

Vincent Aleven
Judy Kay
Jack Mostow (Eds.)

LNCS 6095

Intelligent Tutoring Systems

10th International Conference, ITS 2010
Pittsburgh, PA, USA, June 2010
Proceedings, Part II

2
Part II



 Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Germany

Madhu Sudan

Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Vincent Aleven Judy Kay Jack Mostow (Eds.)

Intelligent Tutoring Systems

10th International Conference, ITS 2010
Pittsburgh, PA, USA, June 14-18, 2010
Proceedings, Part II

Volume Editors

Vincent Aleven

Carnegie Mellon University, Human-Computer Interaction Institute

5000 Forbes Avenue, Pittsburgh, PA 15213, USA

E-mail: aleven@cs.cmu.edu

Judy Kay

University of Sydney, School of Information Technologies

1 Cleveland Street, Sydney 2006, Australia

E-mail: judy.kay@sydney.edu.au

Jack Mostow

Carnegie Mellon University, School of Computer Science

5000 Forbes Avenue, Pittsburgh, PA 15213, USA

E-mail: mostow@cs.cmu.edu

Library of Congress Control Number: 2010927366

CR Subject Classification (1998): I.2.6, J.4, H.1.2, H.5.1, J.5, K.4.2

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743

ISBN-10 3-642-13436-X Springer Berlin Heidelberg New York

ISBN-13 978-3-642-13436-4 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2010

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper 06/3180

Preface

The 10th International Conference on Intelligent Tutoring Systems, ITS 2010, continued the bi-annual series of top-flight international conferences on the use of advanced educational technologies that are adaptive to users or groups of users. These highly interdisciplinary conferences bring together researchers in the learning sciences, computer science, cognitive or educational psychology, cognitive science, artificial intelligence, machine learning, and linguistics. The theme of the ITS 2010 conference was *Bridges to Learning*, a theme that connects the scientific content of the conference and the geography of Pittsburgh, the host city. The conference addressed the use of advanced technologies as bridges for learners and facilitators of robust learning outcomes.

We received a total of 186 submissions from 26 countries on 5 continents: Australia, Brazil, Canada, China, Estonia, France, Georgia, Germany, Greece, India, Italy, Japan, Korea, Mexico, The Netherlands, New Zealand, Pakistan, Philippines, Saudi Arabia, Singapore, Slovakia, Spain, Thailand, Turkey, the UK and USA. We accepted 61 full papers (38%) and 58 short papers. The diversity of the field is reflected in the range of topics represented by the papers submitted, selected by the authors. The most popular topics among the accepted (full and short) papers were: empirical studies of learning with advanced learning technologies (34 accepted papers), educational data mining (EDM) and machine learning (28), evaluation of systems (23), pedagogical agents (21), natural language interaction (20), affect (19), intelligent games (16), pedagogical strategies (15), models of learners, facilitators, groups and communities (15), and domain-specific: mathematics (15). Of course, many papers covered multiple topics.

We are delighted that five outstanding and world-renowned researchers accepted our invitation to give invited talks during the conference. Abstracts of their presentations are included in this set of proceedings. Chee-Kit Looi from the National Institute of Education (Singapore) shared insights into comprehensive initiatives in Singapore's education system, which involve partnerships between researchers and classroom practice. Stacy Marsella from the Institute of Creative Technologies (University of Southern California) spoke about the role of emotion and emotion modeling in systems with virtual characters. Alexander Renkl from the University of Freiburg (Germany) suggested a way of reconciling theoretical views on learning held by proponents of socio-constructivist approaches with cognitively oriented approaches and discussed implications for the design of ITS. Steven Ritter from Carnegie Learning, Inc. (Pittsburgh, USA) spoke about the third wave of ITS, which takes advantage of the large user base of real-world ITS for purposes of data mining and end-user authoring. Finally, Beverly Woolf, from the University of Massachusetts, Amherst, described the emergence of social and caring computer tutors, which respond to both affect and cognition.

The proceedings contain 17 short papers within the important Young Researchers Track (YRT). This track represents the future of our field. It provides a forum in which PhD students present and discuss their work during its early stages, with mentoring from more senior members of the community. All submissions were carefully reviewed by experts. The proceedings also include 18 abstracts of Interactive Events that during the conference showcased an interesting mixture of mature systems and late-breaking developments in ITS and related tools for authoring, assessment, data analysis, etc. Rounding out the scientific program of the conference were six workshops and three tutorials.

All full papers and short papers included in the proceedings were stringently peer-reviewed. Reflecting the strength of the ITS community, we received a large number of submissions of very high quality. The review process rested significantly on the outstanding team of international experts from 24 countries who made up the Program Committee, the Senior Program Committee and the Advisory Board. Reviewers started the process by bidding on abstracts, ensuring that they were reviewing in areas of their particular interest and expertise. Conflicts of interest were identified so that no paper was assigned to a reviewer who is a close collaborator or colleague of any of the paper's authors. Each paper received at least three reviews. One of the reviewers was a member of the Senior Program Committee, who was also responsible for leading an online discussion of the paper and then writing a meta-review. Criteria for reviews of papers were: relevance, originality, significance, evaluation, related work, organization and readability. The final decisions for acceptance were made by the Program Co-chairs who, working in concert, carefully studied the reviews, discussion and meta-reviews, often initiating additional discussion among reviewers. In some cases, we (the Program Co-chairs) sought additional reviewers. For the most difficult decisions, we also read the papers. In making the hard decisions on accepting full papers, we were largely driven by the reviews and meta-reviews. Where the scores were close, we took into account all review criteria, and in our final decision weighed the relative importance of a paper's strengths and weaknesses. We also considered the different classes of contributions: for example, a full paper describing a new system designed to improve learning should include a sound evaluation or at minimum a convincing pilot study. For short papers, the novelty and potential of the work were key requirements. Due to the large number of high-quality submissions, our choices were difficult. This is a very pleasing situation for the ITS community and augurs well for the future as some of the papers we could not accept have the promise to be excellent future publications.

The quality of the reviews was extremely high, which was critical in enabling us to distinguish the highest quality work for acceptance as full papers. In addition, high-quality reviews are critical for researchers as feedback on their research and their papers, regardless of whether they are accepted for publication or not. For example, many reviews pointed to additional relevant literature, identified particular strengths and gave concrete advice on how to address weaknesses. We believe that authors of many of the rejected papers will be able to use this feedback to produce excellent papers in the future. We worked very hard to select the Program Committee, the Senior Program Committee and the Advisory Board so we could meet these goals. We are pleased to announce the following Outstanding Reviewer Awards: Ivon Arroyo,

Kevin Ashley, Ryan Baker, Joseph Beck, Gautam Biswas, Sydney d'Mello, Peter Brusilovsky, Vania Dimitrova, Neil Heffernan, Akihiro Kashiara, Brent Martin, H. Chad Lane, James Lester, Diane Litman, Rose Luckin, Stellan Ohlsson, Niels Pinkwart, Steven Ritter, Ido Roll, Carolyn Rosé, Peter Sloep, John Stamper and Gerhard Weber.

A scientific conference of the size of ITS 2010 can only succeed due to contributions of many people who generously donate their time. Of great significance are the contributions of the large number of people who helped with the review process: the Advisory Board, the Senior Program Committee, the Program Committee, as well as people who volunteered as reviewers. We are extremely grateful to them for the time and effort they put in. Special thanks are due to the people who volunteered to organize workshops and tutorials, which made up a key part of the scientific program of the conference. We also thank the Chairs for Workshops / Tutorials, Young Researcher Track / Doctoral Consortium, Interactive Events, and Panels, all of whom had a major influence on the scientific program. The Local Arrangements Chairs devoted countless hours of preparation to make the conference actually happen successfully “on the ground.” The Volunteers / Outings Chairs recruited and organized dozens of students not only to help run the conference but to lead small-group outings tailored to individual interests in the ITS spirit. The Conference Treasurer organized our budget meticulously, the Sponsorship Chair increased it handsomely, and the Publicity Chair got the word out widely. Lynnetta Miller of Carnegie Mellon deserves special recognition for contributing in multiple guises (conference secretary, artist, webmaster). A special word of thanks is due to Carolyn Manley of Carnegie Mellon's Conference and Event Services, who among other things administered (along with programmer Alex Lang) the online registration system. We would like to thank Kevin Ashley, Vania Dimitrova, Ben du Boulay, Claude Frasson, Art Graesser, Alan Lesgold, James Lester, Roger Nkambou, Beverly Woolf, and other past organizers of ITS and AIED conferences for their kind assistance and sage advice. We are very grateful to Jo Bodnar of Carnegie Mellon and student volunteers Matthew Easterday, Richard Gluga, and Michael Lipschultz for the very significant role they played in assembling the proceedings. And we would like to thank our sponsors, listed later, whose support for the conference we gratefully acknowledge.

Our final thanks must be to the authors whose papers appear in these volumes. They have contributed many exciting new ideas and a comprehensive body of carefully validated work that will serve as an advanced technology bridge to improved learning in real educational settings.

April 2010

Vincent Alevan
Judy Kay
Jack Mostow

Organization

Conference Chair: Jack Mostow
Program Chairs: Vincent Aleven, Judy Kay
General Chair: Alan Lesgold
Conference Secretary: Lynnetta Miller
Conference Treasurer: Albert Corbett
Local Arrangements Chairs: Sandra Katz, Bruce McLaren
Workshops and Tutorials Chairs: Joe Beck, Niels Pinkwart
Young Researchers Track Chairs: Ricardo Conejo, Carolyn Penstein Rosé
Interactive Events Chairs: Noboru Matsuda, Tanja Mitrovic
Panels Chairs: Cristina Conati, Chee-Kit Looi
Publicity Chair: Susan Bull
Volunteers Chairs: Collin Lynch, Amy Ogan
Sponsorship Chair: Steve Ritter

Advisory Board

Bert Bredeweg	University of Amsterdam, The Netherlands
Claude Frasson	University of Montreal, Canada
Monique Grandbastien	Université Henri Poincaré, France
Lewis Johnson	University of Southern California, USA
Kenneth Koedinger	Carnegie Mellon University, USA
Gordon McCalla	University of Saskatchewan, Canada
Helen Pain	University of Edinburgh, UK
Beverly Woolf	University of Massachusetts, USA

Senior Program Committee

Esmá Aimeur	University of Montreal, Canada
Ivon Arroyo	University of Massachusetts, USA
Kevin Ashley	University of Pittsburgh, USA
Ryan Baker	Worcester Polytechnic Institute, USA
Beatriz Barros	University of Malaga, Spain
Joseph Beck	Worcester Polytechnic Institute, USA
Gautam Biswas	Vanderbilt University, USA
Paul Brna	University of Edinburgh, UK
Peter Brusilovsky	University of Pittsburgh, USA
Tak-Wai Chan	National Central University of Taiwan, Taiwan
Cristina Conati	University of British Columbia, Canada
Ulrike Cress	University of Tübingen, Germany
Vania Dimitrova	University of Leeds, UK

Benedict du Boulay	University of Sussex, UK
Art Graesser	University of Memphis, USA
Jim Greer	University of Saskatchewan, Canada
Peter Hastings	DePaul University, USA
Neil Heffernan	Worcester Polytechnic Institute, USA
Susanne Lajoie	McGill University, Canada
Chad Lane	University of Southern California, USA
James Lester	North Carolina State University, USA
Diane Litman	University of Pittsburgh, USA
Chee-Kit Looi	National Institute of Education, Singapore
Rosemary Luckin	University of Sussex, UK
Jean-Marc Labat	Université Pierre et Marie Curie, France
Brent Martin	University of Canterbury, New Zealand
Tanja Mitrovic	University of Canterbury, New Zealand
Riichiro Mizoguchi	University of Osaka, Japan
Rafael Morales	University of Guadalajara, Mexico
Wolfgang Nejdl	L3S and University of Hannover, Germany
Roger Nkambou	University of Quebec at Montreal, Canada
Niels Pinkwart	Clausthal University of Technology, Germany
Kaska Porayska-Pomsta	London Knowledge Lab, UK
Carolyn Rosé	Carnegie Mellon University, USA
Kurt Van Lehn	Arizona State University, USA
Julita Vassileva	University of Saskatchewan, Canada
Maria Virvou	University of Piraeus, Greece
Vincent Wade	Trinity College Dublin, Ireland
Kalina Yacef	University of Sydney, Australia

Program Committee

Ana Arruarte	University of the Basque Country, Spain
Roger Azevedo	University of Memphis, USA
Tiffany Barnes	University of North Carolina at Charlotte, USA
Mária Bielíková	Slovak University of Technology in Bratislava, Slovakia
Emmanuel Blanchard	McGill University, Canada
Steve Blessing	University of Tampa, USA
Jacqueline Bourdeau	Distance University, University of Quebec at Montreal, Canada
Nicola Capuano	University of Salerno, Italy
Zhi-Hong Chen	National Central University, Taiwan
Chih-Yueh Chou	Yuan Ze University, Taiwan
Scotty Craig	University of Memphis, USA
Alexandra Cristea	University of Warwick, UK
Richard Cox	University of Sussex, UK
Michel Desmarais	Polytechnique Montreal, Canada
Sydney D'Mello	University of Memphis, USA

Peter Dolog	Aalborg University, Denmark
Isabel Fernandez de Castro	University of Basque Country, Spain
Yusuke Hayashi	Osaka University, Japan
Tsukasa Hirashima	Hiroshima University, Japan
Pamela Jordan	University of Pittsburgh, USA
Akihiro Kashiara	The University of Electro-Communications, Japan
Chao-Lin Liu	National Chengchi University, Taiwan
Manolis Mavrikis	London Knowledge Lab, UK
Riccardo Mazza	University of Lugano/University of Applied Sciences of Southern Switzerland, Switzerland
Danielle McNamara	University of Memphis, USA
Erica Melis	German Artificial Intelligence Centre, Germany
Alessandro Micarelli	University of Rome, Italy
Kazuhisa Miwa	Nagoya University, Japan
Chas Murray	Carnegie Learning, Inc., USA
Stellan Ohlsson	University of Illinois at Chicago, USA
Ana Paiva	University of Lisbon, Portugal
Andrew Ravenscroft	London Metropolitan University, UK
Genaro Rebolledo-Mendez	University of Sussex, UK
Steve Ritter	Carnegie Learning, Inc., USA
Didith Rodrigo	Ateneo de Manila University, Philippines
Ido Roll	University of British Columbia, Canada
Sudeshna Sarkar	IIT Kharagpur, India
Yong Se Kim	Sungkyunkwan University, Republic of Korea
Mike Sharples	University of Nottingham, UK
Peter Sloep	Open University, The Netherlands
John Stamper	Carnegie Mellon University, USA
Leen-Kiat Soh	University of Nebraska-Lincoln, USA
Akira Takeuchi	Kyushu Institute of Technology, Japan
Pierre Tchounikine	University du Maine, France
Andre Tricot	University of Toulouse, France
Wouter van Joolingen	University of Twente, The Netherlands
Nicolas Vanlabeke	University of Nottingham, UK
Rosa Vicari	The Federal University of Rio Grande do Sul, Brazil
Gerhard Weber	University of Education Freiburg, Germany
Stephan Weibelzahl	National College of Ireland, Ireland
Lung-Hsiang Wong	Nanyang Technological University, Singapore
Diego Zapata-Rivera	Educational Testing Service, USA

Reviewers

Nilufar Baghaei	CSIRO ICT Centre, Australia
Quincy Brown	University of Maryland, USA
Mingyu Feng	SRI International, USA
Sylvie Girard	Université du Maine, France
Natalie Person	University of Rhodes, USA
Rachel Pilkington	University of Birmingham, UK

Leena Razzaq	University of Massachusetts, USA
Felisa Verdejo	National University of Distance Education, Spain
Ning Wang	University of Southern California, USA
Shumin Wu	IBM, USA

Workshops

Question Generation

Kristy Elizabeth Boyer and Paul Piwek

Culturally Aware Tutoring Systems

Emmanuel G. Blanchard, W. Lewis Johnson, Amy Ogan and Danièle Allard

Supporting the Social Inclusion of Communities with Adaptive Learning Media

Fabio Akhras and Paul Brna

Opportunities for Intelligent and Adaptive Behavior in Collaborative Learning Systems

Ari Bader-Natal, Erin Walker and Carolyn Rosé

Computer-Supported Peer Review in Education: Synergies with Intelligent Tutoring Systems

Ilya Goldin, Peter Brusilovsky, Christian Schunn, Kevin Ashley and I-Han Hsiao

Intelligent Tutoring Technologies for Ill-Defined Problems and Ill-Defined Domains

Collin Lynch, Kevin Ashley, Tanja Mitrovic, Vania Dimitrova, Niels Pinkwart and Vincent Aleven

Tutorials

Using DataShop to Analyze Educational Data

John Stamper

How to Apply Software Architecture and Patterns to Tutoring System Development?

Javier Gonzalez-Sanchez and Maria-Elena Chavez-Echeagaray

Seven Deadly Sins: Avoiding Several Common Statistical Pitfalls

Joseph Beck

Sponsors



Apangea Learning
Pittsburgh, PA
www.apangea.com



Carnegie Learning, Inc.
Pittsburgh, PA
www.carnegielearning.com



Edalytics, LLC
Pittsburgh
www.edalytics.com



Grockit
San Francisco, CA
grockit.com



Kaplan, Inc.
New York, NY
www.kaplan.com



Learning Research and Development Center
University of Pittsburgh
www.lrdc.pitt.edu



National Science Foundation
Division of Information and Intelligent Systems
Human-Centered Computing Program
Washington, DC
www.nsf.gov



Pittsburgh Science of Learning Center
www.learnlab.org



School of Computer Science
Carnegie Mellon University
www.cs.cmu.edu



School of Education
University of Pittsburgh
www.education.pitt.edu

Table of Contents – Part II

Affect 2

The Intricate Dance between Cognition and Emotion during Expert Tutoring	1
<i>Blair Lehman, Sidney D’Mello, and Natalie Person</i>	
Subliminally Enhancing Self-esteem: Impact on Learner Performance and Affective State	11
<i>Imène Jraïdi and Claude Frasson</i>	
Detecting Learner Frustration: Towards Mainstream Use Cases	21
<i>Judi McCuaig, Mike Pearlstein, and Andrew Judd</i>	

Educational Data Mining 2

Enhancing the Automatic Generation of Hints with Expert Seeding	31
<i>John Stamper, Tiffany Barnes, and Marvin Croy</i>	
Learning What Works in ITS from Non-traditional Randomized Controlled Trial Data	41
<i>Zachary A. Pardos, Matthew D. Dailey, and Neil T. Heffernan</i>	

Natural Language Interaction 2

Persuasive Dialogues in an Intelligent Tutoring System for Medical Diagnosis	51
<i>Amin Rahati and Froduald Kabanza</i>	
Predicting Student Knowledge Level from Domain-Independent Function and Content Words	62
<i>Claire Williams and Sidney D’Mello</i>	
KSC-PaL: A Peer Learning Agent	72
<i>Cynthia Kersey, Barbara Di Eugenio, Pamela Jordan, and Sandra Katz</i>	

Authoring Tools and Theoretical Synthesis

Transforming a Linear Module into an Adaptive One: Tackling the Challenge	82
<i>Jonathan G.K. Foss and Alexandra I. Cristea</i>	

An Authoring Tool to Support the Design and Use of Theory-Based Collaborative Learning Activities 92
Seiji Isotani, Rūichiro Mizoguchi, Sadao Isotani, Olimpio M. Capeli, Naoko Isotani, and Antonio R.P.L. de Albuquerque

How to Build Bridges between Intelligent Tutoring System Subfields of Research 103
Philip Pavlik Jr. and Joe Toth

Collaborative and Group Learning 2

Recognizing Dialogue Content in Student Collaborative Conversation . . . 113
Toby Dragon, Mark Floryan, Beverly Woolf, and Tom Murray

Supporting Learners’ Self-organization: An Exploratory Study 123
Patrice Moguel, Pierre Tchounikine, and André Tricot

Exploring the Effectiveness of Social Capabilities and Goal Alignment in Computer Supported Collaborative Learning 134
Hua Ai, Rohit Kumar, Dong Nguyen, Amrut Nagasunder, and Carolyn P. Rosé

Intelligent Games 2

Virtual Humans with Secrets: Learning to Detect Verbal Cues to Deception 144
H. Chad Lane, Mike Schneider, Stephen W. Michael, Justin S. Albrechtsen, and Christian A. Meissner

Optimizing Story-Based Learning: An Investigation of Student Narrative Profiles 155
Seung Y. Lee, Bradford W. Mott, and James C. Lester

Integrating Learning and Engagement in Narrative-Centered Learning Environments 166
Jonathan P. Rowe, Lucy R. Shores, Bradford W. Mott, and James C. Lester

Intelligent Tutoring and Scaffolding 2

Collaborative Lecturing by Human and Computer Tutors 178
Sidney D’Mello, Patrick Hays, Claire Williams, Whitney Cade, Jennifer Brown, and Andrew Olney

Computational Workflows for Assessing Student Learning 188
Jun Ma, Erin Shaw, and Jihie Kim

Predictors of Transfer of Experimental Design Skills in Elementary and Middle School Children	198
<i>Stephanie Siler, David Klahr, Cressida Magaro, Kevin Willows, and Dana Mowery</i>	

Young Researchers Track

Moodle Discussion Forum Analyzer Tool (DFAT)	209
<i>Palak Baid, Hui Soo Chae, Faisal Anwar, and Gary Natriello</i>	
Peer-Based Intelligent Tutoring Systems: A Corpus-Oriented Approach	212
<i>John Champaign and Robin Cohen</i>	
Intelligent Tutoring Systems, Educational Data Mining, and the Design and Evaluation of Video Games	215
<i>Michael Eagle and Tiffany Barnes</i>	
An Intelligent Debater for Teaching Argumentation	218
<i>Matthew W. Easterday</i>	
Multiple Interactive Representations for Fractions Learning	221
<i>Laurens Feenstra, Vincent Alevén, Nikol Rummel, and Niels Taatgen</i>	
An Interactive Educational Diagrammatic System for Assessing and Remediating the Graph-as-Picture Misconception	224
<i>Grecia Garcia Garcia and Richard Cox</i>	
Long Term Student Learner Modeling and Curriculum Mapping	227
<i>Richard Gluga</i>	
Student Dispositions and Help-Seeking in Collaborative Learning	230
<i>Iris K. Howley and Carolyn Penstein Rosé</i>	
Visualizing Educational Data from Logic Tutors	233
<i>Matthew Johnson and Tiffany Barnes</i>	
An Authoring Language as a Key to Usability in a Problem-Solving ITS Framework	236
<i>Jean-François Lebeau, Luc Paquette, Mikaël Fortin, and André Mayers</i>	
Towards the Creation of a Data-Driven Programming Tutor	239
<i>Behrooz Mostafavi and Tiffany Barnes</i>	
Using Expert Models to Provide Feedback on Clinical Reasoning Skills	242
<i>Laura Naismith and Susanne P. Lajoie</i>	

Algorithms for Robust Knowledge Extraction in Learning Environments	245
<i>Ifeyinwa Okoye, Keith Maull, and Tamara Sumner</i>	
Integrating Sophisticated Domain-Independent Pedagogical Behaviors in an ITS Framework	248
<i>Luc Paquette, Jean-François Lebeau, and André Mayers</i>	
Delivering Tutoring Feedback Using Persuasive Dialogues	251
<i>Amin Rahati and Froduald Kabanza</i>	
Coordinate Geometry Learning Environment with Game-Like Properties	254
<i>Dovan Rai, Joseph E. Beck, and Neil T. Heffernan</i>	
Long-Term Benefits of Direct Instruction with Reification for Learning the Control of Variables Strategy	257
<i>Michael A. Sao Pedro, Janice D. Gobert, and Juelaila J. Raziuddin</i>	
Short Papers	
Can Affect Be Detected from Intelligent Tutoring System Interaction Data? – A Preliminary Study	260
<i>Elizabeth A. Anglo and Ma. Mercedes T. Rodrigo</i>	
Comparing Disengaged Behavior within a Cognitive Tutor in the USA and Philippines	263
<i>Ma. Mercedes T. Rodrigo, Ryan S.J.d. Baker, Jenilyn Agapito, Julieta Nabos, Ma. Concepcion Repalam, and Salvador S. Reyes Jr.</i>	
Adaptive Tutorials for Virtual Microscopy: A Design Paradigm to Promote Pedagogical Ownership	266
<i>Dror Ben-Naim, Gary Velan, Nadine Marcus, and Michael Bain</i>	
The Online Deteriorating Patient: An Adaptive Simulation to Foster Expertise in Emergency Decision-Making	269
<i>Emmanuel G. Blanchard, Jeffrey Wiseman, Laura Naismith, Yuan-Jin Hong, and Susanne P. Lajoie</i>	
DynaLearn: Architecture and Approach for Investigating Conceptual System Knowledge Acquisition	272
<i>Bert Bredeweg, Jochem Liem, Floris Linnebank, René Bühling, Michael Wißner, Jorge Gracia del Río, Paulo Salles, Wouter Beek, and Asunción Gómez Pérez</i>	
Interfaces for Inspectable Learner Models	275
<i>Susan Bull, Andrew Mabbott, Rasyidi Johan, Matthew Johnson, Kris Lee-Shim, and Tim Lloyd</i>	

