecting multiple chunks of information in order to achieve new understandings about how they are related [11-12]. It can target several cognitive processes, such as: (i) goal orientation, in which learners integrate chunks of information with their understanding of their current goal; (ii) explanation construction, in which learners assemble chunks of information to explain a system, process, or phenomenon; (iii) prediction, in which learners integrate chunks of information with a hypothetical scenario, and several others.

Assertion scaffolds communicate information to learners as being true; ideally, learners will integrate this information with their current understanding as they continue working toward completing their learning task. Unlike suggestions, assertion scaffolds don't directly encourage learners to engage in a particular behavior; they only state information.

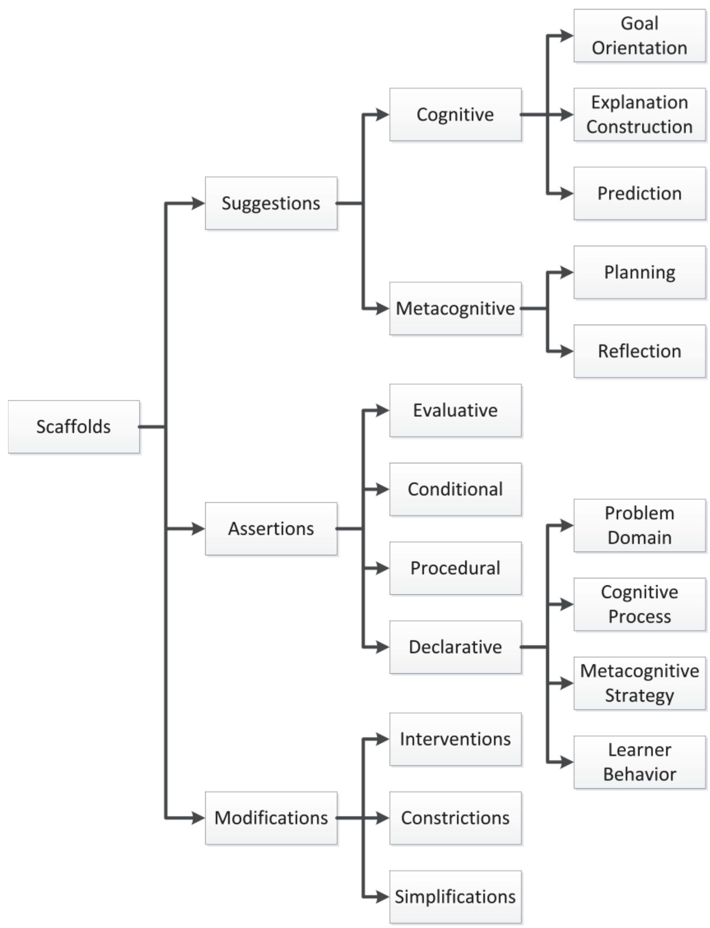


Fig. 1. The SAM Taxonomy for Adaptive Scaffolds

The taxonomy distinguishes between four types of assertion scaffolds: declarative, procedural, conditional, and evaluative. Declarative assertions communicate "knowing that" information [11]. Such information is often conceptualized as being represented as and with schemata: mental structures that represent a concept and the features that characterize it [12]. For example, a schema representing an animal might contain features such as the animal's number of legs and the sound that the animal makes. Features correspond to variables in an algebra expression or computer program; they can take on any of a number of values when instantiated; and an "instance" of an animal schema may represent an actual animal in the world. Thus, declarative assertions contain information that may be represented by a schema; this includes facts, definitions, concepts, and understandings of relationships and interrelationships among actors in complex systems. In the proposed taxonomy, declarative assertions are sub-divided based on their topic, which may be the problem domain, cognitive processes, metacognitive strategies, and the learner's behavior while using the system. Examples of each type of declarative assertion are listed in Table 1.