Conceptual Personalization Technology: Promoting Effective Self-directed, Online Learning	278
<i>Kirsten R. Butcher, Tamara Sumner, Keith Maull, and Ifeyinwa Okoye</i>	
Learning to Identify Students' Relevant and Irrelevant Questions in a Micro-blogging Supported Classroom	281
<i>Suleyman Cetintas, Luo Si, Sugato Chakravarty, Hans Aagard, and Kyle Bowen</i>	
Using Emotional Coping Strategies in Intelligent Tutoring Systems	285
<i>Soumaya Chaffar and Claude Frasson</i>	
Showing the Positive Influence of Subliminal Cues on Learner's Performance and Intuition: An ERP Study	288
<i>Pierre Chalfoun and Claude Frasson</i>	
Exploring the Relationship between Learner EEG Mental Engagement and Affect	291
<i>Maher Chaouachi and Claude Frasson</i>	
MiBoard: Creating a Virtual Environment from a Physical Environment	294
<i>Kyle Dempsey, G. Tanner Jackson, and Danielle S. McNamara</i>	
Players' Motivation and EEG Waves Patterns in a Serious Game Environment	297
<i>Lotfi Derbali and Claude Frasson</i>	
Predicting the Effects of Skill Model Changes on Student Progress	300
<i>Daniel Dickison, Steven Ritter, Tristan Nixon, Thomas K. Harris, Brendon Towle, R. Charles Murray, and Robert G.M. Hausmann</i>	
Data Mining to Generate Individualised Feedback	303
<i>Anna Katrina Dominguez, Kalina Yacef, and James R. Curran</i>	
In the Zone: Towards Detecting Student Zoning Out Using Supervised Machine Learning	306
<i>Joanna Drummond and Diane Litman</i>	
Can We Get Better Assessment from a Tutoring System Compared to Traditional Paper Testing? Can We Have Our Cake (Better Assessment) and Eat It too (Student Learning during the Test)?	309
<i>Mingyu Feng and Neil Heffernan</i>	
Using Data Mining Findings to Aid Searching for Better Cognitive Models	312
<i>Mingyu Feng, Neil T. Heffernan, and Kenneth Koedinger</i>	

Generating Proactive Feedback to Help Students Stay on Track	315
<i>Davide Fossati, Barbara Di Eugenio, Stellan Ohlsson, Christopher Brown, and Lin Chen</i>	
ITS in Ill-Defined Domains: Toward Hybrid Approaches	318
<i>Philippe Fournier-Viger, Roger Nkambou, Engelbert Mephu Nguifo, and André Mayers</i>	
Analyzing Student Gaming with Bayesian Networks	321
<i>Stephen Giguere, Joseph Beck, and Ryan Baker</i>	
EdiScenE: A System to Help the Design of Online Learning Activities	324
<i>Patricia Gounon and Pascal Leroux</i>	
Critiquing Media Reports with Flawed Scientific Findings: <i>Operation ARIES!</i> A Game with Animated Agents and Natural Language Dialogues	327
<i>Art Graesser, Anne Britt, Keith Millis, Patty Wallace, Diane Halpern, Zhiqiang Cai, Kris Kopp, and Carol Forsyth</i>	
A Case-Based Reasoning Approach to Provide Adaptive Feedback in Microworlds	330
<i>Sergio Gutierrez-Santos, Mihaela Cocea, and George Magoulas</i>	
Real-Time Control of a Remote Virtual Tutor Using Minimal Pen-Gestures	334
<i>Yonca Haciahmetoglu and Francis Quek</i>	
Theoretical Model for Interplay between Some Learning Situations and Brainwaves	337
<i>Alicia Heraz and Claude Frasson</i>	
Cultural Adaptation of Pedagogical Resources within Intelligent Tutorial Systems	340
<i>Franck Hervé Mpondo Eboa, François Courtemanche, and Esma Aïmeur</i>	
An Interactive Learning Environment for Problem-Changing Exercise . . .	343
<i>Tsukasa Hirashima, Sho Yamamoto, and Hiromi Waki</i>	
Towards Intelligent Tutoring with Erroneous Examples: A Taxonomy of Decimal Misconceptions	346
<i>Seiji Isotani, Bruce M. McLaren, and Max Altman</i>	
The Efficacy of iSTART Extended Practice: Low Ability Students Catch Up	349
<i>G. Tanner Jackson, Chutima Boonthum, and Danielle S. McNamara</i>	

Expecting the Unexpected: Warehousing and Analyzing Data from ITS Field Use	352
<i>W. Lewis Johnson, Naveen Ashish, Stephen Bodnar, and Alicia Sagae</i>	
Developing an Intelligent Tutoring System Using Natural Language for Knowledge Representation	355
<i>Sung-Young Jung and Kurt VanLehn</i>	
A Network Analysis of Student Groups in Threaded Discussions	359
<i>Jeon-Hyung Kang, Jihie Kim, and Erin Shaw</i>	
A New Framework of Metacognition with Abstraction/Instantiation Operations	362
<i>Michiko Kayashima and Riichiro Mizoguchi</i>	
Expansion of the xPST Framework to Enable Non-programmers to Create Intelligent Tutoring Systems in 3D Game Environments	365
<i>Sateesh Kumar Kodavali, Stephen Gilbert, and Stephen B. Blessing</i>	
A Computational Model of Accelerated Future Learning through Feature Recognition	368
<i>Nan Li, William W. Cohen, and Kenneth R. Koedinger</i>	
Automated and Flexible Comparison of Course Sequencing Algorithms in the LS-Lab Framework	371
<i>Carla Limongelli, Filippo Sciarrone, Marco Temperini, and Giulia Vaste</i>	
Correcting Scientific Knowledge in a General-Purpose Ontology	374
<i>Michael Lipschultz and Diane Litman</i>	
Learning to Argue Using Computers – A View from Teachers, Researchers, and System Developers	377
<i>Frank Loll, Oliver Scheuer, Bruce M. McLaren, and Niels Pinkwart</i>	
How to Take into Account Different Problem Solving Modalities for Doing a Diagnosis? Experiment and Results	380
<i>Sandra Michelet, Vanda Luengo, Jean-Michel Adam, and Nadine Madran</i>	
Behavior Effect of Hint Selection Penalties and Availability in an Intelligent Tutoring System	384
<i>Pedro J. Muñoz-Merino, Carlos Delgado Kloos, and Mario Muñoz-Organero</i>	
DesignWebs: A Tool for Automatic Construction of Interactive Conceptual Maps from Document Collections	387
<i>Sharad V. Oberoi, Dong Nguyen, Gahgene Gweon, Susan Finger, and Carolyn Penstein Rosé</i>	

Extraction of Concept Maps from Textbooks for Domain Modeling	390
<i>Andrew M. Olney</i>	
Levels of Interaction (LoI): A Model for Scaffolding Learner Engagement in an Immersive Environment	393
<i>David Panzoli, Adam Qureshi, Ian Dunwell, Panagiotis Petridis, Sara de Freitas, and Genaro Rebolledo-Mendez</i>	
Tools for Acquiring Data about Student Work in Interactive Learning Environment T-Algebra	396
<i>Rein Prank and Dmitri Lepp</i>	
Mily’s World: A Coordinate Geometry Learning Environment with Game-Like Properties	399
<i>Dovan Rai, Joseph E. Beck, and Neil T. Heffernan</i>	
An Intelligent Tutoring System Supporting Metacognition and Sharing Learners’ Experiences	402
<i>Triomphe Ramandalahy, Philippe Vidal, and Julien Broisin</i>	
Are ILEs Ready for the Classroom? Bringing Teachers into the Feedback Loop	405
<i>James Segedy, Brian Sulcer, and Gautam Biswas</i>	
Comparison of a Computer-Based to Hands-On Lesson in Experimental Design	408
<i>Stephanie Siler, Dana Mowery, Cressida Magaro, Kevin Willows, and David Klahr</i>	
Toward the Development of an Intelligent Tutoring System for Distributed Team Training through Passive Sensing	411
<i>Robert A. Sottolare</i>	
Open Educational Resource Assessments (OPERA)	414
<i>Tamara Sumner, Kirsten Butcher, and Philipp Wetzler</i>	
Annie: A Tutor That Works in Digital Games	417
<i>James M. Thomas and R. Michael Young</i>	
Learning from Erroneous Examples	420
<i>Dimitra Tsovaltzi, Bruce M. McLaren, Erica Melis, Ann-Kristin Meyer, Michael Dietrich, and George Goguardze</i>	
Feasibility of a Socially Intelligent Tutor	423
<i>Jozef Tvarožek and Mária Bielíková</i>	
Agent Prompts: Scaffolding Students for Productive Reflection in an Intelligent Learning Environment	426
<i>Longkai Wu and Chee-Kit Looi</i>	

Identifying Problem Localization in Peer-Review Feedback	429
<i>Wenting Xiong and Diane Litman</i>	
AlgoTutor: From Algorithm Design to Coding	432
<i>Sung Yoo and Jungsoon Yoo</i>	
Adaptive, Assessment-Based Educational Games	435
<i>Diego Zapata-Rivera</i>	

Interactive Events

ITS Authoring through Programming-by-Demonstration	438
<i>Vincent Alevan, Brett Leber, and Jonathan Sewall</i>	
A Coordinate Geometry Learning Environment with Game-Like Properties	439
<i>Dovan Rai, Joseph E. Beck, and Neil T. Heffernan</i>	
Adaptive Tutorials and the Adaptive eLearning Platform	440
<i>Dror Ben-Naim</i>	
DomainBuilder – An Authoring System for Visual Classification Tutoring Systems	441
<i>Eugene Tseytlin, Melissa Castine, and Rebecca Crowley</i>	
AWESOME Computing: Using Corpus Data to Tailor a Community Environment for Dissertation Writing	443
<i>Vania Dimitriva, Royce Neagle, Sirisha Bajanki, Lydia Lau, and Roger Boyle</i>	
Collaboration and Content Recognition Features in an Inquiry Tutor . . .	444
<i>Mark Floryan, Toby Dragon, Beverly Woolf, and Tom Murray</i>	
The Science Assistments Project: Scaffolding Scientific Inquiry Skills . . .	445
<i>Janice D. Gobert, Orlando Montalvo, Ermal Toto, Michael A. Sao Pedro, and Ryan S.J.d. Baker</i>	
Incorporating Interactive Examples into the Cognitive Tutor	446
<i>Robert G.M. Hausmann, Steven Ritter, Brendon Towle, R. Charles Murray, and John Connelly</i>	
iGeom: Towards an Interactive Geometry Software with Intelligent Guidance Capabilities	447
<i>Leônidas O. Brandão, Seiji Isotani, and Danilo L. Dalmon</i>	
Acquiring Conceptual Knowledge about How Systems Behave	448
<i>Jochem Liem, Bert Bredeweg, Floris Linnebank, René Bühling, Michael Wißner, Jorge Gracia del Río, Wouter Beek, and Asunción Gómez Pérez</i>	

Learning by Teaching SimStudent	449
<i>Noboru Matsuda, Victoria Keiser, Rohan Raizada, Gabriel Stylianides, William W. Cohen, and Ken Koedinger</i>	
Authoring Problem-Solving ITS with ASTUS	450
<i>Jean-François Lebeau, Luc Paquette, and André Mayers</i>	
A Better Reading Tutor That Listens	451
<i>Jack Mostow, Greg Aist, Juliet Bey, Wei Chen, Al Corbett, Weisi Duan, Nell Duke, Minh Duong, Donna Gates, José P. González, Octavio Juarez, Martin Kantorzyk, Yuanpeng Li, Liu Liu, Margaret McKeown, Christina Trotochaud, Joe Valeri, Anders Weinstein, and David Yen</i>	
Research-Based Improvements in Cognitive Tutor Geometry	452
<i>Steven Ritter, Brendon Towle, R. Charles Murray, Robert G.M. Hausmann, and John Connelly</i>	
A Cognitive Tutor for Geometric Proof	453
<i>Steven Ritter, Brendon Towle, R. Charles Murray, Robert G.M. Hausmann, and John Connelly</i>	
Multiplayer Language and Culture Training in ISLET	454
<i>Kevin Saunders and W. Lewis Johnson</i>	
PSLC DataShop: A Data Analysis Service for the Learning Science Community	455
<i>John Stamper, Ken Koedinger, Ryan S.J.d. Baker, Alida Skogsholm, Brett Leber, Jim Rankin, and Sandy Demi</i>	
A DIY Pressure Sensitive Chair for Intelligent Tutoring Systems	456
<i>Andrew M. Olney and Sidney D’Mello</i>	
Author Index	457

Table of Contents – Part I

Invited Talks

Can Research-Based Technology Change School-Based Learning? Perspectives from Singapore	1
<i>Chee-Kit Looi</i>	
Modeling Emotion and Its Expression	2
<i>Stacy Marsella</i>	
Active Learning in Technology-Enhanced Environments: On Sensible and Less Sensible Conceptions of “Active” and Their Instructional Consequences	3
<i>Alexander Renkl</i>	
Riding the Third Wave	4
<i>Steven Ritter</i>	
Social and Caring Tutors: ITS 2010 Keynote Address	5
<i>Beverly Park Woolf</i>	

Educational Data Mining 1

Predicting Correctness of Problem Solving in ITS with a Temporal Collaborative Filtering Approach	15
<i>Suleyman Cetintas, Luo Si, Yan Ping Xin, and Casey Hord</i>	
Detecting the Moment of Learning	25
<i>Ryan S.J.d. Baker, Adam B. Goldstein, and Neil T. Heffernan</i>	
Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures	35
<i>Yue Gong, Joseph E. Beck, and Neil T. Heffernan</i>	

Natural Language Interaction 1

Automatic Question Generation for Literature Review Writing Support	45
<i>Ming Liu, Rafael A. Calvo, and Vasile Rus</i>	
Characterizing the Effectiveness of Tutorial Dialogue with Hidden Markov Models	55
<i>Kristy Elizabeth Boyer, Robert Phillips, Amy Ingram, Eun Young Ha, Michael Wallis, Mladen Vouk, and James Lester</i>	

Exploiting Predictable Response Training to Improve Automatic Recognition of Children’s Spoken Responses 65
Wei Chen, Jack Mostow, and Gregory Aist

ITS in Ill-Defined Domains

Leveraging a Domain Ontology to Increase the Quality of Feedback in an Intelligent Tutoring System 75
Hameedullah Kazi, Peter Haddawy, and Siriwan Suebnukarn

Modeling Long Term Learning of Generic Skills 85
Richard Gluga, Judy Kay, and Tim Lever

Eliciting Informative Feedback in Peer Review: Importance of Problem-Specific Scaffolding 95
Ilya M. Goldin and Kevin D. Ashley

Inquiry Learning

Layered Development and Evaluation for Intelligent Support in Exploratory Environments: The Case of Microworlds 105
Sergio Gutierrez-Santos, Manolis Mavrikis, and George Magoulas

The Invention Lab: Using a Hybrid of Model Tracing and Constraint-Based Modeling to Offer Intelligent Support in Inquiry Environments 115
Ido Roll, Vincent Aleven, and Kenneth R. Koedinger

Discovering and Recognizing Student Interaction Patterns in Exploratory Learning Environments 125
Andrea Bernardini and Cristina Conati

Collaborative and Group Learning 1

Lesson Study Communities on Web to Support Teacher Collaboration for Professional Development 135
Yukari Kato and Masatoshi Ishikawa

Using Problem-Solving Context to Assess Help Quality in Computer-Mediated Peer Tutoring 145
Erin Walker, Sean Walker, Nikol Rummel, and Kenneth R. Koedinger

Socially Capable Conversational Tutors Can Be Effective in Collaborative Learning Situations 156
Rohit Kumar, Hua Ai, Jack L. Beuth, and Carolyn P. Rosé

Intelligent Games 1

Facial Expressions and Politeness Effect in Foreign Language Training System	165
<i>Ning Wang, W. Lewis Johnson, and Jonathan Gratch</i>	
Intercultural Negotiation with Virtual Humans: The Effect of Social Goals on Gameplay and Learning	174
<i>Amy Ogan, Vincent Aleven, Julia Kim, and Christopher Jones</i>	

Gaming the System

An Analysis of Gaming Behaviors in an Intelligent Tutoring System	184
<i>Kasia Muldner, Winslow Burlison, Brett Van de Sande, and Kurt VanLehn</i>	
The Fine-Grained Impact of Gaming (?) on Learning	194
<i>Yue Gong, Joseph E. Beck, Neil T. Heffernan, and Elijah Forbes-Summers</i>	
Squeezing Out Gaming Behavior in a Dialog-Based ITS	204
<i>Peter Hastings, Elizabeth Arnott-Hill, and David Allbritton</i>	

Pedagogical Strategies 1

Analogies, Explanations, and Practice: Examining How Task Types Affect Second Language Grammar Learning	214
<i>Ruth Wylie, Kenneth R. Koedinger, and Teruko Mitamura</i>	
Do Micro-Level Tutorial Decisions Matter: Applying Reinforcement Learning to Induce Pedagogical Tutorial Tactics	224
<i>Min Chi, Kurt VanLehn, and Diane Litman</i>	
Examining the Role of Gestures in Expert Tutoring	235
<i>Betsy Williams, Claire Williams, Nick Volgas, Brian Yuan, and Natalie Person</i>	

Affect 1

A Time for Emoting: When Affect-Sensitivity Is and Isn't Effective at Promoting Deep Learning	245
<i>Sidney D'Mello, Blair Lehman, Jeremiah Sullins, Rosaire Daigle, Rebekah Combs, Kimberly Vogt, Lydia Perkins, and Art Graesser</i>	
The Affective and Learning Profiles of Students Using an Intelligent Tutoring System for Algebra	255
<i>Maria Carminda V. Lagud and Ma. Mercedes T. Rodrigo</i>	

The Impact of System Feedback on Learners’ Affective and Physiological States	264
<i>Payam Aghaei Pour, M. Sazzad Hussain, Omar AlZoubi, Sidney D’Mello, and Rafael A. Calvo</i>	

Games and Augmented Reality

Investigating the Relationship between Presence and Learning in a Serious Game	274
<i>H. Chad Lane, Matthew J. Hays, Daniel Auerbach, and Mark G. Core</i>	

Developing Empirically Based Student Personality Profiles for Affective Feedback Models	285
<i>Jennifer Robison, Scott McQuiggan, and James Lester</i>	

Evaluating the Usability of an Augmented Reality Based Educational Application	296
<i>Jorge Martín-Gutiérrez, Manuel Contero, and Mariano Alcañiz</i>	

Pedagogical Agents, Learning Companions, and Teachable Agents

What Do Children Favor as Embodied Pedagogical Agents?	307
<i>Sylvie Girard and Hilary Johnson</i>	

Learning by Teaching SimStudent: Technical Accomplishments and an Initial Use with Students	317
<i>Noboru Matsuda, Victoria Keiser, Rohan Raizada, Arthur Tu, Gabriel Stylianides, William W. Cohen, and Kenneth R. Koedinger</i>	

The Effect of Motivational Learning Companions on Low Achieving Students and Students with Disabilities	327
<i>Beverly Park Woolf, Ivon Arroyo, Kasia Muldner, Winslow Bursleson, David G. Cooper, Robert Dolan, and Robert M. Christopherson</i>	

Intelligent Tutoring and Scaffolding 1

Use of a Medical ITS Improves Reporting Performance among Community Pathologists	338
<i>Rebecca Crowley, Dana Grzybicki, Elizabeth Legowski, Lynn Wagner, Melissa Castine, Olga Medvedeva, Eugene Tseytlin, Drazen Jukic, and Stephen Raab</i>	

Hints: Is It Better to Give or Wait to Be Asked?	349
<i>Leena Razzaq and Neil T. Heffernan</i>	

Error-Flagging Support for Testing and Its Effect on Adaptation	359
<i>Amruth N. Kumar</i>	

Metacognition

Emotions and Motivation on Performance during Multimedia Learning: How Do I Feel and Why Do I Care?	369
<i>Amber Chauncey and Roger Azevedo</i>	
Metacognition and Learning in Spoken Dialogue Computer Tutoring . . .	379
<i>Kate Forbes-Riley and Diane Litman</i>	
A Self-regulator for Navigational Learning in Hyperspace	389
<i>Akihiro Kashihara and Ryoya Kawai</i>	

Pedagogical Strategies 2

How Adaptive Is an Expert Human Tutor?	401
<i>Micheline T.H. Chi and Marguerite Roy</i>	
Blocked versus Interleaved Practice with Multiple Representations in an Intelligent Tutoring System for Fractions	413
<i>Martina A. Rau, Vincent Alevan, and Nikol Rummel</i>	
Improving Math Learning through Intelligent Tutoring and Basic Skills Training	423
<i>Ivon Arroyo, Beverly Park Woolf, James M. Royer, Minghui Tai, and Sara English</i>	
Author Index	433

The Intricate Dance between Cognition and Emotion during Expert Tutoring

Blair Lehman¹, Sidney D'Mello¹, and Natalie Person²

¹Institute for Intelligent Systems, University of Memphis, Memphis, TN 38152
{balehman, sdmello}@memphis.edu

²Department of Psychology, Rhodes College, Memphis, TN 38112
person@rhodes.edu

Abstract. Although, many have theorized about the link between cognition and affect and its potential importance in complex tasks such as problem solving and deep learning, this link has seldom been explicitly investigated during tutoring. Consequently, this paper investigates the relationship between learners' cognitive and affective states during 50 tutoring sessions with expert human tutors. Association rule mining analyses revealed significant co-occurrence relationships between several of the cognitive measures (i.e., student answer types, question types, misconceptions, and metacomments) and the affective states of confusion, frustration, and anxiety, but not happiness. We also derived a number of association rules (Cognitive State \rightarrow Affective State) from the co-occurrence relationships. We discuss the implications of our findings for theories that link affect and cognition during learning and for the development of affect-sensitive ITSs.

Keywords: affect, cognition, confusion, frustration, expert tutoring, ITSs.

1 Introduction

Cognition and emotion have historically been considered to be distinct, separate processes [1], yet decades of scientific research have indicated that the two are inextricably linked [2-5]. The scientific research indicates that emotion and cognition are complimentary processes in learning environments that require students to generate inferences, answer causal questions, diagnose and solve problems, make conceptual comparisons, generate coherent explanations, and demonstrate application and transfer of acquired knowledge. Contemporary theories of emotion and cognition assume that cognitive processes such as memory encoding and retrieval, causal reasoning, deliberation, goal appraisal, and planning operate continually throughout the experience of emotion [2, 5-9]. The intricate relation between emotion and cognition is sufficiently compelling that some claim the scientific distinction between emotion and cognition to be artificial, arbitrary, and of limited value [4]. Hence, a cognitively demanding, complex learning task would be best understood with an approach that monitors both the cognitive as well as the affective states of learners.

Important insights into the link between affect and cognition during learning can be gleaned from theoretical perspectives that highlight the importance of cognitive

disequilibrium and goal appraisal processes during learning. Cognitive disequilibrium theory [10-13] proposes that when students encounter an impasse [14], they enter a state of cognitive disequilibrium, which is presumably accompanied by confusion. Students then begin a process of effortful problem solving in order to restore equilibrium. Hence, this theory proposes a direct connection between the student’s cognitive state and their affective experience. In addition to confusion being hypothesized to occur when an impasse is detected, persistent failure to resolve the impasse might be accompanied by frustration, while delight or happiness might occur if the impasse is resolved and an important goal is achieved.

Goal-appraisal theory postulates that emotions are triggered by events that facilitate or block achieving goals [7]. The availability of a plan to continue in the decision-making process differentiates between different types of inhibitory events. It is then that a cognitive appraisal [8] of the current situation elicits a particular affective state. Similar to cognitive disequilibrium theory, events that facilitate achieving a goal elicit happiness. On the other hand, events which block the achievement of a goal will elicit frustration, anger, or sadness depending on the ability to formulate a plan to overcome the current obstacle.

While cognitive disequilibrium and goal-appraisal theories focus on the occurrence states such as confusion, frustration, delight, etc, state-trait anxiety theory [15] focuses on the impact of high and low anxiety on cognitive states and performance outcomes. High anxiety can improve performance on simple tasks but drastically reduces performance on complex tasks. The combination of high anxiety and a complex task reduces performance due to the person’s inability to differentiate between the myriad of options present. Thus, state-trait anxiety theory would generally predict that the presence of anxiety would be linked to decreased performance.

Although the affect-cognition link has been alluded to by these theories, the theories link affect and cognition somewhat generally. For example, the highly influential network theories pioneered by Bower and Isen that emphasize the important role of mood states (positive, negative, or neutral) on creative problem solving. In particular, flexibility, creative thinking, and efficient decision-making in problem solving have been linked to experiences of positive affect [16,17], while negative affect has been associated with a more methodical approach to assessing the problem and finding the solution [18,19].

Network theories, however, do not explicitly address the intricate dance between cognition and affect during complex learning. The present paper explores this relationship during one-to-one expert tutoring sessions [20]. By investigating simultaneous occurrences of cognitive and affective states, we hope to obtain a better understanding of the affect-cognition link during learning and to apply this basic research towards the development of affect-sensitive ITSs, that is, ITSs that are sensitive to learners’ cognitive and affective states [21-25].

2 Expert Tutoring Corpus

The corpus consisted of 50 tutoring sessions between ten expert tutors and 39 students. Expert status was defined as: licensed at the secondary level, five or more years

of ongoing tutoring experience, employed by a professional tutoring agency, and highly recommended by local school personnel. The students were all having difficulty in a science or math course and were either recommended for tutoring by school personnel or voluntarily sought professional tutoring help. Fifty-five percent of students were female and 45% were male. Each session lasted approximately one hour. All sessions were videotaped with a camera positioned at a great enough distance to not disturb the tutoring session but close enough to record audio and visual data. The researcher left the room during the tutoring session. The videos were digitized and then transcribed. Transcripts were then coded with respect to tutor moves (not described here), student dialogue moves, and student affective states.

Student affective states were coded by two trained judges. Although 12 affective states (anger, anxiety, confusion, contempt, curiosity, disgust, eureka, fear, frustration, happiness, sadness, surprise) were coded, four were found to be the most prominent in the expert tutoring sessions (*anxiety, confusion, frustration, & happiness*) [26]. Affective states were defined as visible changes in affect based on facial expressions, paralinguistic changes, and gross body movements lasting from one to three seconds. Proportional occurrences for these states were, 0.221, 0.346, 0.038, 0.307, respectively. They accounted for 91.2% of the emotions that students experienced during the expert tutoring sessions. Cohen's kappas between the judges were .68, .65, .72, and .80 for *anxiety, confusion, frustration, and happiness*, respectively.

A 16-item coding scheme was derived to code student dialogue moves [20]. Of relevance to the current paper is a subset of dialogue moves that represented student cognitive states. Cognitive states were bounded by associated dialogue moves, thus the length of cognitive states was variable. The included dialogue moves pertaining to student answer types, question types, and metacognition. Answer types were classified as *correct* ("In meiosis it starts out the same with 1 diploid"), *partially correct* ("It has to do with cells"), *vague* ("Because it helps to, umm, you know"), *error-ridden* ("Prokaryotes are human and eukaryotes are bacteria"), and *no answers* ("Umm"). Question types were separated into two categories: *knowledge deficit* ("What do you mean by it doesn't have a skeleton?") and *common ground questions* ("Aren't they more lined up, like more in order?"). Finally metacognition occurred when students verbalized a previously held *misconception* ("I always used to get diploid and haploid mixed up") or directly made *metacomments* ("I don't know" or "Yes, I understand"). Student dialogue moves were coded by four trained judges, with a kappa of .88.

3 Results and Discussion

Association rule mining analyses [27] were used to identify co-occurrences between cognitive and affective states and to extract association rules that could conditionally detect the presence of an affective state from a cognitive state. Association rules are probabilistic in nature and take the form *Antecedent* \rightarrow *Consequent* [*support, confidence*]. The antecedent is a cognitive state or a set of cognitive states whose occurrence predicts the occurrence of the consequent (a set of affective states). The support of a rule measures its usefulness and is the probability that the antecedent (A) and the

consequent (C) occur simultaneously. The confidence is the conditional probability that the consequent will occur if the antecedent occurs.

For example, we observed an association rule where *error-ridden answers* predict *confusion* (Error-Ridden \rightarrow Confusion). Here, the *error-ridden answer* is the antecedent and *confusion* is the consequent. The support of the rule is expressed as 0.002 ($P[\text{Error}, \text{Confusion}]$), which is the proportion of dialogue moves containing both *error-ridden answers* and *confusion*. If, 0.028 is the proportion of moves containing *error-ridden answers* ($P[\text{Error}]$), then the confidence of the association is 0.071 ($P[\text{Error}, \text{Confusion}] / P[\text{Error}]$).

The process of mining association rules can be decomposed into two steps. First, we identify cognitive and affective states that co-occur. Second, the association rules are derived from the frequently occurring cognitive-affective amalgamations. The results from each of these phases are described below.

Before describing the results, it is important to emphasize one important distinction between the present analysis and classical association rule mining. Association rule mining algorithms, such as the popular Apriori algorithm [27], require arbitrary support and confidence values to isolate “interesting” associations. Instead, the present analyses used null hypothesis significance testing to identify frequent associations between the cognitive and affective states.

The analyses proceeded as follows. For a given cognitive state and affective state, we first computed the probability that they simultaneously occurred during the same student move (i.e. $P[A, C]$). Next, the probability of occurrence was computed from a randomly shuffled surrogate of the corpus. In this surrogate corpus, the temporal ordering of cognitive states was preserved, however, the ordering of affective states was randomized (i.e. $P[A, C]$). This process breaks temporal dependencies, but preserves base rates. The process was repeated for each of the 50 sessions, thereby yielding values for each session. However, there is a potential limitation in the creation of only one surrogate corpus. Paired samples *t*-tests were then used to determine whether these quantities (i.e. $P[A, C]$ and $P[A, C]$) significantly differed.

3.1 Co-occurrence Relationships between Student Affective and Cognitive States

The analyses proceeded by computing a 12×4 (cognitive \times affective) co-occurrence matrix for each session from the original corpus and comparing this to a 12×4 matrix obtained from the randomly shuffled surrogate corpus. The effect sizes (Cohen’s *d*) for the cognitive-affective co-occurrences are presented in Table 1.

It appears that 16 out of the 48 potential co-occurrences were statistically significant at the $p < .05$ level. There is the potential concern of committing Type I errors due to the large number of significance tests conducted in the present analyses. Fortunately, Monte-Carlo simulations across 100,000 runs confirmed that the probability of obtaining 16 out of 48 significant transitions (33.3%) by chance alone is approximately 0. Therefore, it is unlikely that the patterns in Table 1 were obtained by a mere capitalization of chance.

Let us first consider co-occurrences between affective states and answer types. The results indicate that *confusion* is associated with all forms of incorrect responses, but not with *correct* responses, thereby confirming the major predictions of cognitive

disequilibrium theory. Interestingly, the magnitude of the effects of these associations scales with the severity of student answer quality. In particular, there is a medium effect for the *confusion-partially-correct* answer association, a medium to large effect for the *confusion-vague* answer association, and a large effect for the *confusion-error-ridden* answer association.

Consistent with state-trait anxiety theory, *anxiety* does not occur with *correct* answers, but occurs with *error-ridden* and *no answers*. These patterns may be indicative of student's awareness of their knowledge gaps and embarrassment or worry over those deficits. However, it may also be the case that the presence of anxiety impedes the student's ability to access the correct answer, as is also predicted by state-trait anxiety theory. Thus anxiety-ridden answers alone may appear to be knowledge deficits, but it may be that the student does have the knowledge and their anxiety is impeding access to that knowledge.

Table 1. Effect sizes for co-occurrence of student affective and cognitive states

Cognitive State	Affective State			
	Anxiety	Confusion	Frustration	Happiness
Answer Type				
Correct	-.32*	.23	-.29	-.01
Partially-Correct	.26	.50*	.26	-.20
Vague	.26	.65*	.40*	-.26
Error-Ridden	.42*	.83*	.19	-.28
None	.59*	.41*	.28	-.32
Question Type				
Common Ground	.36*	.92*	.20	-.10
Knowledge Deficit	.08	.60*	-.01	-.15
Metacognition				
Misconception	.55*	.38	1.91*	.24
Metacomment	.74*	.50*	.46*	.17

* $p < .05$. $d \approx .2, .5, .8$ indicate small, medium, and large effects, respectively [28]

Although goal-appraisal theories would predict that *frustration* would be associated with *vague*, *error-ridden*, and *no answers*, some of these predictions were not supported in the present analyses. In particular, *frustration* was associated with *vague* answers but not with *error-ridden* and *no answers*. *Vague answers* involve a difficulty in formulating a coherent response (“Write the, uh, before the...”). It may be that this difficulty in conjunction with knowledge deficits brings about *frustration*.

Turning to associations between affect and question asking behaviors, the results indicate that *confusion* co-occurs with both question types, which is what would be

expected. Consistent with the aforementioned discussion, the presence of *anxiety* with *common ground questions* indicates feelings of uncertainty or a lack of confidence. The lack of an association with *knowledge deficit questions* may be linked to the performance reduction predicted by state-trait anxiety theory. Failure to ask knowledge deficit questions when knowledge gaps exist will likely lead to poorer outcomes for the student.

Misconceptions involve students asserting that their prior idea or belief is in fact erroneous. The results indicate that *confusion* was not associated with misconception statements, an expected finding because students presumably alleviate their confusion when they verbally acknowledge their misconception. In contrast, the presence of *anxiety* or *frustration* with *misconceptions* may indicate that the student has recognized the erroneous belief, and are troubled by their misconceptions.

The results indicated that *confusion*, *frustration*, and *anxiety* were associated with *metacomments*. This relationship is consistent with goal-appraisal theory in that the student is assessing their knowledge in reference to the goal of learning the material. The distinction between each associated emotion may be due to how far the student is from mastering the material and whether they perceive an available plan for learning the material. For example, *frustration-metacomment* associations could occur when a student does not understand the material and has no plan for how they will learn the material in time to pass their class.

Our results also indicate that *happiness* was not associated with any of the cognitive or metacognitive states. This finding is intuitively plausible because the cognitive states we investigated are more related to the learning process rather than learning outcomes. Happiness might be related to outcomes such as receiving positive feedback from the tutor.

3.2 Association Rules between Cognitive and Affective States

Next we investigated the association rules between the cognitive and affective states that significantly co-occurred. The cognitive state \rightarrow affective state association was the focus of this paper, although the reverse relationship can be investigated as well. We chose to focus on this relationship because of its potential for informing the development of affect-sensitive ITSs (as will be discussed below).

The analyses proceeded by computing confidence values for significant co-occurrences from the actual and randomly shuffled data sets (see above). These were then compared with paired-sample *t*-tests. Table 2 displays effect sizes for the association rules.

As could be expected, incorrect answers predicted *anxiety*, *confusion*, and *frustration*, but with important differences. Students being unable to provide an answer was a stronger trigger for *anxiety* than *confusion*. The inability to even provide an answer represents the highest degree of error, consistent with state-trait anxiety theory’s prediction of high anxiety negatively impacting performance. In contrast, *error-ridden* answers were linked to *confusion*, which is consistent with theories that highlight impasses and knowledge gaps during learning. Finally, *vague* answers were predictive of both *confusion* and *frustration*.

Student questions were predictive of *confusion*, but not any other state, a finding that is consistent with cognitive disequilibrium theory and research on the merits of

question asking during learning [10-13]. *Common ground questions* ($d = 0.88$) show a stronger association with *confusion* than *knowledge deficit questions* ($d = 0.63$). *Common ground questions* suggest a level of doubt (“We don’t distribute between, like this? Or we don’t do”), while *knowledge deficit questions* suggest a gap in knowledge (“What’s the line?”). Hence, *confusion* may be related to uncertainty with knowledge rather than gaps in knowledge.

Misconceptions and *metacomments* showed highly similar association patterns in both analyses. Interestingly, both are stronger triggers for *anxiety* than *frustration*. *Anxiety* may be triggered because the student feels embarrassed or worried about past misconceptions, rather than feeling irritated with these past failures. Although resolved, these *misconceptions* may continue to trouble the student due to their history of struggling with academics and a lack of confidence in their own abilities.

Table 2. Effect sizes for association rules

Dialogue Move	Affective State			
	Anxiety	Confusion	Frustration	Happiness
Answer Type				
Correct	---	---	---	---
Partial	---	.17	---	---
Vague	---	.55*	.41*	---
Error-Ridden	.33	.87*	---	---
None	.52*	.24	---	---
Question Type				
Common Ground	.17	.88*	---	---
Knowledge Deficit	---	.63*	---	---
Metacognition				
Misconception	.40*	.37	.07	---
Metacomment	.75*	---	.43*	---

* *significant at $p < .05$* . --- indicates associations not tested because the co-occurrence was not significant in prior analyses (See Table 1)

In summary, the results indicate that *confusion* is predicted by four factors (*vague* answers, *error-ridden* answers, *common ground questions*, and *knowledge deficient questions*), with a mean effect of .73 sigma (medium to large effect). *Anxiety*, on the other hand, is predicted by three separate factors (*no answers*, *misconceptions*, and *metacomments*) with a mean effect of .553 sigma (medium effect). Finally, *frustration* is predicted by two factors that overlap with predictors of confusion and anxiety (*vague* answers and *metacomments*). The mean effect size for predicting *frustration* was .417 sigma, which is consistent with a small to medium effect. Hence, the

cognitive states are most effective in predicting *confusion*, less effective for *frustration*, and somewhat effective for *anxiety*.

4 Discussion

In this paper we investigated the cognition-emotion relationship with association rule mining. Overall these findings support the idea that affect and cognition are inextricably linked during learning [2, 5-9]. *Confusion* appears to be bound to problem solving and learning as predicted by cognitive disequilibrium theory. *Happiness*, conversely, appears to have a different role during learning. The lack of any cognitive association suggests that it is tied to the product rather than the process of learning. *Anxiety* and *frustration* fall in between these two extremes. While both are generally predicted to be detrimental to learning [10-13,15], they are strongly related to *misconceptions* and *metacomments*. These statements allow for direct access into the student's current knowledge. So while a prevalence of these two affective states would not be advantageous, their presence during learning is important.

To truly understand the student learning experience, a combination of cognitive disequilibrium, state-trait, and goal-appraisal theories seems necessary. Cognitive disequilibrium theory accounts for the typical struggles of problem solving and broad associations between affect and failure (i.e., incorrect answers-*frustration* and -*confusion*), state-trait anxiety theory predicts the important role of *anxiety* during learning, and goal-appraisal theory allows for distinctions within levels of incorrect answers (i.e., *partially-correct* vs. *error-ridden*). An amalgamation of these relationships will allow for a better understanding of the student's knowledge and more effective responses by both human and computer tutors.

The goal of affect-sensitive ITSs is now a reality [21-25]. However, the best method for identifying affective states and determining how this information will be used is still unclear. These relationships found in human-human expert tutoring sessions can target key moments to assess student affect. After detection, these associations can guide differentiated feedback to students. ITSs can give individualized feedback based on answer quality and the associated affective state. Thus, an *error-ridden* answer combined with *confusion* would receive different feedback than the same answer accompanied by *anxiety*. This will allow for simultaneous sensitivity to student cognition and affect. While this level of individuation in feedback is hypothesized to cause greater learning gains, only future research will tell its true usefulness. However, this is a further step to achieving the expert human tutoring standards of excellence (individualization, immediacy, and interactivity) [29,30] in ITSs.

Acknowledgement. The research reported here was supported by the Institute of Education Sciences (R305A080594), the U. S. Office of Naval Research (N00014-05-1-0241), and the National Science Foundation (REC 0106965, ITR 0325428, and REC 0633918). The opinions expressed are those of the authors and do not represent views of the Institute of Education Sciences, the U.S. Department of Education, the Office of Naval Research, NSF, or DoD.

References

1. Damasio, A.: *Descartes's Error: Emotion, Reason, and the Human Brain*. Gosset/Putnam, New York (1994)
2. Bower, G.H.: Emotional Mood and Memory. *American Psychologist* 36, 129–148 (1981)
3. Dagleish, T., Power, M.J.: *Handbook of Cognition and Emotion*. Wiley, Chichester (1999)
4. Lazarus, R.S.: The Cognition-Emotion Debate: A Bit of History. In: Dagleish, T., Power, M.J. (eds.) *Handbook of Cognition and Emotion*, pp. 3–19. Wiley, Chichester (1999)
5. Mandler, G.: *Mind and Body: Psychology of Emotion and Stress*. Norton, New York (1984)
6. Barrett, L.F., Mesquita, B., Ochsner, K.N., Gross, J.J.: The Experience of Emotion. *Annual Reviews* 58, 373–403 (2007)
7. Ortony, A., Clore, G.L., Collines, A.: *The Cognitive Structure of Emotions*. Cambridge University Press, Cambridge (1988)
8. Scherer, K.R., Schorr, A., Johnstone, T.: *Appraisal Processes in Emotion: Theory, Methods, Research*. Oxford University Press, New York (2001)
9. Stein, N.L., Levine, J.L.: Making Sense Out of Emotion: The Representation and Use of Goal-Structured Knowledge. In: Kessen, W., Ortony, A., Craik, F. (eds.) *Memories, Thoughts, and Emotions: Essays in Honor of George Mandler*, pp. 295–322. Lawrence Erlbaum Associates, Inc., England (1991)
10. Bjork, R.A., Linn, M.C.: The Science of Learning and The Learning of Science: Introducing Desirable Difficulties. *APS Observer* 19, 29 (2006)
11. Festinger, L.: *A Theory of Cognitive Dissonance*. Row Peterson, Evanston (1957)
12. Graesser, A.C., Lu, S., Olde, B.A., Cooper-Pye, E., Whitten, S.: Question Asking and Eye Tracking during Cognitive Disequilibrium: Comprehending Illustrated Texts on Devices When The Devices Breakdown. *Memory & Cognition* 33(7), 1235–1247 (2005)
13. Piaget, J.: *The Origins of Intelligence in Children*. Basic Books, New York (1952)
14. VanLehn, K., Siler, S., Murray, C., Yamauchi, T., Baggett, W.B.: Why Do Only Some Events Cause Learning During Human Tutoring? *Cognition and Instruction* 21(3), 209–249 (2003)
15. Spielberger, C.D., Goodstein, L.D., Dahlstrom, W.G.: Complex Incidental Learning as a Function of Anxiety and Task Difficulty. *J. of Experimental Psychology* 56(1), 58–61 (1958)
16. Fielder, K.: Affective States Trigger Processes of Assimilation and Accommodation. In: Martin, L., Clore, G. (eds.) *Theories of Mood and Cognition: A User's Guidebook*, pp. 85–98. Erlbaum, Mahwah (2001)
17. Isen, A.: An Influence of Positive Affect on Decision Making in Complex Situations: Theoretical Issues with Practical Implications. *J. of Consumer Psychology* 11, 75–85 (2001)
18. Hertel, G., Neuhof, J., Theuer, T., Kerr, N.: Mood Effect on Cooperation in Small Groups: Does Positive Mood Simply Lead to More Cooperation? *Cognition and Emotion* 14, 441–472 (2000)
19. Schwarz, N., Skurnik, I.: Feeling and Thinking: Implications for Problem Solving. In: Davidson, J., Sternberg, R. (eds.) *The Psychology of Problem Solving*, pp. 263–290. Cambridge University Press, New York (2003)
20. Person, N., Lerhman, B., Ozbun, R.: Pedagogical and Motivational Dialogue Moves Used by Expert Tutors. In: Presented at the 17th Annual Meeting of the Society for Text and Discourse, Glasgow, Scotland (2007)

21. D'Mello, S.K., Craig, S.D., Fike, K., Graesser, A.C.: Responding to Learner's Cognitive-Affective States with Supportive and Shakeup Dialogues. In: Jacko, J.A. (ed.) *Human Computer Interaction; Ambient, Ubiquitous and Intelligent Interaction*, pp. 595–604. Springer, Heidelberg (2009)
22. Conati, C., Maclaren, H.: Empirically Building and Evaluating a Probabilistic Model of User Affect. *User Modeling and User-Adapted Interaction* 19(3), 267–303 (2009)
23. Arroyo, I., Woolf, B., Cooper, D., Burleson, W., Muldner, K., Christopherson, R.: Emotion Sensors Go to School. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Graesser, A. (eds.) *Proceedings of 14th International Conference on Artificial Intelligence in Education*, IOS Press, Amsterdam (2009)
24. D'Mello, S.K., Person, N., Lehman, B.A.: Antecedent-Consequent Relationships and Cyclical Patterns between Affective States and Problem Solving Outcomes. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Graesser, A. (eds.) *Proceedings of 14th International Conference on Artificial Intelligence in Education*, pp. 57–64. IOS Press, Amsterdam (2009)
25. Litman, D., Forbes-Riley, K.: Recognizing Student Emotions and Attitudes on the Basis of Utterances in Spoken Tutoring Dialogues with Both Human and Computer Tutors. *Speech Communication* 48(5), 559–590 (2006)
26. Lehman, B., Matthews, M., D'Mello, S., Person, N.: What Are You Feeling? Investigating Student Affective States during Expert Human Tutoring Sessions. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 50–59. Springer, Heidelberg (2008)
27. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules in Large Databases. In: *Proceedings of the VDLB Conferences*, Santiago, Chile (1994)
28. Cohen, J.: A Power Primer. *Psychological Bulletin* 112(1), 155–159 (1992)
29. Bloom, B.S.: The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher* 13, 4–16 (1984)
30. Lepper, M.R., Woolverton, M.: The Wisdom of Practice: Lessons Learned from Highly Effective Tutors. In: Aronson, J. (ed.) *Improving Academic Achievement: Impact of Psychological Factors on Education*, pp. 135–158. Academic Press, San Diego (2002)

Subliminally Enhancing Self-esteem: Impact on Learner Performance and Affective State

Imène Jraidi and Claude Frasson

HERON Lab, Computer Science Department
University of Montreal, CP 6128 succ. Centre Ville
Montréal, QC, H3T-1J4, Canada
{jraidiim, frasson}@iro.umontreal.ca

Abstract. The purpose of this work is to enhance learner self-esteem while interacting with a tutoring system. Our approach is based on a subliminal priming technique that implicitly conditions learner self-esteem. An experimental study has been conducted to analyze the impact of this method on participants' reported self-esteem on one hand and learning performance on the other hand. Furthermore, three physiological sensors were used to continuously monitor participants' affective reactions, namely electroencephalogram, skin conductance and blood volume pulse sensors. The purpose was to measure the effect of our approach on both learner mental state and emotions. We then proposed a model that links learners' physiological signals and priming conditions to learning results.

Keywords: subliminal priming, self-esteem, learning performance, sensors, learner affect.

1 Introduction

Nowadays learner affect has become a key construct in Intelligent Tutoring System (ITS) researchers. Several works focus on affective student modeling [4], identifying learner emotions [5], detecting frustration and stress [23] or assessing attention levels [25]. Most of these systems use a variety of physical cues to recognize affective state [22] including observable changes like face expressions, body postures, vocal tones, and physiological signal changes such as heart rate, skin conductivity, temperature, respiration and brain electrical activity. Ultimately, these works seek to properly adapt tutorial interventions and improve learner performance.

On the other side, many educators and pedagogues advocate the benefits of self-esteem in learning. A broad strand of research investigated the positive effects of self-esteem on learner self-confidence [20]. Besides, several studies have shown strong correlations between self-esteem and academic achievement and success [12].

Recently, McQuiggan, Mott, and Lester [21] proposed an inductive approach to model learner self-efficacy. They used learners' demographic and physiological data to predict their self-efficacy level. While self-efficacy represents the individual's

belief about her ability to execute specific tasks, self-esteem is a more generalized aspect [14]. It reflects the overall personal self evaluation.

Mainly, literature differentiates between explicit self-esteem and implicit self-esteem [8]. The former is based on conscious mode of thinking and can be measured by means of questionnaires, whereas the latter is the result of automatic self-evaluative process and can be assessed with indirect measures. Unlike explicit measures which are based on generally biased self-report, implicit measures are based on unconscious attitude toward the self [8].

These latter measures are mostly used in unconscious process based researches, mainly in the neuro-psychological communities. The core of these researches is the existence of a threshold-line of conscious perception. The idea is that a stimulus below this threshold of awareness, also called subliminal stimulus cannot be consciously perceived but can yield affective reactions without awareness [6]. This technique is known as subliminal priming. It has been applied in different contexts [15] including self-esteem conditioning and learning improvement.

In this paper, we propose to integrate the implicit self-esteem component within learning process. More precisely, the aim is to condition learner self-esteem while interacting with a tutoring system, using a subliminal priming strategy. The hypothesis we establish is that this method can improve learner performance. We propose to conduct an experimental study using a subliminal priming technique.

Our research questions are the followings: can subliminal priming enhance participants' self-esteem? Can this method produce a positive effect on learning performance? Is there any effect on learners' emotions and mental states? What is the influence of learner physiological activity and priming conditions on learning results?

The remainder of the paper is organized as follows. We start by outlining the background concerning the subliminal priming approach. Next, we describe the developed tutoring system and experimental setup. Then, we discuss the obtained results, conclude and present directions for future work.

2 Previous Work on Subliminal Priming

Researches devoted to automatic or unconscious processes have increased over the last years. Their basic assumption lies on the existence of a threshold-line between conscious and unconscious perception [6]. A stimulus is known as subliminal, if it is received below this threshold of awareness and cannot be consciously reported. High-level semantic and even emotional processing has been observed during this stage [15]. Masked priming is one of the main techniques used to project subliminal information [6]. In this method, a subliminal stimulus, also called prime, is projected during very short time. The prime is preceded and/or followed by the projection of a mask for a specific time. This mask usually takes the form of a series of symbols having nothing to do with the prime in order to elude its conscious detection.

In the Human Computer Interaction (HCI) community, Wallace, Flanery and Knezek [27] implemented subliminal clues for task-supported operation within a text editor program. They found that the frequency at which subjects demanded help was much lower when the required information was subliminally presented. In another perspective, DeVaul, Pentland and Corey [7] used subliminal clues for just-in time

memory support. They investigated the effect of various subliminal information on retention in a word-face learning paradigm.

In the ITS community, Chalfoun and Frasson [3] used a subliminal priming method within a 3D virtual tutoring system. It was found that overall performance was better, and time for answering questions was shorter for learners primed with subliminal clues. Learners' emotional reactions were also different; subliminal stimuli elicited high arousal states. Hence, besides yielding better results, subliminal priming seemed to elicit emotional consequences not only in learning, but also in various other domains like: social behavior, advertisement, stereotypes, food preferences, etc. (see [15] for a review). On the other side, evidence from this body of literature indicates that this effect is more important compared to consciously perceived and reported stimulus effects [1]. A recent work of Radel, and colleagues [24] put forward an interesting effect that subliminal priming can have on motivational processes. They investigated the impact of motivational primes in a natural setting, namely the classroom. A positive effect of subliminal priming on academic performance was found; this effect was basically moderated by learner mindfulness.

In this paper, we propose to introduce a new approach to subliminally enhance learner self-esteem while interacting with a tutoring system. We are interested in analyzing the effect of this method on participants' reported self-esteem, on their learning performance and affective states. Our methodology and experimental setup are described in the next section.

3 Experimental Methodology

Materials. The tutoring environment developed for this experiment consists of a multiple choice questionnaire related to logic. The questions are typically found in brain training exercises or in tests of reasoning ability. They involve inferential skills on information series and do not require particular prerequisites in any field of knowledge. The questionnaire is composed of 3 modules. Each module is concerned with specific forms of data: the first module deals with geometrical shapes, the second module with numbers and the third module focuses on letters. In each module, learners have to answer to 5 multiple choice questions. Figure 1 depicts a screenshot of our system from each module. The idea is to try to find the logical rule between the data, and guess the missing one.

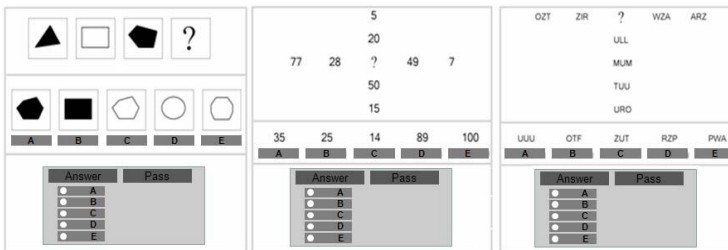


Fig. 1. Screenshots from each module

Each Module starts with a tutorial giving instructions and examples to get learners accustomed with the user interface and types of questions. Learners are asked to respond as quickly and efficiently as possible to each of the 15 questions of the quiz. A correct answer was worth 4 points, an incorrect answer -1, and a no-answer 0.