Assertion Category	Example
Declarative – Problem Domain	Sunfish eat mosquito fish.
Declarative – Cognitive Processes	You have to know how to multiply fractions.
Declarative – Metacognitive Strategies	The "cross-multiply" strategy may help you.
Declarative – Learner Behavior	You haven't tried any division problems.
Procedural	To multiply fractions, first multiply the numerators, and then multiply the denominators.
Conditional	The "cross-multiply" strategy should be used whenever you need to solve for an unknown value in an equation consisting of only fractions.
Evaluative	You don't seem to have a good understanding of how to divide fractions.

Table 1. Types of Assertion Scaffolds with Examples.

Procedural assertions communicate "how-to" information: sets of actions that, when executed in a loosely-ordered sequence, can accomplish a task. These assertions explain how to perform cognitive processes, such as identifying important information in text passages or applying causal reasoning to answer hypothetical questions. Conditional assertions communicate information represented as "if-then" rules that identify both when cognitive processes are applicable and whether or not they should be executed based on the current context [12]. These assertions usually explain metacognitive strategies. In a fractions learning environment, for example, the system might assert that a good strategy for solving algebraic expressions that consist entirely of fractions is to use a "cross-multiply" strategy. This would be represented as the following "if-then" rule: *IF you want to solve an algebraic expression consisting entirely of fractions, THEN employ the cross-multiply strategy*. Finally, evaluative assertions communicate evaluations of the learner's performance and understanding. For example, the system may assert that the learner does not seem to understand how to divide fractions.

Modification scaffolds, unlike suggestion and assertion scaffolds, do not operate by communicating information to the learner; rather, they change aspects of the learning task itself. In doing so, they seek to adapt the task to the learner's needs and abilities. The taxonomy differentiates between three types of modification scaffolds: simplifications, constrictions, and interventions. Simplification modifications, as specified by Wood, Bruner, & Ross [13], operate by "reducing the number of constituent acts required to reach solution." Constriction modifications operate by reducing the number of options available to the learner. For example, the scaffolding agent may block access to tools or resources in order to focus learners' attention on other, more useful approaches to solving the task. Intervention scaffolds, rather than modifying features of the overall task, operate by temporarily shifting learners' attention from their primary task to an intervention task. Upon completion of the intervention task, learners may return to the primary task.

The SAM taxonomy addresses the *how* of scaffolding by describing the atomic elements of adaptive scaffolds, and it provides a language for communicating both individual scaffolds and entire scaffolding strategies. For example, the scaffolding strategy for ITSs discussed by VanLehn [6] could be described as a progression from cognitive suggestions (pointing hints) to declarative assertions that describe a knowledge component (teaching hints) to declarative assertions that provide the answer to the current problem step (bottom-out hints). In comparison to the scaffolding classifications presented in Section 2, we argue that the SAM taxonomy is more comprehensive and general than its predecessors.

4 Conclusion

This paper has presented a novel taxonomy for describing and classifying adaptive scaffolds in computer-based learning environments. The taxonomy classifies adaptive scaffolds as one or more suggestions, assertions, and learning task modifications, and it provides a general, widely-applicable language for communicating and interpreting scaffolding strategies.

The SAM taxonomy, however, is not without limitations. First, the distinction between suggestions and assertions is sometimes ambiguous, and a scaffold may consist of an assertion that implies a suggestion. For example, a scaffold in an algebra learning environment may assert that successful students used a particular problem solving strategy in order to indirectly suggest that the learner adopt that strategy. Second, the SAM taxonomy does not currently distinguish between different types of intervention scaffolds. Future work should investigate methods for breaking down interventions according to the types of activities learners are expected to accomplish during the intervention. For example, it may be valuable to separate modeling interventions (*e.g.*, demonstrating how to solve a problem), metacognitive interventions (*e.g.*, requiring learners to gauge their own comprehension), and cognitive interventions (*e.g.*, requiring learners to correctly define terms or explain properties of a complex system).

It is important to note that the presented taxonomy represents an initial step toward a standardized language for describing the *how* of adaptive scaffolding strategies. As we continue to scan the literature for more examples of adaptive scaffolds in educational technology, we will update the taxonomy as needed to reflect distinguishing features of adaptive scaffolds.

Acknowledgements. This work has been supported by Institute of Educational Sciences CASL Grant #R305A120186

References

1. Park, O.C., Lee, J.: Adaptive Instructional Systems. In: Jonassen, D.H. (ed.) Handbook of Research for Education Communications and Technology, pp. 651-684. Erlbaum, Mahwah (2004)
2. Azevedo, R., Jacobson, M.J.: Advances in scaffolding learning with hypertext and hypermedia: A summary and critical analysis. Educational Technology Research and Development, 56, 93-100 (2008)
3. Puntambekar, S., Hübscher, R.: Tools for scaffolding students in a complex learning environment: What have we gained and what have we missed? Educational Psychologist, 40, 1-12 (2005)
4. Chi, M.T.H., Siler, S.A., Jeong, H., Yamauchi, T., Hausmann, R.G.: Learning from human tutoring. Cognitive Science, 25, 471-533 (2001)
5. Pea, R.D.: The social and technological dimensions of scaffolding and related theoretical concepts for learning, education, and human activity. The Journal of the Learning Sciences, 13, 423-451 (2004)
6. VanLehn, K.: The behavior of tutoring systems. International Journal of Artificial Intelligence in Education, 16, 227–265 (2006)
7. Graesser, A.C., McNamara, D.: Self-regulated learning in learning environments with pedagogical agents that interact in natural language. Educational Psychologist, 45, 234-244 (2010)
8. Bell, P., Davis, E.A.: Designing Mildred: Scaffolding students' reflection and argumentation using a cognitive software guide. In: Fishman, B., O'Connor-Divelbliss, S. (eds.) Fourth International Conference of the Learning Sciences, pp. 142-149. Erlbaum, Mahwah (2000)
9. Jackson, G.T., Guess, R.H., McNamara, D.S.: Assessing Cognitively Complex Strategy Use in an Untrained Domain. Topics in Cognitive Science, 2, 127-137 (2010)
10. Belland, B.R., Glazewski, K.D., Richardson, J.C.: A scaffolding framework to support the construction of evidence-based arguments among middle school students. Educational Technology Research and Development, 56, 401-422 (2008)
11. Anderson, J.R.: ACT: A simple theory of complex cognition. American Psychologist, 51, 355-365 (1996)
12. Winne, P.H.: Self-regulated learning viewed from models of information processing. In: Zimmerman, B.J., Schunk, D.H. (eds.) Self-Regulated Learning and Academic Achievement: Theoretical Perspectives, pp. 153-189. Erlbaum, Mahwah (2001)
13. Wood, D., Bruner, J.S., Ross, G.: The role of tutoring in problem solving. Journal of Child Psychology and Psychiatry, 17, 89-100 (1976)

Exploring Adaptive Scaffolding in a Multifaceted Tangible Learning Environment

Elissa Thomas, Victor Girotto, Alex Abreu, Cecil Lozano, Kasia Muldner, Winslow Burleson, Erin Walker

Computing, Informatics & Decision Systems Engineering, Arizona State University

{eethomas, victor.girotto, alexabreu, cecil.lozano, katarzyna.muldner, winslow.burleson, erin.a.walker}@asu.edu

Abstract. The majority of educational software is designed for traditional computers, which allow little opportunity for physical manipulation of an environment. Tangible Activities for Geometry (TAG) provides students a tangible learning environment. Currently, however, TAG does not employ adaptive scaffolding techniques. Accordingly, we describe how scaffolding techniques and teachable agent behaviors can be integrated into TAG to improve this tangible learning environment.

Keywords: adaptive scaffolding, tangible learning environments, teachable agents

1 Introduction

Open-ended learning environments (OELEs) enable students to actively engage in problem solving, such as generation, testing and revision of a hypothesis [1]. However, most educational systems target personal computers and their typical WIMP (window, icon, menu, pointing device) setup. These systems rarely allow for embodied interaction between the student and the learning environment, despite the fact that students learn a great deal through physically engaging with their environment [2]. The *Tangible Activities for Geometry* system (TAG) aims to fill this gap, by providing a tangible OELE where students can move beyond the boundaries of the virtual world and explore different strategies for solving geometric problems [3].