3.1 Enhancing Self-esteem

In order to enhance learner self-esteem, we used an evaluative conditioning (EC) subliminal procedure [11]. This method consists in subliminally projecting self-referent words (conditioned stimulus or CS) paired with positive words (unconditioned stimulus or US). The idea behind EC, is that conditioning influences the structure of associations in memory, and hence the automatic affective reactions resulting from these associations [11]. This method has already been found to influence self-esteem in earlier experiments (e.g. [8, 11]). Besides, it has been found that EC effects occur without awareness of the stimulus pairing.

Hence, in our experiment, some participants (experimental condition), were repeatedly presented with the subliminal primes (CS and US stimuli).¹ The other participants (control condition), were not presented with subliminal primes. Projecting thresholds were carefully chosen according to neural recommendations [6]. Each subliminal prime (self-referent word and positive word)² was displayed for 29 ms preceded and followed by a 271 ms mask of a set of sharp (#) symbols.

Self-Esteem Measure. Learner self-esteem was assessed using the Initial Preference Task (IPT), [19].³ Participants were asked to evaluate their attractiveness for all letters of the alphabet on a 7-point scale. Letters were presented individually, in random order on the screen. Participants pressed the corresponding key to evaluate each letter. High self-esteem is indexed by the extent to which a person prefers his or her initials to other letters of the alphabet.

3.2 Physiological Measures

Physiological measures were recorded continuously during the experiment using a ProComp Infinity encoder. Three types of sensors were used: electroencephalogram (EEG), skin conductance (SC) and blood volume pulse (BVP) sensors. (1) EEG electrical brain activity was recorded using a lycra stretch cap placed on the scalp. Cap electrodes were positioned according to the International 10/20 Electrode Placement System [16]. EEG signals were recorded from 4 scalp sites (P3, C3, Pz and Fz). Each site was referred to Cz and grounded at Fpz. EEG signals were calibrated with regards

¹ In order to get learners focused on the screen, questionnaire materials (shapes, numbers and letters) were presented sequentially. Subliminal stimuli were then presented just before the materials appeared.

² Self referent words were: I, and participant's first name. Positive words were: nice, smart, strong, success, and competent. These words were selected from previous studies addressing self-esteem [10].

³ Participants were also asked to complete an additional self-esteem measure, namely the Implicit Association Test (IAT, [10]). However, since IAT yielded essentially the same results as the IPT, only the effects concerning IPT are reported. These measures were chosen according to [2] assessing the most promising implicit measures of self-esteem.

to the average of left and right earlobe sites (A1 and A2). Each electrode site was filled with a small amount of electrolyte gel and sensor impedance was maintained below 5 K Ω . The recorded sampling rate was at 256 Hz. (2) SC sensors were placed in the 2nd and 4th left hand finger. (3) BVP sensor was placed in the 3rd left hand finger. SC and BVP data were recorded at 1024 Hz of sampling rate. Heart rates (HR) were derived from BVP signals and galvanic skin response (GSR) from SC. All signals were notch filtered at 60 Hz to remove environmental interference during data acquisition. Besides, two webcams were used to synchronize physiological signals with the tutoring system tasks. The former monitored the learner's facial activity and the latter recorded the learner's interactions on the computer screen.

Affect Recognition. From the physiological recorded signals, we wanted to analyze both learners' mental and emotional activities. In order to analyze the mental state, we used the recorded EEG signals. Indeed, neural research established various EEG-based mental states and neural indexes of cognition [9]. More precisely, EEG studies on mental concentration and attention [13] defined an EEG indicator of attention to internal processing during performance of mental tasks. It was found that an increase in the delta and low theta (delta_low_theta) activity is related to an increase in subjects' internal concentration [13]. For analyzing learner brain activity within this frequency band, we applied a Fast Fourier Transform (FFT) to transform the EEG signal into a power spectrum. We then extracted the percentage of delta_low_theta band (1.56 - 5.46 Hz) [13] from the transformed signal. We used these values as an indication of learner mental concentration while answering to the questionnaire.

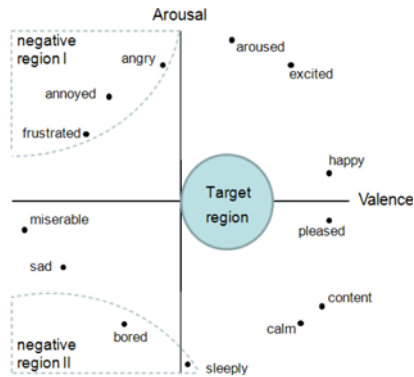


Fig. 2. Russell's circumplex model of emotions with regions

In order to assess learner emotional state we considered HR and GSR signals which are known to be correlated to valence (positive to negative) and arousal (low to high) [18] and we used Russell's circumplex model of emotions [26]. This model classifies emotions in terms of valence and arousal dimensional spaces. Two strategic emotional regions were defined during learning as depicted in figure 2 [17]. The first region involves negative emotions like frustration, boredom or anger (negative region I and II) and should be avoided. The second region is the target emotional region specified by a slight positive valence and neutral arousal. This region provides

a maximum of efficiency and productivity in learning [17]. In our study, we focused on the proportion of positive emotions in the target region. We weighted then the number of HR and GSR recordings corresponding to this region by the total number of recordings.

3.3 Experimental Protocol

Upon arrival at the laboratory, participants were briefed about the procedure and consent was obtained. They were then randomly assigned either to the experimental condition or to the control condition. The former took place with self-esteem conditioning subliminal stimuli and the latter with no subliminal stimuli. Baselines for physiological signals were recorded during which participants were instructed to relax. The logic materials were then displayed with the instructions, warm-up examples and questions related to each of the three modules as described earlier. Finally, participants were asked to complete the IPT self-esteem scale.

3.4 Participants

39 participants ranged in age from 19 to 47 years ($M = 27.34$, $SD = 6.78$) took part to our study. They received 10 CAD compensation for their participation. They were assigned either to the experimental condition or to the control condition. Repartition of participants is given in Table 1.

Table 1. Repartition of participants

	Males	Females
Experimental condition	13	7
Control condition	11	8

4 Results and Discussion

Results are presented in four sections. The first section presents self-esteem measure results. The second section deals with learner performance. The third section analyzes learner affective states. Finally, the fourth section describes the overall influence of priming conditions and affective measures on learner performance.

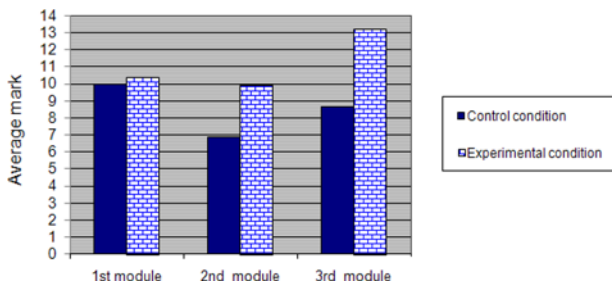
Self-esteem. Learner self-esteem was measured in terms of IPT effect by using the I-algorithm [19]. Mean rating of all non-initial letters is subtracted from each letter rating. Normative letter baselines are then computed by averaging the ipsatized letter ratings for individuals whose initials do not include the letter. The difference score is finally computed between the ipsatized initial ratings and the respective ipsatized baselines [19]. Means and standard deviation of IPT scores are listed in Table 2.

It is shown that the IPT effect was more pronounced for participants in the conditioned self-esteem group (experimental condition) compared to participants in the control condition indicating higher self-esteem. This difference was statistically reliable, $F(1, 37) = 4.84$, $p < .05$. Hence, results confirm that our method produced the expected main effect on learners' self-esteem.

Table 2. Means and standard deviations of IPT self-esteem measure

	M	SD
Experimental condition	1.68	.94
Control condition	1.08	.99

Learning Performance. To measure learner performance we considered marks obtained in the logic questionnaire. Figure 3 presents the average marks in each module of the quiz. It is shown that participants in the experimental group have had better marks in the 3 categories of questions. Besides, questionnaire final marks were significantly higher in the experimental condition ($M = 33.4$, $SD = 12.36$) compared to those in the control condition ($M = 25.5$, $SD = 9.87$), $F(1, 37) = 4.37$, $p < .05$.

**Fig. 3.** Average marks per module

In another prospect, we analyzed the number of no-answers in both groups of participants. A main effect was found: $F(1, 37) = 7.45$, $p < .05$. The number of no-answers was significantly lower in the experimental condition ($M = .95$, $SD = .83$) compared to the control condition ($M = 2.11$, $SD = 1.91$).

To sum up, a clear evidence of the positive effect of the priming strategy on learners' marks in the questionnaire was found. This was a priori explained by a higher risk taking in the conditioned self-esteem group of participants since that an incorrect answer was worth -1 point and a no-answer worth 0 point in the final mark.

Learner affect. In our next investigation, we compared mental and emotional activities between participants of the experimental condition and participants of the control condition. For the mental activity, we considered participants' mental concentration while answering to the questionnaire tasks. Figure 4 sketches out the variation of the mean percentage of delta_low_theta in each question of the quiz with regards to the baseline for two participants. The first participant was primed with subliminal self-esteem conditioning primes, and the second one was not projected with primes: mean percentage of delta_low_theta band in each question was subtracted from the baseline value. It is shown that in 12 questions over 15, the first participant has had a higher increase in delta_low_theta activity than the second one regarding to their respective baselines.

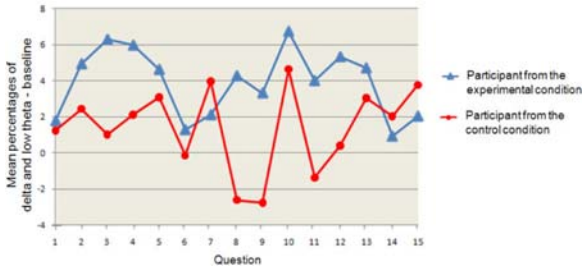


Fig. 4. Variation of mean percentage of delta_low_theta

An overall effect was found: subliminally primed participants reported a higher increase in the percentage of delta_low_theta band with regards to the baseline ($M = 3.59$, $SD = 16.81$), compared to not primed participants ($M = 1.14$, $SD = 10.14$). The effect was statistically reliable ($F(1, 583) = 4.3$, $p < .05$).

Hence, it was found that participants with conditioned self-esteem yielded higher delta_low_theta activity and thus higher concentration level during tasks [13].

To analyze participants' emotional states, we look at the positive target emotion proportions over each question of the quiz. A significant main effect of priming conditions was found, $F(1, 583) = 6.03$, $p < .05$. Participants of the experimental group reported higher target emotion proportions ($M = 48.15$, $SD = 37.32$) than participants in the control condition ($M = 40.36$, $SD = 34.02$). This reflects that conditioned self-esteem participants were more frequently in the positive emotional target region.

Regression analysis. Our last investigation dealt with the overall impact of the learners' recorded affective data and self-esteem conditioning on learning performance. That is can we predict the learner results in the questionnaire on the basis of his priming condition, mental concentration, and emotional state. A multiple regression analysis was conducted to measure the influence of each of these parameters. The dependent variable used was participants' marks on each question of the quiz. Three predictors were used in the analysis: (1) priming conditions (coded +1 for the experimental condition and -1 for the control condition), (2) variation of the mean percentage of delta_low_theta with regards to baseline and (3) mean proportion in the emotional target region.

The overall model was significant ($F(3, 581) = 6.91$, $p < .01$, $R^2 = .34$). Conditional main effect analyzes revealed the expected positive effect of priming conditions ($\beta = .95$, $p < .05$). A main effect was also found for the variation of percentage of delta_low_theta ($\beta = .13$, $p < .05$) and for the proportion in the emotional target region ($\beta = .93$, $p < .05$). From this result, we can statistically deduce that learners' results in the questionnaire were positively influenced by the self-esteem priming condition, high mental concentration and more frequent emotions in the target region characterised by a slight positive valence and neutral arousal.

5 Conclusion

In this paper, we have proposed to use the self-esteem component within learning process. More precisely, our objective was to enhance learner implicit self-esteem

while interacting with a tutoring system. Our approach is based on a subliminal, non-consciously perceived, self-esteem conditioning method.

Our experimental study has shown that this method enhanced participants' implicit self-esteem on one hand, and learning performance in terms of marks obtained in the logic questionnaire, on the other hand. Besides, priming conditions elicited different mental and emotional reactions: conditioned self-esteem participants showed higher mental concentration and higher proportions of positive emotions with regards to the target emotional region of Russel's circumplex model. Finally, we proposed to evaluate the contribution of each variable derived from learners' physiological signals and priming conditions for the prediction of learners' results in the questionnaire.

We believe that these findings can yield interesting implications in intelligent tutoring systems. Nevertheless, many ethical concerns should be established before applying subliminal strategies in applied settings [15]. Our future work is directed towards studying the impact of the self-esteem conditioning approach on a broader set of learner physiological features such as motivation, and mental workload. We also plan to conduct deeper analysis on correlations between learner self-esteem level, emotions and mental state within more complex learning situations.

In another perspective, we intend to model learners' level of self-esteem from their personal characteristics and physiological activities in order to extend the learner's module within an intelligent tutoring system.

Acknowledgments. We acknowledge the CRSNG (Conseil de Recherches en Sciences Naturelles et en Génie du Canada) and the Tunisian Government for their support.

References

1. Bornstein, R.F., D'Agostino, P.R.: Stimulus recognition and the mere exposure effect. *JPSP* 63, 545–552 (1992)
2. Bosson, J.K., Swann, W.B., Pennebaker, J.W.: Stalking the perfect measure of self esteem: The blind men and the elephant revisited. *JPSP* 79, 631–643 (2000)
3. Chalfoun, P., Frasson, C.: Subliminal priming enhances learning in a distant virtual 3D Intelligent Tutoring System. *IEEE MEEM* 3, 125–130 (2008)
4. Conati, C., Maclaren, H.: Data-Driven Refinement of a Probabilistic Model of User Affect. *User Modeling*, 40–49 (2005)
5. D'Mello, S.K., Graesser, A.: Automatic detection of learner's affect from gross body language. *AAI* 23, 123–150 (2009)
6. Del Cul, A., Baillet, S., Dehaene, S.: Brain dynamics underlying the nonlinear threshold for access to consciousness. *PLoS, Biology* 5, 2408–2423 (2007)
7. DeVaul, R.W., Pentland, A., Corey, V.R.: The Memory Glasses: Subliminal vs. Overt Memory Support with Imperfect Information. In: *Wearable Computers, IEEE International Symposium*, pp. 146–153 (2003)
8. Dijksterhuis, A.P.: I like myself but I don't know why: Enhancing implicit self-esteem by subliminal evaluative conditioning. *JPSP* 86, 345–355 (2004)
9. Fabiani, M., Gratton, G., Coles, M.G.: Event-related brain potentials. In: Cacioppo, J.T., Tassinary, L.G., Berntson, G.G. (eds.) *Handbook of psychophysiology*, pp. 53–84. Cambridge University Press, Cambridge (2000)

10. Greenwald, A.G., Farnham, S.D.: Using the implicit association test to measure self-esteem and self-concept. *JPSP* 79, 1022–1038 (2000)
11. Grumm, M., Nestler, S., Collani, G.v.: Changing explicit and implicit attitudes: The case of self-esteem. *JESP* 45, 327–335 (2009)
12. Hansford, B.C., Hattie, J.A.: The Relationship between self and achievement/performance measures. *Review of Educational Research* 52, 123–142 (1982)
13. Harmony, T., Fernández, T., Silva, J., Bernal, J., Díaz-Comas, L., Reyes, A., Marosi, E., Rodríguez, M., Rodríguez, M.: EEG delta activity: an indicator of attention to internal processing during performance of mental tasks. *IJP* 24, 161–171 (1996)
14. Harter, S.: Causes, correlates, and the functional role of global self-worth: A life-span perspective. In: Sternberg, R.J., Kolligian, J. (eds.) *Competence considered*, pp. 67–97. Yale University Press, US (1990)
15. Hassin, R., Uleman, J., Bargh, J.: *The new unconscious*. Oxford University Press, Oxford (2005)
16. Jasper, H.H.: The ten-twenty electrode system of the International Federation. *Electroencephalography and Clinical Neurophysiology*, 371–375 (1958)
17. Kaiser, R.: Prototypical development of an affective component for an e-learning system; Master Thesis, University of Rostock, Germany (2006)
18. Lang, P.J.: The emotion probe: Studies of motivation and attention. *American Psychologist* 50, 372–385 (1995)
19. LeBel, E.P., Gawronski, B.: How to find what's in a name: Scrutinizing the optimality of five scoring algorithms for the name-letter task. *EJP* 23, 85–106 (2009)
20. McFarlin, D.B., Blascovich, J.: Effects of self-esteem and performance feedback on future affective preferences and cognitive expectations. *JPSP* 40, 521–531 (1981)
21. McQuiggan, S.W., Mott, B.W., Lester, J.C.: Modeling self-efficacy in intelligent tutoring systems: An inductive approach. *UMUAI* 18, 81–123 (2008)
22. Picard, R.: *Affective Computing*. MIT Press, Cambridge (1997)
23. Prendinger, H., Ishizuka, M.: The empathic companion: A character-based interface that addresses users' affective states. *AAI* 19, 267–285 (2005)
24. Radel, R., Sarrazin, P., Legrain, P., Gobancé, L.: Subliminal Priming of Motivational Orientation in Educational Settings: Effect on Academic Performance Moderated by Mindfulness. *J. Res. Pers.* 43, 695–698 (2009)
25. Rebolledo-Mendez, G., Dunwell, I., Martínez-Mirón, E., Vargas-Cerdán, M., de Freitas, S., Liarokapis, F., García-Gaona, A.: Assessing NeuroSky's Usability to Detect Attention Levels in an Assessment Exercise. In: *HCI, New Trends*, pp. 149–158 (2009)
26. Russell, J.: A circumplex model of affect. *JPSP* 39, 1161–1178 (1980)
27. Wallace, F.L., Flanery, J.M., Knezek, G.A.: The effect of subliminal help presentations on learning a text editor. *Inf. Process. Manage.* 27, 211–218 (1991)

Detecting Learner Frustration: Towards Mainstream Use Cases

Judi McCuaig, Mike Pearlstein, and Andrew Judd

Department of Computing and Information Science
University of Guelph
{judi,mpearlst,judda}@uoguelph.ca

Abstract. When our computers act in unexpected (and unhelpful) ways, we become frustrated with them. Were the computers human assistants, they would react by doing something to mitigate our frustration and increase their helpfulness. However, computers typically do not know we are frustrated. This paper presents research showing that user frustration can be detected with good accuracy (84%) using only two types of input data (head tilt and pupil dilation). We also show that reasonable accuracy (73%) can be achieved using only information about head tilt. We then propose how such technology could be employed to reduce learner frustration in adaptive tutoring applications.

Keywords: frustration detection, neural network, head and eye tracking.

1 Introduction

Human interaction with computers is still predominately unidirectional because, while users are usually sensitive to the computer's state during a sequence of interactions, the computer is generally unaware of the user's mental and emotional state given the same set of interactions. In recent years, the capacity for computer programs to detect the affective states of human operators has greatly increased, paving the way for programs that react not only to input but also to the user's emotions. A user's emotional state, or affect, can be reasoned about using one, or many, types of sensed data including natural language, facial features, posture and head position, galvanic skin responses and interactions with the program [1-4]. At present, much of this data must be collected using specialized hardware and software. This research proposes that software can be given the capability of observing user affective state in a limited fashion using unobtrusive off-the-shelf sensors and algorithms that run in real time on consumer hardware.

Little evidence exists to support, or refute, the theory that a computer program that can react appropriately to user emotions will be easier or more effective to use. However, within the field of educational computing, there is evidence suggesting that learners who are feeling positive are more motivated and engaged than students who experience negative emotions [5]. It follows then, that software used by learners might be more effective and/or usable as a learning tool if it was able to react to the learner's affect. Frustration is one of the affective states that has been shown to be present during learning [5].

Emotions, positive or negative, occur in context, and, while humans are good at detecting emotion, they are not as good at accurately describing the characteristics of a single emotion in a repeatable fashion, such as is needed to make mechanical measurements to catalog emotional responses [6]. A physical observation of a person might show an elevated heart rate, a high tone of voice and tense muscles, and that person could be experiencing any of several emotions. The human nervous system responds to stress by increasing skin conductivity, increasing heart rate or tensing muscles, and these symptoms are present when someone is experiencing a stressful emotion [7]. Within the context of learning, frustration is a common emotion. A certain amount of frustration and confusion is normal when learning new skills and concepts, but too much frustration will lead to a negative experience for the learner. Because frustration is a negative feeling arising from a failure to complete a defined task, the frustration level of a learner can be reasoned about by observing the human characteristics associated with negative emotions.

The research reported here shows that frustration can be accurately detected using visual information about the user. The research lays the groundwork for the creation of a tutoring component that detects learner frustration on-the-fly and alerts the underlying tutoring system of the learner's state of mind. In this paper, we discuss the data collection exercise and an experiment to determine the inputs that have the greatest influence on the accuracy of the frustration detection algorithm. We conclude the paper with a discussion about our current research examining the ability of commercial web cameras to obtain the necessary information about the user's head position and expression.

1.1 Frustration Detection

Recent advances in affect detection have been particularly promising for the detection of frustration. Researchers have obtained good results in detecting frustration with physiological factors by using modified sensors with reduced invasiveness [4]. Kapoor et al [3] used a variety of sensor types, including head tilt, skin conductance, and fidgeting, to detect frustration with 79% accuracy. The research showed that there exist non-verbal, observable, cues that can be used to predict the point at which a learner will self-report frustration. Conati and Maclaren presented a probabilistic model for identifying which one of several (possibly overlapping) emotions a learner is experiencing based on the achievement of learner goals. The model was reasonably successful in distinguishing between emotions and it was also able to provide explanations for the emotions [8].

Frustration can also be measured using physical measures such as head tilt, pupil dilation, posture and saccade timing. D'Mello and Graesser [9] use body posture to detect emotions. Interestingly they found that frustration was the hardest emotion to detect from visual cues (body cues or facial cues) and that it was best detected by mining the log files of the tutoring system being used. The researchers hypothesize that learners try to hide feelings of frustration because of cultural norms that encourage the disguise of negative emotions. Kapoor and Picard [10] obtained consistent results in detecting frustration using specialized tracking cameras. However, the computational requirements needed to process the raw data from this

experiment were too large for real-time frustration detection. In summary, current research shows that physiological measures are unnecessary for successful detection of frustration [1], but in most cases the data required (especially eye gaze and posture) can't be gathered without sophisticated hardware.

All researchers comment on the challenge of creating a ground-truth data set for evaluating affect-detection. The first difficulty is to elicit emotional responses in subjects that are authentic [11]. The second challenge is to identify the point in time that the emotion occurred [9]. Self-reported data is often unreliable, especially for negative emotions that learners may wish to cover up. Observers or judges can often report on whether the emotion is positive or negative, but sometimes disagree on the specific emotion being displayed. Peer judgments are also often used to try to identify points of significant emotion in a sequence of computer interaction. There is little agreement about what process is best, even within a single research paper.

2 Mainstream, Non-intrusive, Frustration Detection

The goal of this research is to develop a mechanism for detecting frustration that is a) noninvasive, b) computationally cheap and c) achievable with non-specialized hardware. A web camera can be employed to observe the user, potentially eliminating the need for sophisticated devices such as GSR sensors, gaze trackers, and keystroke monitors. Using visual cues given by 'watching' the user's head and face can greatly reduce the invasiveness of sensors. For this research, an eye-tracking system was used to capture pupil dilation, perclos, saccade and head tilt information about computer users. Computational requirements can be reduced by simplifying the algorithm or often by reducing the number of inputs that must be manipulated. A contribution of this research is a methodical examination of the impact of each of the captured inputs on the accuracy of the frustration detection algorithm.

2.1 Creating the Data Set

The first step in this research was the creation of a data set containing data for users in both frustrated and unfrustrated states. In addition to raw data, the data set required identified frustration points (ground truth) to allow evaluation of the detection algorithms. Data was captured while participants played a simple video game (Tetris) and users identified frustrating points in their interactions after the fact by watching a video of their game play and clicking on points that they remembered as being frustrating.

Video games have been shown to be frustrating, and to provide affective (emotional) responses in those who play them [12]. A simple video game was selected to ensure that most participants would be successful quickly and to help ensure that the affective responses of participants would be genuine. Participants first played an unmodified version of Tetris while the eyetracker was collecting data about their eye and head positions. This data was labeled as 'unfrustrated' data. Subsequently, each participant played a modified (frustrating) Tetris game. Several elements of the Tetris game were modified in ways that conflicted with participants' expectations for the game. The modified game was visually identical to the original

game and participants were told that they were playing the same game a second time. Again the eye tracking system was used to collect data about eye and head positions.

The data collected from the second Tetris session was used to provide samples of data from frustrated users. To confirm that the two sets of data represented different states of frustration, participants were given the Nasa Task Load Index (TLX¹) after each session of Tetris. In all cases users reported higher cognitive load during the second session, and nearly all reported higher levels of frustration. The TLX scores after playing the Normal Tetris games showed a low incidence of frustration – median score 7 – compared to a high median score of 31 for the TLX results taken after participants played the Modified Tetris game.

To create a ground-truth set of data about frustrated episodes, each user was asked to review a video of their gameplay after the second Tetris session. During that review, users were asked to identify points in the session that were especially frustrating by clicking the mouse when the video reached a point that they remembered as being frustrating. The click points were marked in the video and the time-stamps were later extracted. The extracted time-stamps of these self-identified frustration points were used to form a gold-standard data set to which the output of the frustration detection algorithm was compared.

Unfortunately, due to inexperience with the eyetracking system, many of the data files collected were either empty or consisted of single, unvarying (and obviously invalid) values. We believe that some of the unusable data was caused by mis-configuration of the hardware, some was a result of users moving too much when playing the game (the eye tracking system used tracks head movement based on facial features and is only accurate when it has a good view of the face), and some of the nonsensical values may have been because prescription lenses affected the accuracy of the pupil dilation readings. The experiment was conducted with the entire subset of data that contained valid values (i.e. not an entire set of zeros). We intend to repeat the experiment over the summer to confirm our results with a larger sample of data.