The current TAG system provides no feedback or adaptation to the user's performance. Therefore, our goal with this paper is to propose ways of integrating adaptive scaffolding techniques into this tangible learning environment (TUI), laying the foundation for studying the effects that they would have in this type of learning environment. The majority of TUIs do not currently possess such capabilities, which allows us to start exploring this intersection. Here, we will review existing frameworks and techniques that can be used for scaffolding the user's learning in an adaptive manner and will describe ways in which they could be applied to our system.

2 Description of Current System

In the current implementation of the TAG system, a student solves geometry problems by instructing a teachable agent on the steps needed to solve the problem. Problems include plotting a point in a given quadrant, translating a point, or rotating a point around a center of origin. While answers are sometimes the same, problems can often be solved in different ways. The system is comprised of three main components [3]. The *problem space* is a Cartesian plane projected on the ground. This is where the teachable agent and the problem objects, such as lines and points, are displayed. The interactions with the problem space occur through a *hanging pointer* that hangs from the ceiling, functioning as a mouse. Hovering the pointer over the problem space moves the cursor. Clicking is performed when the user moves the pointer below a certain height threshold and back up. The feedback for the user's interactions on the problem space is received on the *mobile interface,* displayed on an iPod Touch. In this interface, the user is able to select an action that will be performed by the agent, view the steps already taken, and navigate through problems.

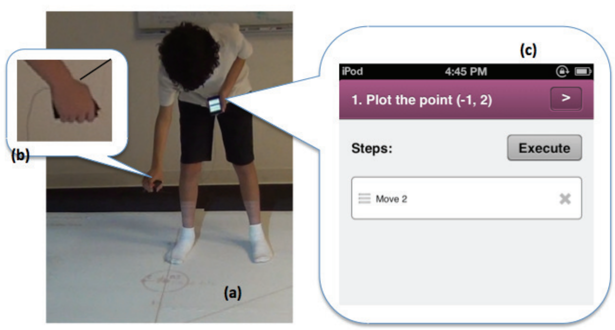


Figure 1: Elements of the TAG system. The problem space (a), where the Cartesian plane is projected, the hanging pointer (b), used by the student to interact with the problem and the mobile interface (c), the iPod interface commands are issued to the agent.

3 Review of Existing Pedagogical Techniques

Prior research has explored how various pedagogical techniques impact student learning. A number of these rely on a teachable agent paradigm, where students learn by tutoring a computerized agent modeled to simulate behaviors of a student tutee. For instance, reflective knowledge building uses questions and explanations generated by a teachable agent to prompt students to reflect on their own understanding of various concepts, and refine their ideas [4]. Agents could also use this technique to introduce new ideas to a student's existing knowledge [5].

Other research has shown that the level of abstraction in the advice provided by a teachable agent can impact a student's perceptions and performance. Students who work with agents that give different kinds of feedback, ranging from high-level advice to concrete, task specific suggestions, performed better than students who interacted with agents that only used task-specific suggestions [6].

Techniques used in cognitive tutors can also be useful for extending TAG. Cognitive tutors provide the user with feedback on a step-by-step basis, in response to common errors and with on-demand instructional hints, and adapt the selection of problems based on user-performance [7]. The challenge is to adapt these techniques to an open-ended system such as TAG while still encouraging open-ended exploration.

4 Proposed Extensions on the Current System

We propose expanding TAG to employ adaptive scaffolding as a way to increase the system's effectiveness. Techniques such as reflective knowledge building could be integrated into our system to improve student learning while also enhancing unique tangible aspects of our system. For example, if the student is attempting to plot a point in quadrant II, but moved the agent into quadrant IV, a question from the agent might prompt the student to recognize that their actions are not leading them to the correct solution. As another example, after a student solves a problem, the TAG agent could propose an alternate solution, helping students evolve their ideas, which some students struggle to do in OELEs [8]. As an extension of adaptive scaffolding in a traditional learning environment, students could also be encouraged to try additional tangible interactions that may not have been incorporated into their original solution.