2.2 Training and Testing the Neural Network

It was not a goal of this research to compare the abilities of different algorithms and approaches for frustration detection so a single approach was selected based only on the necessary characteristics to solve the classification problem. The frustration detection algorithm needed to accept multiple inputs, consider the inputs over time, and arrive at a single answer (frustrated/not frustrated). An Elman network was selected for this research because it was capable of meeting all of these requirements. The network itself was implemented using the Fast Artificial Neural Network (FANN) version 2.0.0².

This neural network was used to determine exactly how few inputs would still give accurate frustration detection. The experiment was conducted in phases, where each phase used a different number and configuration of inputs. For each phase of the experiment, the network was trained, using 10-second windows. Training was conducted using one subset of the collected data and testing was conducted using a different set of untrained data.

¹ <http://humansystems.arc.nasa.gov/groups/TLX/>

² <http://leenissen.dk/fann/>

Training consisted of data representing both frustrated and non-frustrated time points. Non-frustrated time points were selected randomly from data collected during participant's first Tetris session. Frustrated time-points were taken from the data collected while the participant played the modified Tetris game, and were aligned with the self-reported frustration points. The first middle and last instances of self-reported frustration were used for training, and the other instances were used for testing. The trained network was then evaluated using 10-second intervals of data that had not been used for training. To determine accuracy, the output of the network was compared to participant-reported windows of frustration.

The SeeingMachines³ eye tracker used for this experiment provides head position information such as tilt and orientation, information about facial features such as mouth position, and information about eye movements and state such as gaze location, perclos, saccade and blink rate. Given that the long-term goal of this research is to negate the need for sophisticated hardware, the inputs to the neural network were restricted to inputs that might reasonably be obtainable using a off the shelf web camera. The inputs selected were; perclos, saccade, pupil dilation (both eyes), and head tilt (x,y and z). These choices align with research showing that people tend to have dilated pupils, heads that tilt left and jumpy eye movements in response to frustrating input [7].

Experiment Phase 1- All Inputs: In the first phase of the experiment the network input layer consisted of 14 neurons. There were 2 hidden layers and an output layer organized in a funnel. The first layer contained 10 neurons, or two thirds of the 14 inputs in the first layer. The second hidden layer contained 7 neurons, or two thirds of the 10 neurons used in the first hidden layer. Finally, the output layer contained two neurons corresponding to 1 and 0 showing our results. The output weightings were adjusted to reflect a graduate build up of frustration.

Since a network output of 0 represents absolutely no frustration and an output of 1 represents total frustration, it was necessary to consider a range of outputs as being frustrated or not frustrated. Participants were unlikely to be completely frustrated, thus it seemed unreasonable to detect only complete frustration. As a result, network outputs that were between 0 and 0.3 were classified as an example of non-frustration, while outputs that were between 0.7 and 1 were classified an example of frustration. Results that were between those two points (0.3 to 0.7) were discarded for this research as they were considered to represent a neutral affective state.

Testing data consisted of both frustrated and unfrustrated windows of data. Each window of frustrated data consisted of five seconds of data prior to the participant's self-reported frustration time and five seconds of data following the participant's self-reported frustration time. The non-frustrated data segments were selected randomly from the first Tetris sessions. Using all seven of the selected inputs, the neural network consistently detected user frustration with between 79% and 85% accuracy with an average accuracy of 81.8% and a median accuracy of 81.2%. The next phase of the experiment was to methodically reduce the number of inputs and discover which combination of inputs would result in accuracy at or near the results with all inputs.

³ <http://www.seeingmachines.com/>

Table 1. Categories of Input captured by the Eyetracker

Input	Definition
Perclos	Perclos or Percent Closed is a measure of how closed the eye is.
Saccade	A Saccade is a jump from one spot to another by the eye, a movement made frequently as we use our eyes. Saccade measures indicate the distance of the saccade.
Pupil Diameter	As an anxiety response, people dilate their pupils. Pupil diameter is measured in mm.
Head Tilt	Humans tilt their heads in a specific direction as an anxiety response. Head tilt is represented as angles in x,y, and z directions.

Experiment Phase 2- Input Reduction: The seven individual inputs used for this experiment actually fall into four categories: Saccades, Perclos, Pupil dilation, and Head Tilt. Rather than vary individual inputs, entire categories of input were removed and/or added in each sub-phase. Future experimentation will include the removal of individual inputs in specific categories. Details of these input categories can be found in the Table 1. The number of inputs was reduced gradually, beginning with Saccade inputs, then Perclos, then Pupil dilation, and finally Head Tilt.

An inspection of the data showed that the differences in saccade measurements between the first and second session of Tetris were low. In most cases the differences were less than 0.001mm (measured by averaging the saccade distances over ten seconds). As a result, saccades were the first data type removed from the input stream. An input type was removed by reducing the number of input neurons to the neural network to be double the number of remaining data inputs, and then reducing the size of the hidden layers to reflect the reduced input layer. For example, the network for using perclos, pupil dilation and head tilt (no saccades) had 12 input neurons with 8 neurons and 5 neurons in each of the hidden layers. Each modified network was trained and tested using the procedure that was described earlier. The accuracy of the frustration detection algorithm without the saccade data was nearly as good as the accuracy when using all of the data. The detection algorithm successfully detected frustration 79%-81% of the time with an average success rate of 80.3% and a median rate 80.6%.

The data about perclos was the next data removed from the network, leaving only head tilt and pupil dilation. The removal of the perclos data had little impact on the results from the detection algorithm, despite the psychology literature showing it as a stress and anxiety response [13]. Using head tilt and pupil dilation only, the detection algorithm was consistently between 80% and 85% accurate, with an average accuracy of 84.1 and a median accuracy of 83.4%. In general, the accuracy of the algorithm improved by removing the perclos data, by an average of 3.9%, when compared to the accuracy when just the saccade data was removed.

Pupil dilation was the next set of data removed from the detection algorithm, leaving only the head tilt data. Both pupil dilation and head tilt have been shown to be reliable indicators of frustration [3, 14], so a noticeable impact was expected in the accuracy of detection. When the detection algorithm was restricted to using just the x,y,z axis of head tilt as input, the accuracy was moderately good (70%-76% success, average:73.9%, mean: 74.4%), but was lower than the previous sub-phase when the pupil dilation was paired with the head tilt. When pupil dilation was the only input to the network, the accuracy of detection was also moderately good (70-75%, Average: 73.2%, Mean: 73.5%).

2.3 Discussion

A summary of the results of this experiment can be found in Table 2. The results show that a reasonable job of detecting user frustration can be accomplished using only information about head position or pupil dilation, while a good detection rate can be had if information about pupil dilation and head position are used together. These results represent a slight improvement over the results reported in previous research.

Table 2. Summary of Results

Factor	Detection Rate	Average	Median
All factors	79% to 85%	81.79%	81.20%
Perclos/HT/PD	79% to 81%	80.30%	80.60%
Pupil Dilation (PD)	70% to 75%	73.18%	73.50%
Head Tilt (HT)	70% to 76%	73.86%	74.40%
HT/ PD	82% to 85%	84.06%	83.40%

There are, however, limitations in this research that suggest further investigation is required before broad assertions of significance can be made. The process for creating a ground truth data set for comparison required participants to remember emotions after the fact, which may not have resulted in a completely accurate representation of participant frustration. The high percentage of unusable data created during data collection also creates concern about the broad applicability of these results. The eye tracking system has been reconfigured and recalibrated. A new set of data is being collected with the intention of rerunning the experiment to confirm the results. Experimenters will investigate the possibility of using an approach similar to that of Kapoor [3] to capture the ground-truth frustration points during the creation of this new data set.

The next two phases of this research examine the questions a) *Can the required data (head tilt and pupil dilation) be collected using a standard web camera?* (to satisfy the research goal of using consumer hardware) and b) *Will the Elman network recognize frustration in real time?*(to satisfy the goal of being computationally

feasible in real time). Both of these questions are the subject of current research by members of the research group. The remainder of this paper reports on preliminary findings.

3 Ongoing Work

Progress has been made in the use of web cameras for tracking the head movements of computer users, primarily in the areas of accessible interfaces and gaming. Many of the algorithms used in accessible interface applications make the assumption that the head is upright, or nearly so, and do only course estimates of head tilt. Waber, Magee and Betke [15] report on an image processing technique for detecting head tilt in real time from web camera images and showed that their process was robust enough for users to control a simple application using only head tilt. This algorithm produces a single angle for head tilt measurement (left/right). Since this research has shown that head tilt as a function of three angles can be used to detect frustration, the first step in understanding whether an algorithm such as the one reported above can be used is to determine if the frustration detection network can be successful using only reports of left/right tilt. That determination requires a relatively simple additional experiment in which the x,y and z coordinates for head tilt are used separately (or combined to form a single angle measurement). That experiment is planned as future work.

The detection of pupil dilation using web cameras is also possible, but is still quite invasive. For example, in one case the web camera must be very close to the user's face in order to detect the pupil size; in fact the software comes with the recommendation that the web camera is mounted to a piece of balsa wood held in the mouth of the participant⁴! Good detection rates for blink, eye movement and perclos have been achieved using web cameras [16] and researchers report that less calibration is required for tracking only pupil dilation [17], which should make pupil dilation detection an easier problem. We postulate that, since most web camera eye tracking efforts are concerned with using movement to interact with the computer, pupil dilation detection has simply not been addressed yet. This phase of our research program is actively looking at mechanisms for identifying pupil dilation using only a web camera and possibly infrared light sources. We anticipate building on the widely used OpenCV⁵ library.

Progress on the third goal for this research is anticipated in the near future. We expect that the same Elman network used for this research will easily achieve real time speeds for detection of frustration (excluding training time). An experiment to test that hypothesis is underway and results are expected mid-May. The experiment will provide 10 second windows of data to the network in a continuous stream and will examine the results to determine if the detection accuracy remains high, and then will compare the timing with the time stamps of the self-reported incidents of frustration to see if the recognition of frustration is quick enough to allow a hypothetical software system to react appropriately. Two variations on this experiment will be conducted if time allows. The first is to use overlapping 10 second windows of data to give to the network, which

⁴ <http://www.gazegroup.org/downloads/23-gazetracker>

⁵ <http://opencv.willowgarage.com/wiki/>

would give the network a point to classify every 5 seconds, but the data in the window would be 50% similar to the data in the window immediately previous. A second variation is to lower the recognition threshold from .7 to some lower number. The lowered threshold might allow the algorithm to anticipate frustration, rather than simply recognize it.

4 Summary

This research has shown that accurate detection of user frustration is possible using only two types of input: head tilt and pupil dilation. Both of these input types can be gathered through non-invasive cameras using unspecialized hardware, which sets the stage for detecting frustration on-the-fly. The capacity to detect user frustration is of particular interest to tutoring systems and adaptive help systems because such software is written with the goal of helping users learn something that is presently unknown to them. A tutoring system endowed with frustration-detection ability could react to learner state of mind and, potentially, improve the learning process. This research provides the foundation for such a system.

References

1. Asteriadis, S., Karpouzis, K., Kollias, S.: Feature Extraction and Selection for Inferring User Engagement in an HCI Environment. In: Asteriadis, S., Karpouzis, K., Kollias, S. (eds.) *Secondary Feature Extraction and Selection for Inferring User Engagement in an HCI Environment*, pp. 22–29. Springer, Heidelberg (2009)
2. Conati, C., Maclaren, H.: Empirically building and evaluating a probabilistic model of user affect. *User Modeling and User Adapted Interaction* 19, 267–303 (2009)
3. Kapoor, A., Bursleson, W., Picard, R.W.: Automatic prediction of frustration. *International Journal of Human-Computer Studies* 65, 724–736 (2007)
4. Klein, J., Moon, Y., Picard, R.: This computer responds to user frustration: Theory, design, and results. *Interacting with Computers* 14, 119–140 (2002)
5. Craig, S., Graesser, A.C., Sullins, J., Gholson, B.: Affect and learning: An exploratory look into the role of affect in learning. *Journal of Educational Media* 29, 9 (2004)
6. Barrett, L.: Are Emotions Natural Kinds? *Perspectives on Psychological Science* 1, 31 (2006)
7. Miller, N.I.: The frustration-aggression hypothesis. *Psychological Review* 48, 337–342 (1941)
8. Conati, C., Maclaren, H.: Empirically building and evaluating a probabilistic model of user affect. *User Modeling and User Adapted Interaction* 19, 36 (2009)
9. D’Mello, S., Graesser, A.C.: Automatic Detection of Learner’s Affect from Gross Body Language. *Applied Artificial Intelligence* 23, 26 (2009)
10. Kapoor, A., Picard, R.W.: A real-time head nod and shake detector. In: Kapoor, A., Picard, R.W. (eds.) *Secondary A real-time head nod and shake detector*, ACM, New York (2001)
11. Kapoor, A., Picard, R.W.: Multimodal Affect Recognition in Learning Environments. In: Kapoor, A., Picard, R.W. (eds.) *Secondary Multimodal Affect Recognition in Learning Environments*, p. 6. ACM, New York (2005)

12. Chumbley, J., Griffiths, M.: Affect and the computer game player: The effect of gender, personality, and game reinforcement structure on affective responses to computer gameplay. *Cyberpsychology & Behavior* 9, 308–316 (2006)
13. Nopsuwanchai, R., Noguchi, Y., Ohsuga, M., Kamakura, Y., Inoue, Y.: Driver-Independent Assessment of Arousal States from Video Sequences Based on the Classification of Eyeblink Patterns. In: Nopsuwanchai, R., Noguchi, Y., Ohsuga, M., Kamakura, Y., Inoue, Y. (eds.) 11th International IEEE Conference on Intelligent Transportation Systems, ITSC 2008, pp. 917–924 (2008)
14. Batmaz, I., Ozturk, M.: Using pupil diameter changes for measuring mental workload under mental processing. *Journal of Applied Sciences* 8, 68–76 (2008)
15. Waber, B., Magee, J., Betke, M.: Fast Head Tilt Detection for Human Computer Interaction. In: Waber, B., Magee, J., Betke, M. (eds.) *Secondary Fast Head Tilt Detection for Human Computer Interaction*, pp. 90–99 (2005)
16. Ince, I., Yang, T.-C.: A New Low-Cost Eye Tracking and Blink Detection Approach: Extracting Eye Features with Blob Extraction. In: Ince, I., Yang, T.-C. (eds.) *Secondary A New Low-Cost Eye Tracking and Blink Detection Approach: Extracting Eye Features with Blob Extraction*, pp. 526–533. Springer, Heidelberg (2009)
17. King, L.A.: Visual Navigation Patterns and Cognitive Load. In: King, L.A. (ed.) *Secondary Visual Navigation Patterns and Cognitive Load*, Springer, Heidelberg (2009)

Enhancing the Automatic Generation of Hints with Expert Seeding

John Stamper¹, Tiffany Barnes², and Marvin Croy³

¹ Carnegie Mellon University, Human-Computer Interaction Institute, Pittsburgh, PA
john@stamper.org

² University of North Carolina at Charlotte, Department of Computer Science, Charlotte, NC
tiffany.barnes@gmail.com

³ University of North Carolina at Charlotte, Department of Philosophy, Charlotte, NC
mjcroy@uncc.edu

Abstract. The Hint Factory is an implementation of our novel method to automatically generate hints using past student data for a logic tutor. One disadvantage of the Hint Factory is the time needed to gather enough data on new problems in order to provide hints. In this paper we describe the use of expert sample solutions to “seed” the hint generation process. We show that just a few expert solutions give significant coverage (over 50%) for hints. This seeding method greatly speeds up the time needed to reliably generate hints. We discuss how this feature can be integrated into the Hint Factory and some potential pedagogical issues that the expert solutions introduce.

Keywords: Educational data mining, Markov decision process.

1 Introduction

The goal of the Hint Factory is to make intelligent tutors more accessible by simplifying their creation using educational data mining and machine learning techniques. In particular, we seek a path for educators to add intelligent tutoring capabilities to existing computer aided instruction (CAI) without significantly rewriting the existing software. The Hint Factory is a novel technique that uses a Markov decision process (MDP), created from past student data, to generate specific contextualized hints for students using CAI.

We seek to make our data-driven methods effective quickly. One criticism of data-driven techniques is the amount of time it takes to achieve results for a new problem with no data. Although we have previously addressed this issue with a cold start analysis [1], this research provides an additional method to speed up hint giving capabilities. For this method, an expert (or experts) “seeds” the matrix by completing examples to new problems and using these examples to create the initial MDPs. The experiments presented here focus on how well hints can be provided from an initial expert seeding of the MDP. We hypothesized that hints derived from expert solutions could be used to provide hints in 50% of historical student solution steps. The expert time needed is quite low to achieve this level of hint coverage.

Our primary research implementation of the Hint Factory has been in a tutor to teach deductive logic in Philosophy and discrete mathematics at the college level [5]. In addition to the seeding experiment, this analysis examines additional problems in the logic domain order to further validate our previous work [1]. The analysis of the additional logic problems resulted in similar hint coverage to the problem (NCSU Proof 1) previously studied. This confirms our belief that the method is robust and effective in the logic domain.

2 Background and Related Work

Historically, the research and development of intelligent tutors have relied on subject area experts to provide the background knowledge to give hints and feedback. Both cognitive tutors and constraint based tutors rely on “rules” that experts create [9]. This is a time consuming process, and requires the experts to not only understand the subject material, but also to understand the underlying processes used to give help and feedback. We believe that the development of intelligent tutors can be enhanced by using data collected from students solving problems. The amount of data being collected from CAI continues to grow at an exponential rate. Large data repositories like the PSLC DataShop have been created to store and analyze this data [7]. Data-driven methods applied to such large data repositories can enable the rapid creation of new intelligent tutoring systems, making them accessible for many more students.

Others have used collected student data with machine learning to improve tutoring systems. In the ADVISOR tutor, machine learning was used to build student models that could predict the amount of time students took to solve arithmetic problems, and to adapt instruction to minimize this time while meeting teacher-set instructional goals [4]. Student data has been used to build initial models for an ITS, in an approach called Bootstrapping Novice Data (BND) [10]. Although the BND approach saves time in entering example problems, it still requires expert instructors and programmers to create a tutor interface and annotate the extracted production rules with appropriate hints. Similar to the goal of BND, we seek to use student data to directly create student models for an ITS. However, instead of using student behavior data to build a production rule system, our method generates MDPs that represent all student approaches to a particular problem, and use these MDPs directly to generate hints. RomanTutor is a ITS developed to teach astronauts to operate a robot arm on the International Space Station [11]. This tutor uses sequential pattern mining (SPM) over collected data to find the best sequence of steps at any given point. In this ill-defined domain, data mining has proved to be an effective way to provide feedback where the number of possible combinations would be too immense for experts to cover. SimStudent is an agent based tool for building student knowledge models by example [12]. SimStudent has been used with student log data to build a model that predicts student knowledge.

Our research using visualization tools to explore generated hints based on MDPs extracted from student data verified that the rules extracted by the MDP conformed to expert-derived rules and generated buggy rules that surprised experts [3]. Croy, Barnes, and Stamper applied the technique to visualize student proof approaches to allow

teachers to identify problem areas for students. Barnes and Stamper demonstrated the feasibility of this approach by extracting MDPs from four semesters of student solutions in a logic proof tutor, and calculated the probability that hints could be generated at any point in a given problem [1]. Our results indicated that extracted MDPs and our proposed hint-generating functions were able to provide hints over 80% of the time. The results also indicated that we can provide valuable tradeoffs between hint specificity and the amount of data used to create an MDP. The MDP method was successfully implemented into the Deep Thought logic tutor as part of the Hint Factory in a live classroom setting [2]. Fossati and colleagues have used the MDP method in the iList tutor used to teach linked lists and deliver “proactive feedback” based on previous student attempts [6].

One ITS authoring tool, CTAT, was given a feature to use demonstrated examples to learn ITS production rules [8]. In these tools, teachers work problems in what they predict to be frequent correct and incorrect approaches, and then annotate the learned rules with appropriate hints and feedback. In many ways this is similar to the seeding approach presented here, but in our approach the expert need not supply the hints. Additionally, the example tracing tutors will only have the knowledge that the expert has added, while our methods would allow the tutor to continue to improve as additional expert or student problem attempts are added to them model. Finally, our method computes a value for problem states automatically, and uses this value to make decisions on which path to suggest even when multiple choices are reasonable. This ability to differentiate between several good solutions based on the specific context of student’s current state remains the strong point of the Hint Factory.

3 Markov Decision Processes to Create Student Models

The Hint Factory consists of the MDP generator and the hint provider. The MDP generator is an offline process that assigns values to the states that have occurred in student problem attempts. These values are then used by the hint provider to select the next “best” state at any point in the problem space.

A Markov decision process (MDP) is defined by its state set S , action set A , transition probabilities $T: S \times A \times S \rightarrow [0,1]$, and a reward function $R: S \times A \times S \rightarrow \mathfrak{R}$ [13]. The goal of using an MDP is to determine the best policy, or set of actions students have taken at each state s that maximize its expected cumulative utility (V -value) which corresponds to solving the given problem. The expected cumulative value function can be calculated recursively using equation (1). For a particular point in a student’s logic proof, a state consists of the list of statements generated so far, and actions are the rules used at each step. Actions are directed arcs that connect consecutive states. Therefore, each proof attempt can be seen as a graph with a sequence of states connected by actions.

We combine all student solution graphs into a single graph, by taking the union of all states and actions, and mapping identical states to one another. Once this graph is constructed, it represents all of the paths students have taken in working a proof. Next, value iteration is used to find an optimal solution to the MDP. For the experiments in this work, we set a large reward for the goal state (100) and penalties for

incorrect states (10) and a cost for taking each action (1), resulting in a bias toward short, correct solutions such as those an expert might derive. We apply value iteration using a Bellman backup to iteratively assign values $V(s)$ to all states in the MDP until the values on the left and right sides of equation (1) converge [13]. The equation for calculating the expected reward values $V(s)$ for following an optimal policy from state s is given in equation (1), where $R(s,a)$ is the reward for taking action a from state s , and $P_a(s, s')$ is the probability that action a will take state s to state s' . $P_a(s, s')$ is calculated by dividing the number of times action a is taken from state s to s' by the total number of actions leaving state s .

$$V(s) := \max_a \left(R(s,a) + \sum_{s'} P_a(s,s') V(s') \right) \quad (1)$$

Once value iteration is complete, the optimal solution in the MDP corresponds to taking an expert-like approach to solving the given problem, where from each state the best action to take is the one that leads to the next state with the highest expected reward value [2].

4 Method

We use historical data to estimate the availability of hints using the MDP and seeding approaches. We performed this experiment using student attempts at the Proof Tutorial problems 1-4, as given in Table 1, in the NC State University discrete math course from fall semesters 2003-2006. The givens are the premises that students will use to prove the conclusion in the tutorial. Before using the Proofs Tutorial as homework, students attend several lectures on propositional logic and complete fill-in-the-blank proofs.

Table 1. Description of Proofs Tutorial problems 1 through 4

Problem	Givens	Conclusion
1	If A then B, If C then D, not(If A then D)	B and not C
2	If A then B, If not C then D, not B or not D	If A then C
3	If (B or A) then C	If A then (If B then C)
4	A or (If B then C), B or C, If C then A	A

We generated an MDP for each semester of data separately, and Table 2 shows the number of states generated and number of total moves generated from each problem during each of the four semesters. The number of states represents the number of unique steps that were seen over all problem attempts, while the number of moves represents all student steps or state-action pairs. The number of moves gives a more accurate reflection of class behavior, and comparing states to moves gives a notion of how much repetition occurs in the dataset. Note that problem 1 was used in the

original validation experiments [1]. From the table, several clear trends can be seen. First, the total number of attempts, states, and moves are lower in the Fall 2005 and significantly lower in Fall 2006 semesters. Second, problem 4 has significantly fewer attempts in every semester when compared to the others. According to the course instructor, problem 4 is the hardest problem. The seeding data was provided by two subject area experts, who worked each problem several times, but for less than one hour per problem. An overview of their problem attempts is given in Table 3.

Table 2. Semester data, including attempts, moves, and states in the MDP for each semester

Problem	Semester	# Attempts	MDP states	# Moves
1	f3	172	206	711
1	f4	154	210	622
1	f5	123	94	500
1	f6	74	133	304
2	f3	138	162	628
2	f4	142	237	752
2	f5	105	122	503
2	f6	63	103	279
3	f3	139	145	648
3	f4	145	184	679
3	f5	113	103	577
3	f6	71	94	372
4	f3	103	46	166
4	f4	59	63	103
4	f5	34	30	48
4	f6	33	20	41

Table 3. Expert example seeding attempts, moves, and states for each problem

Problem	# Attempts	MDP states	# Moves
1	3	10	19
2	4	12	27
3	2	15	21
4	3	8	20

To verify our previous results for testing hint availability [1], we performed a cross-validation study, with each semester used as a test set while the remaining semesters are used in training sets for MDPs. Hints are available for a particular state in the test set if the MDP contains that state with a path to the goal state. We count these matching states for each move as “move matches.” Table 4 shows the average percent

move matches between each semester and the remaining combinations of training sets using one, two, and three semesters of data to construct MDPs. On average, one-semester source MDPs match 71.46% of the valid moves in a new semester of data. With two semesters of data the average move coverage reaches 77.32% for a 5.86% marginal increase. Adding a third semester of data results in an average coverage of 79.57%, a 2.25% marginal increase over two semesters. All the individual problems show a similar curve where the marginal return decreases after each subsequent semester. For Problem 4 the total percentage of move matches is approximately 15-20% lower than the other problems, and this occurs since there are significantly fewer attempts on this more difficult problem.