Scaffolding could also be employed through hints given by the agent while a student is working on a problem. In this scenario, the agent uses cues that a student might be confused, such as a long pause without any activity, and provides a hint to guide the student in the right direction. Are there unique cues within TUIs that could be detected to improve an adaptive scaffolding model? To study this, our system could monitor embodied behaviors exhibited by the student, such as pacing back and forth or kneeling down on the projected Cartesian plane. Following standard convention, the agent's hints should vary in detail based on the student's performance within a given problem. Students would initially be provided with high-level feedback from the teachable agent, allowing them to apply the information given to them by the agent to the problem domain. If the student continues having trouble, the system can adaptively adjust the agent's hints to be more direct, allowing students to discover the correct approach, albeit, with less reflection on the metacognitive process. By providing feedback in this manner, we can foster an atmosphere of discovery, which should help students feel more engaged [2]. Since previous work has shown that increasing the sociability of an agent improves student perceptions of the system and student performance [9], hints from the agent could be provided textually through a pop up on the iPod interface while also being spoken by the agent.

On a less localized scale, adaptive scaffolding could also be applied based on a student's performance throughout an entire session. Indicators that could be used to measure student performance include the amount of time taken to solve a problem, the number of correct and incorrect solutions a student has produced, and the number of steps a student uses as compared to an optimal solution with a minimal number of steps. Applying this type of adaptive scaffolding in a TUI introduces some unique challenges. For example, how do we differentiate between students that are struggling with the problem domain and students that are having trouble understanding how to use the unique tangible interactions of our system?

5 Conclusion

By proposing a novel set of techniques to augment the TAG system, we aim to provide the appropriate level of scaffolding needed to improve student learning, while maintaining student engagement when faced with difficulties and failure. The ultimate goal is to ensure that students receive help when it is needed, but are not hindered during open-ended exploration. We also hope to learn more about how this scaffolding should be presented to the student on the different dimensions that a TUI provides, exploring the advantages and drawbacks of each type of scaffolding.

References

1. Land S. Cognitive requirements for learning with open-ended learning environments. Educational Technology Research and Development. 2000, Volume 48, Issue 3, pp 61-78.
2. Walker, E, and Burleson, W. User-Centered design of a teachable robot. Intelligent Tutoring Systems, 2012.
3. Mulder, K., Lozano, C., Girotto, V., Burleson, W., and Walker, E. Designing a Tangible Learning Environment with a Teachable Agent. Artificial Intelligence in Education, 2013.
4. Roscoe, D., Wagster, J., and Biswas, G., Using Teachable Agent Feedback to Support Effective Learning by Teaching, In Proceedings of the 30th Annual Meeting of the Cognitive Science Society, Washington, DC, 2008.
5. Blair, K., Schwartz, D., Biswas, G., and Leelawong, K. Pedagogical Agents for Learning by Teaching: Teachable Agents. In Educational Technology & Society: Special Issue on Pedagogical Agents, 2006.
6. Lester, J. C., Converse, S. A., Kahler, S. E., Barlow, S. T., Stone, B. A., and Bhogal, R. S. The Persona Effect: Affective Impact of Animated Pedagogical Agents. In Proceedings of CHI '97, 1997.
7. Koedinger, K., Aleven, V. Exploring the Assistance Dilemma in Experiments with Cognitive Tutors. 2007.
8. Land, S. M. Cognitive Requirements for Learning with Open-Ended Learning Environments. Educational Technology Research and Development 48.3, 2000.
9. Hershey D. K., Mishra P., and Altermatt, E. All or nothing: Levels of sociability of a pedagogical software agent and its impact on student perceptions and learning. Journal of Educational Multimedia and Hypermedia 14.2, 2005.