Table 4. Average % move matches across problems comparing test sets and MDPs

Problem	1-sem. MDPs	2-sem. MDPs	3-sem. MDPs
1	72.79%	79.57%	82.32%
2	75.08%	80.58%	82.96%
3	79.01%	83.35%	84.89%
4	58.94%	65.77%	68.09%
Average	71.46%	77.32%	79.57%

Table 5. Average % unique state and move matches for seeded and 1-semester MDPs

Problem	Unique state matches		Move matches	
	Seeded MDP	1-sem. MDPs	Seeded MDP	1-sem MDPs
1	6.22%	34.55%	62.08%	72.79%
2	11.40%	34.60%	29.82%	75.08%
3	7.69%	33.36%	53.33%	79.01%
4	12.46%	23.45%	26.57%	58.94%
Average	9.44%	31.49%	42.95%	71.46%

Table 3 shows characteristics of the expert examples used for seeding the problem MDPs. Between two and four attempts were available for each of the problems and these attempts generated between 8 and 15 total states. Table 5 shows the results of comparing each semester of test data with the seeded MDPs and one-semester MDPs. Especially in problems 1 and 3, when compared to the semester data, the seeded MDP states were “high impact” states, which included the most used paths to solve the problems by the students. Comparing the unique state matches to move matches shows that although the seeded MDPs match only a small percentage of unique student problem states, they match a lot of the moves taken by students. It is interesting to note the move matches vary much wider than a semester of MDP data. This should be expected considering the small number of seeding attempts used. In fact, with further analysis of the individual problems we see for problems 1 and 3 there are two

common solutions, that correspond to the expert seeds, resulting in high move coverage percent rates (62.08% and 53.33% respectively), while problems 2 and 4 have more than two common solutions, which results in much lower, but still promising move coverage considering the small number of expert seed attempts.

5 Revisiting the “Cold Start” Problem

We previously explored how quickly an MDP can be used to provide hints to new students, or in other words, how long it takes to solve the cold start problem, for problem 1 [1]. In this his experiment we compare hint availability for incrementally constructed MDPs starting with no data to those starting with seed data. In both cases, hint availability is calculated for the current student attempt, and their states are then added to the MDP. For one trial, the method is given in Table 6. In this experiment, we calculate the hint availability (move matches) for each consecutive student with seeded and non-seeded MDPs. We repeat this process for 100,000

Table 6. Method for one trial of the cold-start simulation

1. Let Test = {all 523 student attempts }
2. Randomly choose and remove the next attempt a from the Test set.
3. Add a’s states and recalculate the MDP.
4. Randomly choose and remove the next attempt b from the Test set.
5. Compute the number of matches between b and MDP.
6. If Test is non-empty, then let a:=b and go to step 3. Otherwise, stop.

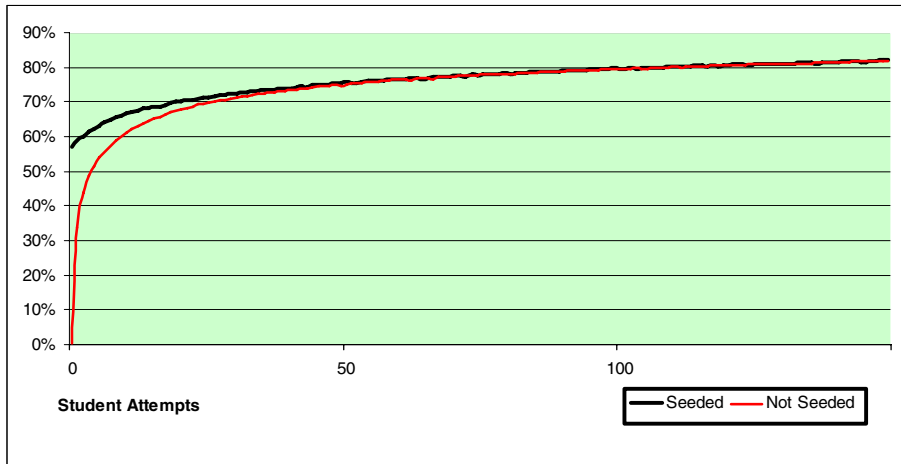


Fig. 1. Percent hints available as attempts are added to the MDP, over 100,000 trials for Problem 3

trials and plot the resulting hint availability curves for problem 3 in Figure 1. The curves for problems 1-4 were all similar, so we present only problem 3 here. Figure 1 shows that seeding shifts the initial starting point of hint availability from 0 to over 50%, giving a boost at the start. By 50 attempts the seeded set is just a few attempts ahead and by 100 attempts the 2 graphs are the same. This shows that seeding helps initial hint coverage, avoiding the steep wait for significant hint coverage when using incrementally constructed MDPs for hints. We note that in the initial boost, the seeded problems are covering “high impact” states, or those that are very frequent in the student data sets.

Table 7. Number of attempts needed to achieve threshold % hints levels for seeded and unseeded MDPs constructed incrementally. Note that for problems 1 and 3, 50% hint coverage was achieved with seeds alone.

Problem		50%	55%	60%	65%	70%	75%	80%	85%	90%
1	Not seeded	8	11	14	20	30	46	80	154	360
1	Seeded	seeds	seeds	4	8	21	46	80	155	360
2	Not seeded	9	15	24	36	59	88	149	286	*
2	Seeded	2	3	16	22	46	80	146	286	*
3	Not seeded	5	7	10	16	27	50	110	266	*
3	Seeded	seeds	1	3	8	20	48	110	266	*
4	Not seeded	25	31	54	82	*	*	*	*	*
4	Seeded	12	22	53	80	*	*	*	*	*

(* means the method did not reach this percentage)

Table 7 shows the number of attempts needed to achieve hint percentage thresholds with and without seeding for each of problems 1-4. Again we see that the seeding of each problem gives an initial boost that fades over time as more student attempts are added, which confirms our hypothesis.

6 Conclusions and Future Work

The main contribution of this paper is to show how expert seeding can enhance the automatic generation of hints by reducing the amount of student data initially needed to effectively deliver hints. Although we believe that our data-driven method already ramps up quickly [1], seeding using expert examples enhances our ability to give hints. For the seeding problems 100% of all the seeded states appeared in each semester of data. Obviously, the educators know the most common solutions to the problems and by seeding the MDPs they can quickly get this data included to help jump-start the hint giving process. Additionally, the experiments presented here replicate and further validate our earlier work in solving the cold start problem.

We do see a few issues with seeding. One issue with using experts to seed the MDP for hint generation is that experts may unintentionally miss a very important solution to the problem. Students who try to solve the problem in this way would not receive hints and therefore may believe that they are doing something wrong. This

problem, however, exists in traditional intelligent tutors as well. Further, the solutions that the expert provides will likely become more popular, since students receiving hints will likely use similar approaches to the experts. This is not necessarily a bad thing, but it could limit the ability of MDPs to provide broad coverage of the student solution space – since student solutions might be more limited if they make heavy use of seeded hints. Seeding would likely reinforce the expert solution for a long time to come even as additional data is acquired. To alleviate this problem the instructors can vary which problems receive hints so that clean data with no hints can be collected on every problem at some point. Alternatively, as enough student attempts are added to the MDPs to generate hints, expert solutions could be removed from the data set to promote diversity in student answers. The seeding approach is very similar to the Bootstrapping Novice Data discussed in the related work section, and we believe it can be useful in many data-driven methods for generating intelligent tutoring capabilities.

In our current and future work, we are using machine learning methods to analyze our MDPs for problem structure and for generating new problems of similar difficulty. We are also building tools for teachers and researchers to visualize MDPs and allow teachers to write their own hints, and modify how the MDPs are used in generating hints.

References

1. Barnes, T., Stamper, J.: Toward Automatic Hint Generation for Logic Proof Tutoring Using Historical Student Data. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 373–382. Springer, Heidelberg (2008)
2. Barnes, T., Stamper, J., Lehmann, L., Croy, M.: A Pilot Study on Logic Proof Tutoring Using Hints Generated from Historical Student Data. In: Baker, R., Barnes, T., Beck, J. (eds.) Proceedings of the 1st International Conference on Educational Data Mining (EDM 2008), Montreal, Canada, pp. 197–201 (2008)
3. Barnes, T., Stamper, J.: Toward the extraction of production rules for solving logic proofs. In: Proc. 13th Intl. Conf. on Artificial Intelligence in Education, Educational Data Mining Workshop, Marina del Rey, CA (2007)
4. Beck, J., Woolf, B.P., Beal, C.R.: ADVISOR: A Machine Learning Architecture for Intelligent Tutor Construction. In: 7th National Conference on Artificial Intelligence, pp. 552–557. AAAI Press / The MIT Press (2000)
5. Croy, M., Barnes, T., Stamper, J.: Towards an Intelligent Tutoring System for propositional proof construction. In: Brey, P., Brügge, A., Waelbers, K. (eds.) European Computing and Philosophy Conference, pp. 145–155. IOS Publishers, Amsterdam (2007)
6. Fossati, D., Di Eugenio, B., Ohlsson, S., Brown, c., Chen, L., Cosejo, D.: I learn from you, you learn from me: How to make iList learn from students. In: Dimitrova, V., Mizoguchi, R., Du Boulay, B., Graesser, A. (eds.) Proc. 14th Intl. Conf. on Artificial Intelligence in Education, AIED 2009, Brighton, UK, pp. 186–195. IOS Press, Amsterdam (2009)
7. Koedinger, K.R., Baker, R.S.J.D., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the EDM community: The PSLC DataShop. In: Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.J.D. (eds.) Handbook of Educational Data Mining, CRC Press, Boca Raton (in press) (to appear)

8. Koedinger, K.R., Aleven, V., Heffernan, T., McLaren, B., Hockenberry, M.: Opening the door to non-programmers: Authoring intelligent tutor behavior by demonstration. In: 7th Intelligent Tutoring Systems Conference, Maceio, Brazil, pp. 162–173 (2004)
9. Mitrovic, A., Koedinger, K., Martin, B.: A comparative analysis of cognitive tutoring and constraint-based modeling. *User Modeling*, 313–322 (2003)
10. McLaren, B., Koedinger, K., Schneider, M., Harrer, A., Bollen, L.: Bootstrapping Novice Data: Semi-automated tutor authoring using student log files. In: Proc. Workshop on Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes. 7th Intl. Conf. Intelligent Tutoring Systems (ITS 2004), Maceió, Brazil (2004)
11. Nkambou, R., Mephu Nguifo, E., Fournier-Viger, P.: Using Knowledge Discovery Techniques to Support Tutoring in an Ill-Defined Domain. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 395–405. Springer, Heidelberg (2008)
12. Matsuda, N., Cohen, W.W., Sewall, J., Lacerda, G., Koedinger, K.R.: Predicting students performance with SimStudent that learns cognitive skills from observation. In: Luckin, R., Koedinger, K.R., Greer, J. (eds.) Proceedings of the international conference on Artificial Intelligence in Education, pp. 467–476. IOS Press, Amsterdam (2007)
13. Sutton, S., Barto, A.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)

Learning What Works in ITS from Non-traditional Randomized Controlled Trial Data

Zachary A. Pardos*, Matthew D. Dailey*, and Neil T. Heffernan

Worcester Polytechnic Institute
{zpardos,mdailey,nth}@wpi.edu

Abstract. The traditional, well established approach to finding out what works in education research is to run a randomized controlled trial (RCT) using a standard pretest and posttest design. RCTs have been used in the intelligent tutoring community for decades to determine which questions and tutorial feedback work best. Practically speaking, however, ITS creators need to make decisions on what content to deploy without the benefit of having run an RCT in advance. Additionally, most log data produced by an ITS is not in a form that can easily be evaluated with traditional methods. As a result, there is much data produced by tutoring systems that we would like to learn from but are not. In prior work we introduced a potential solution to this problem: a Bayesian networks method that could analyze the log data of a tutoring system to determine which items were most effective for learning among a set of items of the same skill. The method was validated by way of simulations. In this work we further evaluate the method by applying it to real world data from 11 experiment datasets that investigate the effectiveness of various forms of tutorial help in a web based math tutoring system. The goal of the method was to determine which questions and tutorial strategies cause the most learning. We compared these results with a more traditional hypothesis testing analysis, adapted to our particular datasets. We analyzed experiments in mastery learning problem sets as well as experiments in problem sets that, even though they were not planned RCTs, took on the standard RCT form. We found that the tutorial help or item chosen by the Bayesian method as having the highest rate of learning agreed with the traditional analysis in 9 out of 11 of the experiments. The practical impact of this work is an abundance of knowledge about what works that can now be learned from the thousands of experimental designs intrinsic in datasets of tutoring systems that assign items in a random order.

Keywords: Knowledge Tracing, Item Effect Model, Bayesian Networks, Randomized Controlled Trials, Data Mining.

1 Introduction

The traditional, well-established approach to finding out what works in an intelligent tutoring system is to run a randomized controlled trial (RCT) using a standard pretest

* National Science Foundation funded GK-12 Fellow.

and posttest design. RCTs have been used in the intelligent tutoring systems (ITS) community for decades to determine best practices for a particular context. Practically speaking, however, ITS creators need to make decisions on what content to deploy without the benefit of having run an RCT in advance. Additionally, most log data produced by an ITS is not in a form that can easily be evaluated with traditional hypothesis testing such as learning gain analysis with t-tests and ANOVAs. As a result, there is much data produced by tutoring systems that we would like to learn from but are not. In prior work [1] we introduced a potential solution to this problem: a Bayesian networks method that could analyze the log data of a tutoring system to determine which items were most effective for learning among a set of items of the same skill. The method was validated by way of simulations with promising results but had been used with few real world datasets. In this work we further evaluate the method by applying it to real world data from 11 experiment datasets that investigate the effectiveness of various forms of tutorial help in a web based math tutoring system. The goal of the method was to determine which questions and tutorial strategies cause the most learning. We compare these results with results from a more traditional hypothesis testing analysis, adapted to our particular datasets.

1.1 The ASSISTment System – A Web-Based Tutoring System

Our datasets consisted of student responses from The ASSISTment System, a web based math tutoring system for 7th-12th grade students that provides preparation for the

state standardized test by using released math items from previous state tests as questions on the system. Figure 1 shows an example of a math item on the system and tutorial help that is given if the student answers the question wrong or asks for help. The tutorial help assists the student in learning the required knowledge by breaking each problem into sub questions called scaffolding or giving the student hints on how to solve the question. A question is only marked as correct if the student answers it correctly on the first attempt without requesting help.

The screenshot displays the ASSISTment System interface. At the top, a red box highlights the original question: "Triangles ABC and DEF are congruent. The perimeter of triangle ABC is 23 inches. What is the length of side DF in triangle DEF?". Below this are two triangles, ABC and DEF, with side lengths and angles labeled. A "Request Help" button is visible. Below the question, a yellow box highlights the 1st scaffold: "Which side of triangle ABC has the same length as side DF of triangle DEF?". A "Request Help" button is also present. Below the scaffold, a green box highlights a hint: "Let's make sure you understand what corresponding sides are. In this picture the corresponding sides are marked. Does this help you?". The hint shows two triangles with corresponding sides marked in red and green. Below the hint, a blue box highlights a buggy message: "Side AB corresponds to side DE of triangle DEF, not DF. Try again, please.". A "Request Help" button is also present.

Fig. 1. An example of an ASSISTment item where the student answers incorrectly and is given tutorial help

1.2 Item Templates in the ASSISTment System

Our mastery learning data consists of responses to multiple questions generated from an item template. A template is a skeleton of a problem created by a content developer in our web based builder application. For example, the template would specify a Pythagorean Theorem problem, but without the numbers for the problem filled in. In this example the problem template could be: “What is the hypotenuse of a right triangle with sides of length X and Y?”

where X and Y are variables that will be filled in with values when questions are created from the template. The solution is also dynamically determined from a solution template specified by the content developer. In this example the solution template would be, “Solution = $\sqrt{X^2+Y^2}$ ”. Ranges of values for the variables can be specified and more advance template features are available to the developer such as dynamic graphs, tables and even randomly selected cover stories for word problems. Templates are also used to construct the tutorial help of the template items. Items created from these templates are used extensively in the mastery learning problem sets as a pragmatic way to provide a high volume of items for students to practice particular skills on.

2 The Item Effect Model

We use a Bayesian Networks method called the Item Effect Model [1] for our analysis. This model is based on Knowledge tracing [3] which has been the leading modeling technique used in tracking student knowledge in intelligent tutoring systems for over a decade. Knowledge tracing assumes that the probability of learning on each item, or piece of learning content, is the same. However, the fact that certain content has been shown to be more effective than other content is reason enough to question this assumption. Knowledge tracing is a special case of the Item Effect model, which allows different items to cause different amounts of learning, including the same amount. This is scientifically interesting and practically useful in helping ITS designers and investigators better learn what type of tutoring can maximize student learning.

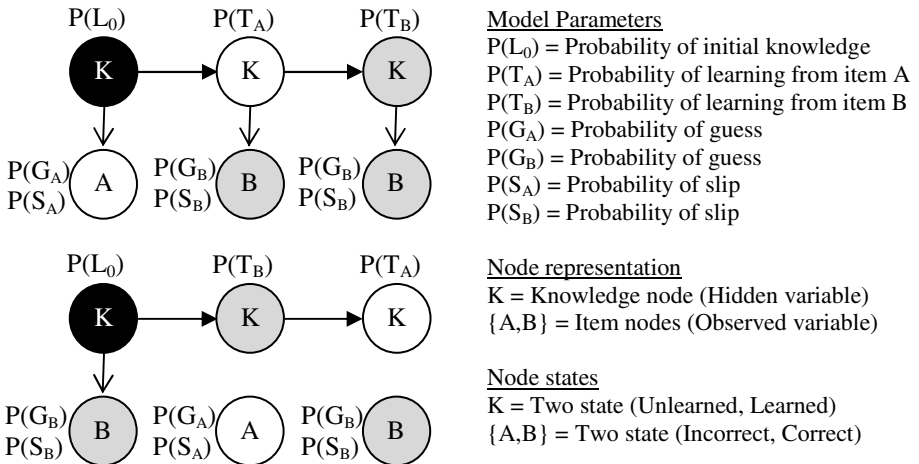


Fig. 2. An example two sequence topology of the Item Effect Model with two item types and descriptions of the model’s knowledge and performance parameters

The Item Effect Model, depicted in Figure 2, allows for a learning rate as well as a guess and slip rate per item, to be learned. The guess rate is the probability that a student will answer an item correctly even if they do not know the skill involved. The slip rate is the probability that a student will answer an item incorrectly even if they know the skill involved.

The Item Effect Model looks for when a student is believed to have transitioned from the unlearned state to the learned state, which is generally indicated by incorrect answers followed by correct answers to a series of items. The model credits the last item answered incorrectly as most probably causing this learning. If the model observes a pattern of learning that frequently occurs after a particular item, that item is attributed with a higher learn rate than the other items in the problem set being considered. The probabilities of learning associated with the items are relative to the other items in the problem set and are indicative of a particular item's ability to cause positive performance on the other items.

Because of the fashion in which the Item Effect Model looks for patterns of learning, it requires that the items in a problem set be given to students in a random order. The Item Effect Model in fact models every permutation of a set of items. This means that when analyzing a 3 item problem set, all 6 permutations of sequences are modeled. Randomization is required much in the same way that an RCT requires randomization in the assignment of conditions. Neither hypothesis testing nor the Item Effect Model can identify learning effects from a single linear sequence. In order to determine the statistical reliability of differences in learn rates, the data is randomly split into 10 bins of equal size. The method is run on each of the 10 bins and the probability of one item having a higher learning rate than another in N or more bins is determined by the binomial function: $1 - \text{binocdf}(N-1, 10, 0.50)$. In this paper an $N \Rightarrow 8$ will be considered statistically significant.

3 Analyzing Experiments in Mastery Learning Content

Mastery learning [2] is a method of instruction where by students are only able to progress to the next topic after they have demonstrated mastery of the current topic. In the cognitive tutor [4], mastery is achieved when their knowledge tracing model believes the student knows the topic with 0.95 or better probability. Items are often selected in a random order and students spend a lot of time in this type of instruction making it especially lucrative to mine. If a student gets an item wrong or requests help she is given tutorial feedback aimed at improving understanding of the current topic. This is a currently untapped opportunity for investigators to be testing different types of tutorial feedback. By making multiple types of tutorial feedback that can be chosen at random by the system when a student begins an item, investigators can plan hypotheses to test with experiments and embed those experiments within mastery problem sets. After data is gathered, the student response sequences can be analyzed and the learning rates of each strategy calculated using the Item Effect Model. A statistical significance test is employed with the method to tell investigators the probability that their result occurred by chance. We ran 5 such tutorial feedback experiments embedded in mastery learning content. The following sections will show how this data was analyzed with the model and the conclusions made.

3.1 The Tutorial Feedback Experiments

We planned out five experiments to investigate the effectiveness of various types of tutorial feedback shown in Table 1. The choices of feedback types were selected based on past studies of effective tutor feedback and interventions [5, 6, 7] that have been run on The ASSISTment System. To create the experiments we took existing mastery learning problem sets from various math subjects and created two types of feedback conditions for each item in the problem set. The two types of feedback corresponded to the conditions we had planned for that experiment. This authoring process was made less tedious by utilizing the template feature described in section 1.2 to create the two types of tutorial help templates for each item template in the problem sets.

Table 1. The five planned mastery tutoring experiments and a description of their subject matter and the two types of tutor feedback being tested

Experiment #	Condition A	Condition B	Subject Matter
1	Solution (steps)	TPS	Ordering fractions
2	Solution	Worked Example	Finding percents
3	Hints	TPS	Equation solving (Easy)
4	Solution (steps)	Solution	Equation solving (Medium)
5	Solution (steps)	TPS	Equation solving (Hard)

There were five types of tutorial feedback tested. This is a description of each:

- TPS: Tutored Problem Solving [5, 6, 7]. This is a scaffolding type of feedback where students are asked to solve a number of short problems that help the student solve the original harder problem.
- Worked Example: In this condition [7] students are shown a complete solution to a problem similar to the one they were originally asked to solve.
- Solution: In this condition [5] students are shown a complete solution to the exact question they were originally asked to solve.
- Solution (steps): In this condition [5] students were shown a complete solution to the problem they were originally asked to solve but broken up in to steps. The student needed to click a check box confirming she or he had read the current solution step to move on.
- Hints: In this condition [6] students were given text based hints on how to solve the problem. Students had the opportunity to attempt to answer the original question again at any time. If the student asked for additional hints, the hints would start informing the student exactly how to solve the problem. The last hint would tell them the correct answer to the original problem.

3.2 Modeling Mastery Learning Content Containing Multiple Types of Tutoring

We adapted the Item Effect Model to suit our needs by making small changes to the assumption of what an item represents. In the standard Item Effect Model, an item

directly represents the question and tutorial feedback associated with that item. Since we were concerned with the effectiveness of multiple types of tutorial feedback for the same items, we let the tutorial strategy that was selected for the student to see be the representation of an item.

For example, suppose we had a mastery learning problem set that used two templates, 1 and 2, and we also have two types of tutor feedback, A and B, that were created for both templates. We might observe student responses like the ones in Table 2 shown below.

Table 2. Example of mastery learning data from two students

Student 1 response sequence	0	0	1
Student 1 item sequence	1.A	2.B	1.B
Student 2 response sequence	0	1	1
Student 2 item sequence	1.B	1.A	1.B

Table 2 shows two students' response patterns and the corresponding item and tutorial feedback assigned (A or B). Student 1 answers the first two items incorrectly (0) but the last one correctly (1). Student 2 answers the first item incorrectly but the remaining two correctly. If we assume both students learned when they started to answer items correctly then we can look at which tutorial strategy directly preceded the correct answers and credit that tutorial strategy with the learning. In the example data above, tutorial feedback B precedes both students' learning. In essence, this is what our Bayesian method does to determine learning rates of types of tutorial feedback, albeit in a more elegant fashion than the observational method just described. For analyzing the learning effect of different types of tutor feedback we assume all items in a problem set to have the same guess and slip rate since the guess and slip of an item is independent of its tutorial feedback. For simplicity and to keep the amount of data the Bayes method used close to that of the traditional method, only students' first three responses in the problem sets were used.

3.3 Traditional Hypothesis Test Used

In order to compare the Bayesian analysis to a more traditional learning gain approach, we came up with the following method to determine which tutorial feedback produced more learning. First we selected only those students who answered their first question incorrectly because these are the only students that would have seen the tutoring on the first question. These students were then split into two groups determined by the type of feedback (condition A or B) they received on that question. We let the second question pose as a post-test and took the difference between their first response and their second response to be their learning gain. Since only incorrect responses to the first question were selected, the gain will either be 0 or 1. To determine if the difference in learning gains was significant we ran a t-test on student learning gains of each condition. We do not claim this to be the most powerful

statistical analysis that can be achieved but we do believe that it is sound, in that a claim of statistical significance using this method can be believed.

3.4 Results

The Item Effect Model inferred learning rates for each condition per experiment. The condition (A or B) with the higher learning rate for each experiment was chosen as Best. For the traditional analysis the condition with the higher average gain score was chosen as Best. Table 3 shows the results of the analysis including the number of students in each experiment and if the two methods agreed on their choice of best condition for the experiment.

Table 3. Analysis of the five tutoring experiments by the traditional hypothesis testing method and the Item Effect Model. The methods agreed on the best condition in 5 of the 6 experiments.

#	Users	<i>Traditional analysis</i>			<i>Item Effect Model analysis</i>			
		Best	Significance	Agree?	Best	Significance	Guess	Slip
1	155	B	0.48	Yes	B	0.17	0.28	0.07
2	302	A	0.52	Yes	A	0.17	0.11	0.14
3	458	B	0.44	No	A	0.62	0.18	0.09
4	278	A	0.57	Yes	A	0.38	0.15	0.17
5	141	B	0.63	Yes	B	0.62	0.15	0.13
5*	138	B	0.69	Yes	B	0.05	0.14	0.17

The traditional analysis did not find a significant difference in the conditions of any of the experiments while the Item Effect Model found a significant difference in one, 5*. The reliably better condition for this experiment was B, the tutored problem solving condition. This was also the experiment where students who saw condition A were given a bad solution due to a typo. These results show that the Item Effect Model successfully detected this effect and found it to be statistically significantly reliable while the traditional method did not. Experiment 5 represents the data from students after the typo was fixed. Condition B is still best but no longer significantly so.

Also included in the table are the guess and slip rates learned by the Item Effect Model for each experiment. It is noteworthy that the highest guess rate learned of 0.28 was for the only experiment whose items were multiple-choice (four answer choices). It is also noteworthy to observe that the guess values for the easy equation solving experiment (#3) has the highest probability of guess of the three equations solving experiments (3-5). These observations are evidence of a model that learns highly interpretable parameter values; an elusive but essential trait when the interpretation of parameters is informing pedagogical insights.

4 Analyzing RCT Designs with Feedback on All Items

We wanted to investigate the performance of our model on data that took the form of a more traditional RCT. Instead of running five new randomized controlled trials, we

searched our log data for regular problem sets that were not intended to be RCTs but that satisfied a pre-test/condition/post-test experimental design. For this analysis, the importance was not which particular tutor feedback was being tested but rather the model's ability to find significance compared to hypothesis testing methods that were designed to analyze data similar to this.

4.1 Looking for RCT Designs in the Log Data of Randomized Problem Sets

The data used in this analysis also came from The ASSISTment System. We identified four item problem sets in which the items were given in a random order and where each of the four items tested the same skill. Once we identified such problem sets we selected pairs of sequences within each problem set where the first and third items presented to the students were the same. For example, for items A,B,C,D we looked at the specific sequence pairs of orderings CADB and CBDA where item B would serve as the pretest, item D is the posttest, and items A and C would serve as the two conditions. We required that students had completed the four items in a single day and that there were at least 50 students of data for each of the sequence pairs.

4.2 The Inter-rater Plausibility Study

We wanted to provide a human point of reference with which to compare results against. To do this we selected four educators in the field of mathematics as raters. They were told which two items were used as the pre and post-test and which were used as the conditions. They were also able to inspect the feedback of the items and then judge which of the two conditions were more likely to show learning gains.

4.3 Modeling Learning in Problem Sets with RCT Data Sequences

One difference between the modeling of these datasets and the mastery learning datasets is the reduction in sequences and the decision to let each item have its own guess and slip value. Observing only two sequence permutations is not the ideal circumstance for the Item Effect Model but represents a very common design structure of experiment data that will serve as a relevant benchmark.

4.4 Statistical Test

Since this particular data more closely resembled an RCT, we were able to use a more familiar learning gain analysis as the traditional hypothesis testing method of comparison. Learning gain was calculated by taking the post-test minus the pre-test for each student in their respective condition. To calculate if the learning gains of the two conditions were statistically significantly different, a t-test was used.

4.5 Results

Table 4 shows the results of the two analysis methods as well as the best condition picks of the four raters in the subject matter expert survey. For each experiment the condition groups were found to be balanced at pre-test. There was one experiment, #4, in which both methods agreed on the best condition and reported a statistically significant difference between conditions.

Table 4. Analysis of the five RCT style experiments by the traditional hypothesis testing method and the Item Effect Model. The methods agreed on the best condition in 4 of the 5 experiments and agreed on statistical significance in one experiment.

#	Users	Traditional analysis			Agree?	Item Effect Model		Rater Picks			
		Best	Significance			Best	Significance	1	2	3	4
1	149	A	0.74		Yes	A	0.95	A	A	A	A
2	197	A	0.36		Yes	A	0.62	A	A	A	A
3	312	A	0.04		No	B	0.38	A	A	A	B
4	208	A	0.00		Yes	A	0.05	A	A	A	A
5	247	B	0.44		Yes	B	0.82	A	A	A	A

The subject matter experts all agreed on four of the five experiments. On the experiment where one rater disagreed with the others, #3, the majority of raters selected the condition which the traditional method chose as best. This experiment was also one in which the two methods disagreed on the best condition but only the traditional method showed statistical significance. On the experiment in which both methods of analysis showed a statistical difference, #4, there was total agreement among our subject matter experts and both of the methods. On average the traditional method agreed more with the raters choices and also found significance in two of the experiments where as the Item Effect Method only found significance in one. However, a correlation coefficient of 0.935 was calculated between the two methods' significance values indicating that The Item Effect method's significance is highly correlated with that of the hypothesis testing method for these RCT style datasets.

5 Contributions

In this work we presented an empirical validation of some of the capabilities and benefits of the Item Effect Model introduced by Pardos and Heffernan (2009). We conducted five original experiments comparing established forms of tutorial feedback with which to compare the analysis of a traditional hypothesis testing approach to the Item Effect Model. In conducting the experiments by inserting multiple tutorial feedback conditions into items in mastery learning problem sets, we were able to demonstrate an example of running an experiment without interrupting the learner's curriculum and without giving them at times lengthy pre-tests that prohibit feedback which places students' learning on hold.

A second contribution of this work was to highlight how random orderings of questions could be analyzed as *if they were* RCTs. This is a powerful idea, that simply randomizing the order of items creates intrinsic experimental designs that allow us to mine valuable pedagogical insights about what causes learning.

We believe that the methodology we used to calculate significance in the Item Effect Model could be made more powerful but decided to air on the side of caution by using a method that we were sure was not going to lead us to draw spurious conclusions. It was encouraging to see how the Item Effect Model fared when compared to a more traditional hypothesis testing method of analysis. The model agreed with the

traditional tests in 9 out of the 11 experiments and detected an effect that the traditional method could not; a typo effect in a condition of one of our mastery learning problem set experiments. We believe that the implication of these results is that ITS researchers can safely explore their datasets using the Item Effect Model without concern that they will draw spurious conclusions. They can be assured that if there is a difference in learning effect there is a reasonable chance it can be inferred and inferred from most any source of randomized item data in their system.

Acknowledgements

We would like to thank all of the people associated with creating The ASSISTment System listed at www.ASSISTment.org. We would also like to acknowledge funding from the US Department of Education, the National Science Foundation, the Office of Naval Research and the Spencer Foundation. All of the opinions expressed in this paper are those of the authors and do not necessarily reflect the views of our funders.

References

1. Pardos, Z.A., Heffernan, N.T.: Detecting the Learning Value of Items in a Randomized Problem Set. In: Dimitrova, Mizoguchi, du Boulay, Graesser (eds.) Proceedings of the 13th International Conference on Artificial Intelligence in Education, pp. 499–506. IOS Press, Amsterdam (2009)
2. Corbett, A.T.: Cognitive computer tutors: solving the two-sigma problem. In: Bauer, M., Gmytrasiewicz, P.J., Vassileva, J. (eds.) UM 2001. LNCS (LNAI), vol. 2109, pp. 137–147. Springer, Heidelberg (2001)
3. Corbett, A.T., Anderson, J.R.: Knowledge tracing: modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction* 4(4), 253–278 (1995)
4. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education* 8, 30–43 (1997)
5. Razzaq, L., Heffernan, N.T.: To Tutor or Not to Tutor: That is the Question. In: Dimitrova, Mizoguchi, du Boulay, Graesser (eds.) Proceedings of the 13th International Conference on Artificial Intelligence in Education, pp. 457–464. IOS Press, Amsterdam (2009)
6. Razzaq, L., Heffernan, N.T.: Scaffolding vs. hints in the Assistment system. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 635–644. Springer, Heidelberg (2006)
7. Kim, R., Weitz, R., Heffernan, N., Krach, N.: Tutored Problem Solving vs. “Pure”: Worked Examples. In: Taatgen, N.A., van Rijn, H. (eds.) Proceedings of the 31st Annual Conference of the Cognitive Science Society, Cognitive Science Society, Austin (2009)

Persuasive Dialogues in an Intelligent Tutoring System for Medical Diagnosis

Amin Rahati and Froduald Kabanza

Department of Computer Science,
University of Sherbrooke
Sherbrooke, Quebec J1K 2R1
Canada
{amin.rahati,kabanza}@usherbrooke.ca

Abstract. Being able to argue with a student to convince her or him about the rationale of tutoring hints is an important component of pedagogy. In this paper we present an argumentation framework for implementing persuasive tutoring dialogues. The entire interaction between the student and the tutoring system is seen as an argumentation. The tutoring system and the student can settle conflicts arising during their argumentation by accepting, challenging, or questioning each other's arguments or withdrawing their own arguments. Pedagogic strategies guide the tutoring system selecting arguments aimed at convincing the student. We illustrate this framework with a tutoring system for medical diagnosis using a normative expert model.

Keywords: intelligent tutoring system, argumentation, persuasive dialogue, medical diagnosis.

1 Introduction

One of the key problems in the development of an intelligent tutoring system (ITS) concerns the implementation of the verbal exchange (i.e., a dialogue) that takes place between a student and the ITS. A dialogue determines what the ITS tells the student, when and how, to support her learning process in the most effective way. Some approaches implement ITS dialogues using finite state machines (FSM) [1,2]. Other approaches use dynamically generated dialogue structures, by using automated planning techniques [3,4]. In these approaches, the requirement for the ITS to persuade the student is not formally acknowledged. Such a requirement can be implemented by adopting a formal framework of argumentation for the implementation of dialogues between the ITS and the students. A number of such frameworks has been developed, including applications to decision support systems [5,6,7,8,9].

Some ITS involve argumentation as the content of the learning material, for instance to learn skills of argument reasoning by analyzing arguments. In particular, LARGO is a system used to train students on acquiring argumentation skills [10]. LARGO has a graphic interface through which students can represent or visualize arguments they make and their relations. It can also provide

feedback to students on their construction of arguments. However, in those ITS, which teach argumentation, argumentation is not involved as a pedagogical tool aiming at persuading the student on the rationale of the interventions made by the ITS to support her in her learning process.

In this paper we present a general approach for implementing persuasive tutoring dialogues. In our approach, every action performed by a student trying to solve a problem is considered as an argument. The ITS intervenes to help the student also by making arguments. Errors made by the student are considered as a disagreement and the ITS tries to help the student remedy them through an argumentation.

The framework is composed of three key components. The first component is a language for defining dialogue moves between the ITS and the student. A typical dialogue move specifies the content of an argument or a propositional attitude in the exchange of arguments (e.g., accepting the interlocutor's argument or withdrawing own argument). The second component is a protocol regulating the moves and conveyed constraints on allowed move sequences in an argumentation dialogue. The third component is an argument generator used by the ITS to decide arguments to use which are persuasive for the student. We use Walton's argumentation theory ([8]) to model arguments, challenges to arguments and acceptance of arguments. We integrate this theory with the notion of preference among arguments [11], making it possible for the ITS to make decisions on the most convincing arguments. Higher level strategic rules ([12]) are also involved to select arguments based on a pedagogic goal.

The remainder of this paper is organized as follows. In the next section we start by introducing TeachMed [1], a medical diagnosis ITS, which we use as a testbed to illustrate our argumentation framework. This is followed by description of the argumentation framework. We then give an illustration using TeachMed testbed. We conclude with a discussion on related and future work.

2 Argumentative TeachMed

In TeachMed, a student starts by selecting a virtual patient having a particular disease with the objective to generate a correct diagnosis. The student makes a diagnosis by performing an investigation. To formulate some initial hypotheses she starts asking queries to the virtual patient about the different symptoms, life style and family background. She can also make queries in terms of a physical exam on a 3D model of the patient (e.g., reflexes) or in terms of lab tests (e.g., blood samples). Queries and tests are selected from a list including noise queries. Each query has an answer specified in the virtual-patient model, which includes his vital signs, symptoms and results of lab tests or physical exam. As more queries are asked, she will eliminate some hypotheses, strengthen others and generate new ones. This process continues until she can narrow the list of hypothesis down to one or two –that is, the final diagnosis. The challenges in solving this type of problems involve deciding what evidences to observe and how observe it, and determining the list of hypotheses that best explain the observed

evidences. The queries and the differential diagnosis generation are the student's diagnosis (problem solving) actions. We also have student's utterance actions, in the form of requests for help or replies to utterances from TeachMed.

To implement our approach, we modified TeachMed architecture [1], by extending the *user model* to store the student's order of preferences concerning the decision making parameters for making diagnostic actions and utterances. The original model only recorded the student's diagnosis actions. The student's actions are now recorded as arguments. We preserve the use of an influence diagram (ID) to model the *expert knowledge* - the ID represents the causal relationship between symptoms and diseases and the utility of queries. The *pedagogic model* is now replaced by the argumentation framework.

3 The Argumentation Framework

The entire session of a student learning to diagnose a case is considered as an argumentation between the student and the ITS. Whenever a student performs a diagnosis action, TeachMed interprets the action as an assertion which the ITS tries to reject if it can. An assertion will be rejected if the pedagogic model finds a convincing argument against it. This is done based on the information provided by expert model, the commitment store and the dialogue history. For instance, if a student asks a query which is irrelevant to the current differential diagnosis –for instance the value of information for the query is low, according to the ID expert model– the pedagogic model calculates a convincing argument. Then TeachMed tries to reject the query by initiating an argumentation phase during which TeachMed. During this phase, the student's actions will be utterances constituting replies to TeachMed's utterances. These utterances may be arguments, counter arguments, withdrawal of arguments, and acceptance of arguments. The student may also proactively requests help. This too triggers an argumentation phase during which the dialogue moves are utterances.

At any point during the interaction, each arguer is committed to some arguments. For the student, these include the set of gathered evidence and the set of hypotheses. For TeachMed, they include the set of hypotheses explained by the ID expert model from the evidence gathered by the student. Commitments also include arguments asserted during verbal exchanges. They are updated depending on the performed actions. A structure called the "Commitment Store" keeps track of the current commitments. It is a list of pairs (*argument, arguer*).

To have a formal argumentation framework modeling the interactions between the student and the ITS, we need a language for modeling the content of arguments and the exchange –or communication– of arguments.

3.1 Domain Definition and Communication Language

Following [13], we define the language at two levels, namely the *domain level* and the *communication level*. At the domain level, the language provides a syntax and semantics for arguments. At the communication level, the language defines

primitives for move types - propositional attitudes- that are available for exchanging arguments.

An argument is a premise and a conclusion, where the premise is a conjunction of propositions and the conclusion is a proposition . A move type is a template operator described by a precondition (conjunction of predicates) specifying when the move is feasible –it specifies the conditions that the commitment store and/or the dialogue history must satisfy for the move to be applicable– and an effect specifying the update of the commitment store and the dialogue history. Figure 1(a) illustrates examples of move types: OPEN, CLOSE, ASSERT, ACCEPT, WITHDRAW, REJECT, CHALLENGE, and QUESTION.

An arguer commits to an argument by asserting the argument or accepting it [14]. Arguers are not limited to committing to only what they believe or to believe what they commit to. Adopting the concept of a commitment store helps us avoiding the complexity and inefficiency regarding the use of a belief update framework in dialogue modeling [15]. As argued by [16], the commitment store concept also provides a means to settle conflicts between arguers by making the opponent commit to the proponent’s assertion or the proponent withdraws his assertion. The dialogue-history keeps track of the history of moves made by the arguers. This is a path in dialogue tree. Figure 1(b) shows the details of the move ACCEPT.

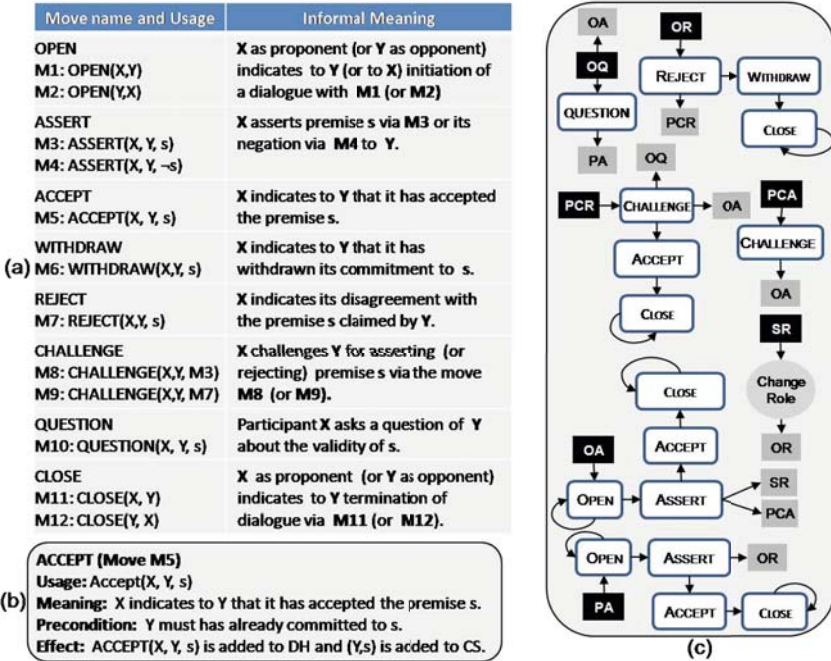


Fig. 1. (a)Communication Moves; (b)Details of ACCEPT;(c) The protocol

3.2 The Protocol

Each time the student makes an assertion, TeachMed checks whether it agrees with the assertion and whether it should reject it. The situation in which TeachMed intervenes is determined by the pedagogic strategic rules, which we discuss later. For the time being, let's assume the rule is to intervene on every error. For example, whenever the student generates a hypothesis not probabilistically related to current evidences given the ID expert model, TeachMed rejects the assertion by making an argument against it. The student may counter with her own argument against TeachMed's argument. And so on, the argumentation can continue until settling the initial disagreement.

Thus the settling of a disagreement could recursively spawn an argumentation dialogue within a current one. Accordingly we define the argumentation protocol using hierarchical state diagrams similar to statecharts [17,18]. In the example of Figure 1(c), we have different diagrams, some representing superstates in other diagrams: *proponent assert*(PA), *opponent assert*(OA), *proponent challenge reject*(PCR), *proponent challenge assert*(PCA), *swap roles* (SR), *opponent rejection* (OR) and *opponent question*(OQ). The entry point of each diagram is shown by a black box. A superstate corresponding to a diagram is shown as gray box. A normal state corresponds to a dialogue move. A circle indicates a change in the roles of the arguers, switching from a proponent role (making an assertion) to an opponent one (challenging an assertion), or vice-versa.

3.3 Computing Convincing Arguments

Given an assertion made by the student, TeachMed must decide whether to reject or accept the student assertion. Here we follow Walton's argumentation theory [8] by specifying rules expressing how to respond to arguments made by the opposing party in a two-participant argumentation. Precisely, we want to model the rules for generating counterarguments by TeachMed to convince the student. These argument generation rules, called test questions by Walton, specify arguments that can challenge assertions made by the student - diagnostic actions as well as utterance actions during a conflict settling dialogue.

An argument generation rule (AGR) is a template rule for generating a counterargument to a given assertion, consisting of:

- **Parameters:** Variables used in the template.
- **Argument:** The challenged argument.
- **Context:** A conjunction of predicates over the commitment store and problem solving state
- **Premise:** Premise of the counterargument (conjunction of predicates).
- **Conclusion:** Conclusion of the counterargument (predicate).
- **Value:** Preference value of the counterargument

The variables in the predicates must be defined in the parameters. We associate preferences to the assertions made by the student and associate strengths to arguments generated by the AGRs. The preference value indicates the value of a

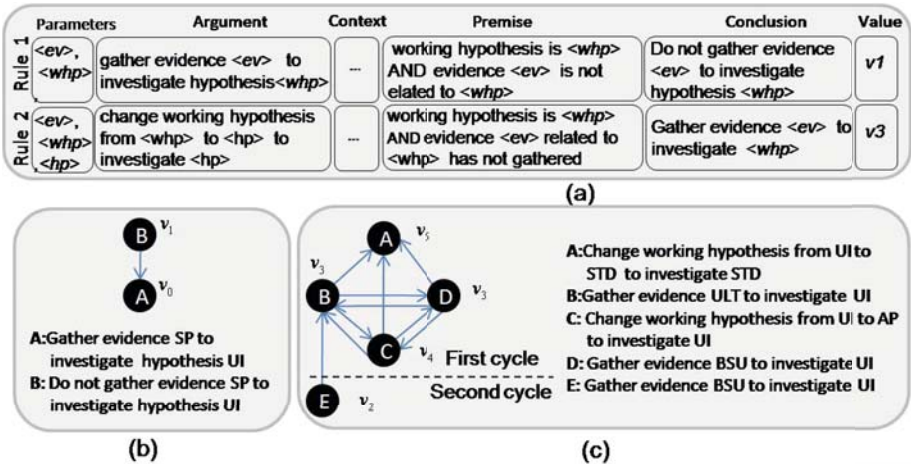


Fig. 2. Rules of test questions (a) and argument generation rules (b, c)

specific decision making parameter in problem solving that is promoted by the counterargument.

Given an assertion and a set of AGRs , we find a set of potential counterarguments by matching the assertion with the assertion component of the AGRs and the premise component with the context (problem solving state, commitment store). If a match is obtained, the resulting instantiation is used uniformly to replace the variables in the conclusion.

A convincing argument exists among this set of generated arguments if it defeats the student’s argument but it is not defeated by any counterargument. To calculate such argument we adopted a decision-theoretic argumentation method from [11] – which is an extension of Dung’s argumentation [19] .

Figure 2(a) shows examples of AGRs. Rule 1 is applicable to counter evidence gathered in order to justify a given a hypothesis. A pedagogic strategy for beginners is to forbid them zigzagging between hypotheses (e.g., asking questions to the patient related to one hypothesis, switching to another then coming back to the previous, and so on). To enforce such a strategy, the teacher may require the student to stick to one hypothesis as far as possible by exhausting the related evidence [20]. Rule 2 is meant to enforce such a pedagogic strategy by countering inappropriate changes of the current working hypothesis. This rule is matched when the student asks a query related to a new hypothesis while the ID suggests there are still evidences relevant to the current working hypothesis.

3.4 Move Selection

Given a convincing argument against the student’s assertion, a dialogue strategy is needed to conveying the different parts of the arguments (premises and conclusion) in form of an argumentation dialogue. Such strategies are specified using an argumentation protocol, specified as a hierarchical transition diagram.

In practice, we specify a dialogue strategy by following a three-level methodology, inspired from [12]:

1. maintain or alter focus of discussion
2. building own point of view or destroying the user's view
3. adopting method to achieve the objective set at level 1 and 2.

Level 1 is appropriate where concepts like relevance are important [21]. Following [22], at level 1, information of the student's profile regarding preferences among decision making parameters is taken into account. At level 2, build and destroy strategies are encoded in the protocol. Building means making the user to accept the proposition which shows TeachMed's point of view. Destroying means making the user withdraw of a proposition which indicates her point of view. The two first levels include some pedagogic goals. The third level refers to some domain dependent tactics which achieve those goals.

For example, the following rules are used at the third level to decide a move for TeachMed in reaction to the student's move:

- **PA** rules: If the student as proponent asserts an argument and TeachMed does not find a convincing argument against it then the argument is accepted and the dialogue ends (PA-1). Otherwise, TeachMed rejects the assertion through a transition to the OR diagram (PA-2).
- **OR** rules: if the student rejects an argument then TeachMed challenges the student through a transition to the PCR diagram to find the cause of conflict.
- **PCR** rules: if the student challenges the rejection of her argument by TeachMed, then the TeachMed tries to resolve the conflict by using a destroy strategy or a build strategy. If the destroy strategy has chosen, and a premise of the convincing argument exists to which the student has not committed so far, TeachMed begins asks a question on that premise through a transition to the OQ diagram (PCR-1). Otherwise, if such premise does not exist, or the build strategy is chosen, then TeachMed asserts the convincing argument through a transition to the OA diagram (PCR-2).
- **OA** rules: if the student as opponent has asserted an argument then TeachMed has to accept the assertion if it does not have a convincing counterargument (OA-1). Otherwise, TeachMed indicates its conflict with the assertion by rejecting the asserted argument through the choice of a transition to the SR diagram, which in turn leads to the OR diagram (OA-2).

We also have rules controlling backtracking after a child dialogue terminates. For instance, if the terminated child dialogue was created by the OA diagram invoked by the OQ diagram, and the proponent has committed to the assertion made by the opponent, then this means that the opponent has succeeded in applying a destroy strategy to justify the rejection. Therefore backtracking is necessary to REJECT state of the OR diagram of the parent. This gives the proponent the opportunity to make another choice as response to the opponent's rejection.

4 Example

Let's define the set of preferences $V = \{v_1, v_2, v_3, v_4, v_5\}$ with partial order $v_1 > v_2 > v_3 > v_4 > v_5$. Let's then associate these preferences to pedagogic goals as follows:

- v_1 : states evidence gathering actions must be consistent with medical knowledge and the available patient information;
- v_2 : evidence gathering actions should be minimized (e.g., take into account the monetary costs of lab tests; delays; and intrusive physical exams). TeachMed evaluates this by taking into account the expect value of information for queries.
- v_3 : the current working hypothesis should remain the focus until exhausting related evidence;
- v_4 : most life-threatening hypothesis should be investigated first.
- v_5 : most likely hypothesis should be selected first.

Initially, TeachMed presents a patient to the student with short description of the patient complain (e.g., "patient complaining of abdominal pain") together with the vital signs. The student can initialize a differential diagnosis right away based on the problem statement. In the scenario illustrated herein, a student was presented with a pelvic inflammation case, the patient complaining of abdominal pain. The ID is the same as in [1] and covers abdominal pains.

Figure 3 depicts an excerpt of the scenario with a trace of the internal inferences behind the argumentation process. Until Step 4, the student was querying the patient. The dialogue states (third column) and participant's role (fourth column) are those in Figure 1. The moves (fifth column) are those discussed in Section 3.1. The rules (last column) are those in Section 3.4.

The student began with the hypotheses of *Urinary Infection*(UI), *Sexual Transmitted Disease*(DST) and *Appendicitis*(AP). UI is the most probable hypothesis and AP is the less probable one but is threatening to the patient's life. Thereafter, the student has started investigation of the UI (current working hypothesis).

According to Step 1, the student's action (proponent's assertion M3), initiated the persuasive dialogue D1. TeachMed and the student synchronize the start of new dialogue by the move OPEN and end it by the move CLOSE. Based on the PA diagram, TeachMed has to choose among two permitted moves: M5 and M7. As it does not find any argument against the student move, the rule PA-1 is matched, and TeachMed agrees with the student's assertion by making the move M5. Note that, to provide a natural problem solving interaction for the student, TeachMed remains silent when it accepts the student's diagnostic action.