"Gaming the system" in Newton's Playground

Lubin Wang, Yoon Jeon Kim, & Valerie Shute

Florida State University, Tallahassee, USA
lw10e@fsu.edu,yk06c@my.fsu.edu,vshute@fsu.edu

Abstract. This paper describes the current status of ongoing research looking into students' "gaming the system" behaviors in an open-ended learning environment—the game Newton's Playground—in relation to their physics learning, enjoyment of the game, and persistence. Our next step is to code students' gaming behaviors and then compare learning via pretest and posttest scores. We'll also examine gaming behaviors relative to enjoyment of the game and persistence. Findings can inform improvements to Newton's Playground (and other games) and guide the design of scaffolding for students in other OELEs.

Keywords: game the system behaviors, game-based learning, physics learning, persistence

1 Introduction

Open-ended learning environments (OELEs) are technology-rich environments that allow learners to participate in authentic problem solving activities, interact with the system by actively making choices, and apply cognitive and metacognitive skills to assess and monitor their learning processes [5]. Providing players the freedom to explore the environment and make choices are essential features of OELEs, which render the environment engaging and meaningful.

Well-designed digital games share similar features with such environments [1]. For example, Gee (2003) discusses properties of good games, such as interactive problem solving, adaptive challenges, feedback, and control that are aligned with learning principles to promote deep and meaningful learning. In games players actively interact with the system by making choices, and this provides a sense of control and ownership to the players. Also, games provide players with complex and interesting problems to solve, allowing freedom in terms of how they reach the solution.

In such wide-open environments, however, it is almost impossible to predict every possible way that learners will interact with the system. Studies have shown that for novice learners, having too much freedom can lead to frustration or unsuccessful learning [5]. This may result in unexpected behaviors by learners such as exploiting loopholes of the system, which is commonly referred to as gaming the system.

Baker (2005) defines gaming the system as "attempting to succeed in an educational environment by exploiting properties of the system rather than by learning the material and trying to use that knowledge to answer correctly (p. 6)." Reasons why learners game the system and how it influences learning have been investigated in various forms of technology rich learning environments, primarily in intelligent tutoring systems [1]. Broadly speaking, learners are more likely to show gaming the system behaviors when (a) they dislike the subject matter, (b) they are frustrated, and/or (c) they lack drive or motivation.

Unlike what happens in learning environments like intelligent tutoring systems, gaming the system is not always viewed negatively in the gaming context. In fact, it can be an important aspect of gaming culture as evidenced by a player proudly sharing certain "tricks" with other players [4]. Therefore, as using games for learning purposes becomes a more common practice in the broader education community, it is important for educators and researchers to understand why players would game the system and how such behavior influences learning.

2 Context

We propose to investigate gaming the system behaviors in a game called Newton's Playground (NP) [6]. NP is a two-dimensional computer game designed to assess and support qualitative physics and persistence. The core mechanic of the game is to guide a green ball to a red balloon by drawing physical objects and simple mechanical devices (i.e., ramp, lever, pendulum, springboard) on the screen that "come to life" once drawn. We call these devices "agents of force and motion" since they trigger or change the direction of motion. There are four types of agents that are categorized in terms of unique features and underlying physics principles: ramp, lever, pendulum, and springboard.

A ramp is any line drawn that guides a ball in linear motion, and it is commonly used for problems that require transfer of potential energy to kinetic energy. A lever rotates around a fixed point usually called a fulcrum or pivot point, and it is used to move the ball vertically. A swinging pendulum directs an impulse tangent to its direction of motion, which is used to exert a horizontal force. A springboard stores elastic potential energy provided by a falling weight, and is used to move the ball vertically.

As the use of these agents provides evidence for students' physics understanding, NP has a built-in evidence identification system that automatically categorizes (with > 95% accuracy when compared with human ratings) the type of agent based on salient features of drawn objects by students. Even though there is no absolute correct or incorrect way of solving problems, they are "probable agents" of force and motion that experts (or the game designers) expect players to use in given problems.

In the fall of 2012, we had 165 ninth graders play the game for around 4 hours (across a one-week time frame). We also administered pre- and posttests of physics to measure improvement of students' qualitative physics as the result of playing NP. As part of the study, we observed that some players came up with various ways to exploit the system, and we categorize them as stacking lines, breaking the system, and cutting corners (Table 1). We define these types of solutions as gaming the system behaviors in NP because these solutions (a) exploit loopholes in the system, and (b) do not require application of appropriate physics principles.

Table 1. Gaming the System in Newton's Playground

Gaming the system behaviors	Features
Stacking	Players consecutively draw short lines right below the ball to lift up the ball to the balloon. Players are likely to show this behavior when the balloon is above the ball.
Breaking the system	Players draw random lines across the given objects until the system crashes and acts randomly. Players are likely to show this behavior when either the balloon is above the ball or the path to the balloon is constrained by obstacles
Cutting corners	Players draw a line quickly beneath the ball that spans over to the balloon. Players are likely to show this behavior when the ball is moving away from the balloon or the starting point of the ball is higher than the balloon.

3 Research Questions

The present study aims to address the following questions:

1. How does gaming the system in NP influence players' physics learning?
2. How does gaming the system in NP relate to players' enjoyment of the game and persistence?

Our hypotheses are:

1. For most students, gaming the system is negatively related to players' physics learning;
2. For most students, gaming the system is negatively related to players' enjoyment of the game and persistence.

4 Method

First, two human raters will replay (with the "level replay" function in the game) all log files of a set of 16 problems that are solved by over 60% of the students, and manually code occurrences of gaming the system behavior related to the three identified categories (i.e., stacking, breaking the system, and cutting corners). Second, we will identify three different subgroups of players in terms of frequencies of the gaming the system behaviors (i.e., none, some, and a lot). Third, we will analyze differences among these subgroups in terms of physics learning (via pretest to posttest gains), enjoyment, and persistence. Note that we already have the data collected, and just need to conduct the observation of replay files, code the behaviors, and analyze the data.

5 Discussion and Implications

To ensure that learners with varying abilities can all benefit from playing games that are designed for learning, we need to identify any subgroups of students who may become lost in the environment and simply try to "cheat through" the problems without applying appropriate knowledge and skills. If our hypotheses are established, we will need to devise appropriate scaffolds in NP to minimize the gaming behavior and thus maximize learning and enjoyment. Potential scaffolds that may fit in NP include tutorial videos and visual aid function. For example, for the visual aid function, dotted lines will show up on the screen upon request, which provide students with clues for appropriate agents rather than having them get stuck and thus frustrated.

However, considering NP is still a game, any decisions regarding scaffolds need to balance with features of good games. That is, we need to be careful about how much scaffolds we provide, and how they are presented to students because poorly designed scaffolds in the game may spoil engaging features of the game (e.g., challenge, control, and adaptive difficulty).

In conclusion, gaming the system behaviors have not been fully investigated in the context of games for learning, and we first need to understand how these behaviors influence learning—i.e., are they always maladaptive or can they sometimes yield positive outcomes? We hope that this study will provide us with useful information about learners' gaming the system behaviors in NP in relation to learning and enjoyment, and also shed light on appropriate forms of scaffolding to be used to prevent such behaviors, if warranted. The findings from this study may also be of interest to researchers who are interested in gaming behaviors and possible scaffolding in OELEs.

References

1. Baker, R. S., Corbett, A. T., Roll, I., & Koedinger, K. R. (2008). Developing a generalizable detector of when students game the system. *User Modeling and User-Adapted Interaction, 18* (3), 287-314.
2. Baker, (2005). *Designing intelligent tutors that adapt to when students game the system* (doctoral dissertation). Carnegie Mellon University, Pittsburgh, PA.
3. Gee, J. P. (2003). *What Video Games Have to Teach Us About Learning and Literacy*? New York: Palgrave Macmillan.
4. Kuecklich, J. (2004). *Other playings— cheating in computer games*. Paper presented at Other Players Conference. Copenhagen, Denmark.
5. Land, S. M. (2000). Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research and Development, 48* (3), 61—78.
6. Shute, V. J., & Ventura, M. (2013). Measuring and supporting learning in games: Stealth assessment. Cambridge, MA: The MIT Press.