Until Step 4, TeachMed has not found any conflict with the student's actions (assertions). At Step 4, TeachMed notices inconsistencies between the evidences and the hypotheses formulated by the student. This matches the counter-argument B in Figure 2(b), which spawns further argumentation with the student to settle the disagreement. This time, the rule PA-2 matches, so that among two permitted moves (M5 and M7) TeachMed chooses M7, meaning it

Collected Evidence: Acute Lower abdominal pain				
Working Hypothesis: Urinary Infection (UI)				
Player	Utterance	Dialog#	Role	Moves Rules
1. <u>Student</u> :	Any fever?	D1(Open)	P	M1, M2, M3
<u>Patient</u> :	I don't know.			
TM Pedagogue:	Silence	D1(Close)	O	M5, M11, M12 PA-1
2. <u>Student</u> :	Is it worse on one side?	D2(Open)	P	M1, M2, M3
<u>Patient</u> :	No			
TM Pedagogue:	Silence	D2(Close)	O	M5, M11, M12 PA-1
3. <u>Student</u> :	Do you urinate more since beginning of your pain?	D3(Open)	P	M1, M2, M3
<u>Patient</u> :	I don't know. More often I think.			
TM Pedagogue:	Silence	D3(Close)	O	M5, M11, M12 PA-1
4. <u>Student</u> :	Do you have a sexual partner(SP)?	D4(Open)	P	M1, M2, M3
5. <u>TM Pedagogue</u> :	You cannot ask such a question.		O	M7 PA-2
6. <u>Student</u> :	Why?			M9
7. <u>TM Pedagogue</u> :	It is not related to the working hypothesis UI.	D5(Open)	P	M2, M1, M4 PCR-2
8. <u>Student</u> :	But I can.		O	M7
9. <u>TM Pedagogue</u> :	Why?		P	M9 OR
10. <u>Student</u> :	Because I investigate Sexual Transmitted Disease (STD).	D6(Open)	P	M1, M2, M3
11. <u>TM Pedagogue</u> :	You cannot investigate STD		O	M7 OA-2
12. <u>Student</u> :	Why?		P	M9
13. <u>TM Pedagogue</u> :	Did you finish working with UI?		O	M10 PCR-1
14. <u>Student</u> :	Yes.	D7(Open)	P	M1, M2, M4
15. <u>TM Pedagogue</u> :	But working on UI has not finished yet.		O	M7 PA-2
16. <u>Student</u> :	Why?		P	M9
17. <u>TM Pedagogue</u> :	Evidence Burning Sensation of Urinating (BSU) has remained.	D8(Open)	P	M2, M1, M3 PCR-2
18. <u>Student</u> :	You are right.	D8(Close)	O	M5, M12, M11
19. <u>Student</u> :	I withdraw my claim that working with UI has finished.	D7(Close)	P	M6, M12, M11 BT
20. <u>TM Pedagogue</u> :	So you should continue working with UI.	D9(Open)	P	M2, M1, M3 BT, PCR-2
21. <u>Student</u> :	Ok.	D9(Close)	O	M5, M12, M11
22. <u>Student</u> :	I withdraw my claim for changing working hypothesis UI to STD.	D6(Close)	P	M6, M12, M11 BT
23. <u>Student</u> :	I accept that my question about SP is not related to the UI.	D5(Close)	O	M5, M11, M12 BT
24. <u>Student</u> :	I withdraw my claim for gathering evidence SP.	D4(Close)	P	M6, M12, M11 BT

Fig. 3. Medical Diagnostic Scenario. Shorthand notations: P (Proponent), O(Opponent), BT (Backtracking).

rejects the student’s action (Step 5). In response, according to choices offered by the OR diagram, the student has to choose among two moves M6 and M9. Permitted moves are provided to the students through a menu selection.

All through steps 5 to 24 , the matched counter arguments and move selection rules drove TeachMed towards persuading the student of her mistake and remedy the situation. More specifically, from step 5 to 10 TeachMed uses only the persuasive argument *B* of figure 2.b but after Step 10 it uses also the persuasive argument *E* of figure 2.c since any assertion made by the student or TeachMed starts a new persuasive dialogue. In this example, the student ends up accepting that she had committed a mistake and switched back to the urinary infection.

5 Conclusion and Future work

In this paper we described an argumentation framework that can be integrated to any ITS for conducting a persuasive dialogue with the student. Our framework is still in a prototyping phase and still has some limitations. The above scenario works as indicated in the current implementation. Arguable, the dialogue with the student is still not yet realistic, mainly because the argument rule base still needs significant fine tunings. In particular, the utterances made by the

students are actually text templates on move choices offered to him at the current step of the interaction. Refinement of the dialogue transitions and the utterance templates will contribute to making the dialogues more realistic.

The fact that an assertion made by a student can be challenged, from a pedagogic point of view, it does not mean that ITS should indeed challenge it. It can be very frustrating for a student to see ITS intervene on every error. Rather, depending on pedagogic goals and constraints set by a teacher, the ITS should intervene when a given number of errors with some level of severity have accumulated. This provides another area of improvement.

Although still quite preliminary, the current experiment demonstrates the potential of our approach in fostering learning by the student, by making her reveal her understanding of current problem solving step, and leading her to actively search in her knowledge to generate a convincing argument, reflect upon it, and remedy to a situation. Besides improving the current implementation as just mentioned, a crucial component of future works concerns an evaluation of the argumentation capability to determine to what extent it indeed facilitates diagnostic skill acquisition compared to some other types of training intervention.

Acknowledgement

We wish to thank the reviewers for insightful and constructive comments which helped us improving the writing of this paper.

References

1. Kabanza, F., Bisson, G., Charneau, A., Jang, T.S.: Implementing tutoring strategies into a patient simulator for clinical reasoning learning. *Journal of Artificial Intelligence In Medicine (AIIM)* 38, 79–96 (2006)
2. Zhou, Y., Freedman, R., Glass, M., Michael, J.A., Rovick, A.A., Evens, M.W.: Delivering hints in a dialogue-based intelligent tutoring system. In: *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (1999)
3. Freedman, R.: Plan-based dialogue management in a physics tutor. In: *Proceedings of the 6th Applied Natural Language Processing Conference*, Seattle (2000)
4. Zinn, C., Moore, J.D., Core, M.G.: A 3-tier planning architecture for managing tutorial dialogue. In: Cerri, S.A., Gouardères, G., Paraguaçu, F. (eds.) *ITS 2002*. LNCS, vol. 2363, pp. 574–584. Springer, Heidelberg (2002)
5. Nguyen, H.: Designing persuasive health behaviour change dialogs. In: *Proceedings of the 11th international conference on User Modeling*, pp. 475–479 (2007)
6. Andrews, P., De Boni, M., Manandhar, S.: Persuasive argumentation in human computer dialogue. In: *Proceedings of the AAAI 2006 Spring Symposia* (2006)
7. Guerini, M., Stock, O., Zancanaro, M.: Persuasion models for intelligent interfaces. In: *Proceedings of the IJCAI Workshop on Computational Models of Natural Argument* (2003)
8. Walton, D.N.: *Argumentation Schemes for Presumptive Reasoning*. Erbaum, Mahwah (1996)
9. Prakken, H.: Formal systems for persuasion dialogue. *Knowl. Eng. Rev.* 21(2), 163–188 (2006)

10. Ashley, K., Pinkwart, N., Lynch, C., Alevan, V.: Learning by diagramming supreme court oral arguments. In: Proceedings of the 11th international conference on Artificial intelligence and law, pp. 271–275 (2007)
11. Bench-Capon, T.J.M.: Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation* 13(3), 429–448 (2003)
12. Moore, D.: Dialogue game theory for intelligent tutoring systems. Ph.d. dissertation, Leeds Metropolitan University, Leeds, UK (1993)
13. Sierra, C., Jennings, N.R., Noriega, P., Parsons, S.: A framework for argumentation-based negotiation. In: Proceedings of 4th International Workshop on Agent Theories Architectures and Languages, pp. 167–182 (1998)
14. Mackenzie, J.D.: Question-begging in non-cumulative systems. *Journal of Philosophical Logic*, 117–133 (1979)
15. Kibble, R.: Reasoning about propositional commitments in dialogue. *Research on Language and Computation* 4(2-3), 179–202 (2006)
16. Walton, D.: *Argument Structure: A Pragmatic Theory*. Toronto Press (1996b)
17. Harel, D.: Statecharts: A visual formalism for complex systems. *Science of Computer Programming* 8(3), 231–274 (1987)
18. Drusinsky, D.: Modeling and Verification Using UML Statecharts. In: *A Working Guide to Reactive System Design, Runtime Monitoring and Execution-based Model Checking*, 1st edn., Newnes (2006)
19. Dung., P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77(2), 321–358 (1995)
20. Chamberland, M., Hivon, R., Tardif, J., Bédard, D.: Évolution du raisonnement clinique au cours d'un stage d'externat: une étude exploratoire. *Pédagogie Médicale* 2, 9–17 (2001)
21. Prakken, H.: Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese* 127, 187–219 (2001)
22. Amgoud, L., Maudet, N.: Strategic considerations for argumentative agents (preliminary report). In: *Proceeding of 9th International Workshop on Non-Monotonic Reasoning, Special session on Argument, Dialogue, and Decision*, pp. 399–407 (2002)

Predicting Student Knowledge Level from Domain-Independent Function and Content Words

Claire Williams and Sidney D’Mello

Institute for Intelligent Systems, University of Memphis, USA
{mcwilliams, sdmello}@memphis.edu

Abstract. We explored the possibility of predicting the quality of student answers (error-ridden, vague, partially-correct, and correct) to tutor questions by examining their linguistic patterns in 50 tutoring sessions with expert human tutors. As an alternative to existing computational linguistic methods that focus on domain-dependent content words (e.g., velocity, RAM, speed) in interpreting a student’s response, we focused on function words (e.g., I, you, but) and domain-independent content words (e.g., think, because, guess). Proportional incidence of these word categories in over 6,000 student responses to tutor questions was automatically computed using Linguistic Inquiry and Word Count (LIWC), a computer program for analyzing text. Multiple regression analyses indicated that two parameter models consisting of pronouns (e.g., I, they, those) and discrepant terms (e.g., should, could, would) were effective in predicting the conceptual quality of student responses. Furthermore, the classification accuracy of discriminant functions derived from the domain-independent LIWC features competed with conventional domain-dependent assessment methods. We discuss the possibility of a composite assessment algorithm that focuses on both domain-dependent and domain-independent words for dialogue-based ITSs.

Keywords: student model, function words, content words, expert tutor, LIWC.

1 Introduction

Intelligent Tutoring Systems (ITSs) can never deliver individualized instruction by tailoring their pedagogical strategies to each learner without a model of the learner. As an important advance to traditional computer based training systems, ITSs are expected to adapt their tutoring strategies to the learner’s aptitude, personality, prior-knowledge, goals, progress, and a host of other parameters that presumably impact learning. Hence, researchers in the ITS community have always considered it important to develop a student model to help guide the tutor’s actions. The model can be derived from different sources, such as static trait measures that can be obtained from learner self-reports as well as dynamic measures that can be mined from the stream of behaviors and responses generated by the learner during the course of the session. At the very least, an ITS should have a model of learners’ knowledge levels in order to provide discriminating feedback on their immediate actions and to select problems that will maximize learning.

Consequently, ITS researchers have developed a host of sophisticated algorithms and techniques for student modeling [1], [2]. Pioneered by Cognitive Tutors, model tracing has emerged as a useful way to infer what the student knows and to use this information to tailor the tutorial intervention to the individual student [1]. The power of model tracing is evident in the success of the Cognitive Tutors [3]; however, the substantial knowledge engineering required to construct the domain model is one important limitation.

This limitation has led some researchers to pursue a number of alternate methods to model domain knowledge [4], [5], [6]. One method, which is implemented in AutoTutor, an ITS with conversational dialogues [7], capitalizes on the statistical regularities of natural language. AutoTutor uses Latent Semantic Analysis (LSA) [8], to automatically build a high dimensional semantic model of domain knowledge from textual corpora such as a textbook or general world knowledge. Expected answers to tutor's questions, common misconceptions, and student responses can all be represented as high-dimensional vectors in this semantic space. The model of student's knowledge is considered to be the semantic match (cosine between vectors) between the student's response to the tutor's questions and the set of expected answers. This model is used to select the tutor's feedback (i.e., positive, neutral, negative), the next dialogue move (e.g., hint, prompt, pump, assertion), and the next part of the problem to cover [6].

AutoTutor's model of the student's knowledge can be considered to be domain-dependent because it is derived from the student's actions (i.e., their responses in this case) and the expected answers to the tutor's questions (i.e., from a curriculum script). Although this is generally the case with most model tracing methods, perhaps an alternative method of tracking student knowledge for dialogue based ITSs is through the words they use. In particular, we are referring to *function* words (e.g., pronouns, propositions, conjunctions) and domain-independent *content* words (e.g., mistake, sad, certain), but not domain-dependent *content* words. These function words have little meaning in themselves, but play an important role in specifying relationships between the content words.

But can a domain-independent (i.e., no curriculum script, or textbook) analysis of the words in learners' responses (called textual features) be diagnostic of their knowledge levels? There is some evidence to suggest that this is a distinct possibility. Recent advances in computational psycholinguistics have demonstrated that the words people use can predict complex phenomenon such as personality, deception, emotion, and even physical and mental health outcomes [9], [10], [11], [12]. Therefore, it is plausible to expect that a textual analysis of learners' responses might be indicative of their knowledge levels, a possibility that is explored in the present paper.

The present study used the Linguistic Inquiry and Word Count (LIWC) [13], a dictionary-based word counting tool, to compute features (i.e., word categories), that might be predictive of learners' knowledge levels. LIWC automatically analyses a text and assigns words to a number of categories of theoretical and practical significance (e.g., function words, affect words, cognitive words). LIWC was originally designed to determine what features of writing might predict one's physical and mental health [13], [14], [15], [16]. However, recent research [17] has found that LIWC can predict a variety of phenomena such as one's physical health, health-related

behaviors (e.g., alcohol use), and most interestingly, outcomes such as class performance and attitudes towards academics [17].

Although the power of simple word counting is somewhat surprising, LIWC's remarkable ability can be attributed to the Linguistic Determinism Hypothesis [18]. Simply put, this theory states that, "thought is determined by language" [18], [19]. In other words, one's thought, or cognitive activities, exude from the words one uses [20]. Within the context of ITSs, cognitive states refer to students' knowledge levels and can be inferred from the types of answers (i.e., error-ridden, vague, partially-correct, correct answers) that students provide to tutor questions [20]. This information is crucial for ITSs to determine how to provide feedback and to select the next dialogue move to scaffold the student towards mastering the content. Therefore, we analyzed student responses from an existing corpus of 50 tutoring sessions between students and expert human tutors, with an eye for implementing any insights gleaned into dialogue-based ITSs.

2 Expert Tutoring Corpus

The corpus consisted of 50 tutoring sessions between 39 students and ten expert tutors on the domains of algebra, geometry, physics, chemistry, and biology. The students were having difficulty in either science or math-related courses and were either recommended for tutoring via school personnel or voluntarily sought professional tutoring help.

The expert tutors were recommended by academic support personnel from public and private schools in a large urban school district. All of the tutors had longstanding relationships with the academic support offices that recommended them to parents and students. The criteria for being an expert tutor were as follows: (a) have a minimum of five years of one-to-one tutoring experience, (b) have a secondary teaching license, (c) have a degree in the subject in which they tutor, (d) have an outstanding reputation as a private tutor, and (e) have an effective track record (i.e., students who work with these tutors show marked improvement in the subject areas in which they are receiving tutoring).

Fifty one-hour tutoring sessions were videotaped and transcribed. To capture the complexity of what transpires during a tutoring interaction a Tutor Coding Scheme and a Student Coding Scheme were used to classify every tutor and student dialogue move [21]. A total of 47,256 dialogue moves were coded in the 50 hours of tutoring with kappa scores of .92 and .88 for tutor and student moves, respectively.

Although detailed descriptions of the coding schemes [21] are beyond the scope of this paper, of relevance, however, is the coding scheme of the student answer moves. Student answers were coded as: (a) *no* answers (e.g., "Umm." "Mmm."), (b) *error-ridden* answers (e.g., "Prokaryotes are human and eukaryotes are bacteria"), (c) *vague* answers (e.g., "Because it helps to, umm, you know"), (d) *partially-correct* answers (e.g., "It has to do with the cells"), and (e) *correct* answers (e.g., "In meiosis, it starts out the same with one diploid"). These five answer categories comprised 28.6% of all student moves. 2.8% of students' answers were error-ridden answers, 4.6% were vague, 5.7% were partially-correct, and 14% were correct. No answers occurred 1.5% of the time.

3 Linguistic Inquiry and Word Count (LIWC)

LIWC (pronounced “Luke”) is a validated computational tool that analyzes bodies of text using dictionary-based approaches. At the heart of the LIWC machinery is a dictionary that maps words to word categories. For example, “crying”, and “grief” are words that map on to the *sad* category, while “love” and “nice” are words that belong to the *positive emotion* category. The version of LIWC (LIWC2007) used in the current study provides measures for approximately 70 word categories. These include linguistic words (e.g., pronouns, verbs), psychological constructs (e.g., causations, sadness), personal constructs (e.g., work, religion), and paralinguistic features (e.g., speech disfluencies). LIWC operates by analyzing a transcript of text and counting the number of words that belong to each word category. A proportional score for each word category is then computed by dividing the number of words in the text that belong to that category by the total number of words in the text.

Our initial analysis focused on a subset of LIWC’s features that we expected to be predicative of students’ answer types. This subset included pronouns and cognitive terms (see Table 1). These feature banks were considered because they have been found to be diagnostic of cognitive processes in previous analyses [13], [17], [18], [19], [20], [22], [23], [24].

Table 1. Pronouns and cognitive features derived from LIWC

Pronouns	Cognitive Features
<p>P1. Personal pronouns (I, her) P2. 1st person singular pronouns (I, me) P3. 1st person plural pronouns (we, us) P4. 2nd person pronouns (you, yours) P5. 3rd person singular pronouns (she, him) P6. 3rd person plural pronouns (they, their) P7. Impersonal pronouns (it, those)</p>	<p>C1. Insight (think, know) C2. Causation (because, effect) C3. Discrepancy (should, would) C4. Tentative (maybe, perhaps) C5. Certainty (always, never) C6. Inhibition (block, constrain) C7. Inclusive (and, with) C8. Exclusive (but, without)</p>

4 Results and Discussion

The present analyses had two goals. The first goal was to assess whether features computed via LIWC could be predictive of the quality of students’ answers and to identify the most predictive features. The second goal was to consider the possibility of automatically classifying a student’s response on the basis of words they use. Correlational, multiple regression, and linear discriminant analyses were performed to address these two goals.

4.1 Multiple Regression Analyses

The primary goal of these analyses was to find a set of predictors that are most diagnostic of learners’ answer types. We addressed this goal by constructing a series of

multiple regression (MLR) analyses with the textual features as independent variables and the proportions of student answer types (i.e., error-ridden, vague, partially-correct, correct) as dependent variables.

There is also the important subgoal of quantifying the predictive power of each feature set (i.e., pronouns versus cognitive). Hence, The MLR models were constructed in two phases. First, we independently compared the predictive power of each feature set. Next, the most diagnostic predictors from each set were collectively included as predictors of student answer types.

The analyses proceeded by submitting the transcribed text of students' responses for the five answer types to LIWC and computing incidence scores for the 15 potential predictors listed in Table 1. The unit of analysis was each student's session; hence, multiple exemplars of the same answer category were grouped together. There was an insufficient amount of text to warrant a linguistic analysis for the no-answer category; hence, the subsequent analyses focus on the other four answer categories.

Prior to constructing the regression models, we reduced the set of predictors with a correlational analysis. The analyses proceeded by constructing a correlational matrix for each answer type. This was a 15×4 (feature \times answer type) matrix that consisted of the correlation between the LIWC features and the proportional occurrence of each answer category. Each matrix was examined separately and we selected features that significantly correlated with at least one of the student answer types. This procedure narrowed the landscape of potential predictors to four linguistic and six cognitive features (P1, P2, P6, P7, C1 C3, C4, C6, C7, and C8).

There were two regression models constructed for each answer type (one for pronouns and the other for cognitive terms), yielding eight models in all. There were inherent differences in the number of words in each answer type (mean number of words were 58 for error-ridden, 88 for vague, 176 for partially-correct, and 268 for correct answers), which might cause some confounds in comparing the models. Hence, the regression models were constructed in two steps. In Step 1, the numbers of words in the responses were entered as the only predictor and the residual variance was passed to the Step 2 models which consisted of the LIWC predictors. The results indicated that none of the Step 1 models were statistically significant, yet all but one of the Step 2 models were significant, suggesting that the patterns described below cannot be merely attributed to differences in verbosity.

Space constraints preclude an extensive discussion of the regression models constructed by examining each feature set independently. Hence, the current discussion is limited to comparison of the predictive power of the pronouns versus cognitive terms (coefficients will be examined in the subsequent analysis). It appears that the pronoun features yielded significant ($p < .05$) models for vague ($R_{adj}^2 = .103$), partially-correct ($R_{adj}^2 = .337$), and correct ($R_{adj}^2 = .187$) answers, but not error-ridden answers ($R_{adj}^2 = .0$). The cognitive features were effective in predicting all four answer types ($R_{adj}^2 = .115, .212, .383, .212$, for error-ridden, vague, partially-correct, and correct answers), respectively. Hence, the results indicate that the cognitive feature set was more effective in predicting students' answer types than the pronouns because they could predict all four categories and explained more variance (mean $R_{adj}^2 = .157$ for pronouns and $.231$ for cognitive features).

The next set of MLR models were constructed by an additive combination of the most diagnostic predictors from the pronoun and cognitive feature sets. There was one pronoun and one cognitive predictor for vague, partially-correct, and correct answers, but only one cognitive predictor for error-ridden answers. As before, the effects of overall response verbosity were partialled out in the regression models.

The parameters of the multiple regression models are listed in Table 2. It appears that the LIWC predictors explained 27% of the variance averaged across the four answer types. This is consistent with a medium to large effect for a statistical power of .8 [25], and supports the hypothesis that it is possible to predict the learners' knowledge levels by monitoring the domain-independent pronouns and cognitive words in their responses.

Table 2. Parameters of multiple regression models

Regression Model	Error	Vague	Partial	Correct
Model Parameters				
<i>F</i>	3.67	6.18	9.39	6.53
<i>df</i> ₁ , <i>df</i> ₂	2,39	3,35	3,36	3,39
<i>P</i>	.035	.002	<.001	.001
<i>Adj. R Sq.</i>	.115	.290	.392	.283
Coefficient Weights (β)				
Personal Pronouns		.390		
1 st Person Singular Pronouns				-.347
3 rd Person Plural Pronouns			-.277	
Discrepant Words	-.395	.374	.551	-.391

Turning our focus to the pronouns listed in Table 2, it appears that students used more personal pronouns when providing vague answers (e.g., “Oh, that’s what *I* am thinking of.”). In contrast, 1st person singular pronouns were rare when students answered correctly, suggesting that they are less focused on themselves and more focused on the material when they know the answer. The results also indicated that students used less 3rd person plural pronouns when they provided partially-correct answers.

One interesting finding was that discrepant word usage was the critical cognitive feature that predicted all four answers categories. In particular, students used more discrepant terms (e.g., should, would, could) when the provided vague and partially-correct answers, but less discrepant terms when providing error-ridden and correct answers. Discrepant terms with vague answers signal that students are hedging to cover their lack of knowledge (e.g., “problems are, actually I mean, you *could* just kind of go over the very basics first...”), while their use with partially-correct answers is consistent with a degree of uncertainty and incomplete information (e.g., “and, you *would* divide by the mass of that”). In contrast, discrepant words are not used when students think they know the answer but are wrong (i.e., error-ridden answers) or when they actually know the answer (i.e., correct answers).

4.2 Linear Discriminant Analysis

Our results so far indicate that the LIWC features extracted from students' answers are predictive of their knowledge levels. Although useful for identifying the important predictors, the regression analyses are of limited applicability because they were constructed from transcripts of the entire session (i.e., individual moves were grouped together and submitted to LIWC). A model-tracing tutor needs to monitor student's knowledge level after each student action or dialogue move. We addressed this problem by computing the LIWC features in each student response and assessed whether these features could discriminate between the different answer types.

The analyses proceeded by randomly selecting 2,345 student responses associated with error-ridden, vague, partially-correct, and correct answers, respectively (approximately 600 responses for each answer type). These were submitted to LIWC and a 70-item feature vector was obtained for each response. A linear discriminant analysis was then used to classify the responses into one of the four answer categories. The discriminant functions were constructed from the entire LIWC feature set, instead of just the pronouns and cognitive terms, in order to explore the full potential of LIWC in predicting answer types.

The analysis yielded three functions that were all statistically significant ($p < .001$). The functions achieved an accuracy of 45.2% in automatically discriminating between the four answer types (chance = 25%). This is an impressive finding given that we are making fine-grained discriminations among answer types without any knowledge of the domain.

Additional insights can be gleaned from the confusion matrix presented in Table 3. It appears that the model was most successful in detecting vague answers, followed by correct answers, and then by partially-correct answers. The detection accuracy for error-ridden answers was below the chance level. This finding is consistent with the multiple regression analyses, thereby providing converging evidence that the features have difficulty in detecting error-ridden answers. It is also informative to note that error-ridden answers are confused with correct answers; however, correct answers are rarely classified as being error-ridden.

Table 3. Confusion matrix (%) from linear discriminant analysis

Observed	Predicted			
	<i>Correct</i>	<i>Error</i>	<i>Partial</i>	<i>Vague</i>
<i>Correct</i>	52.67	10.50	17.17	19.67
<i>Error</i>	31.32	21.98	25.09	21.61
<i>Partial</i>	30.72	11.69	41.74	15.86
<i>Vague</i>	18.17	6.33	13.00	62.50

5 General Discussion

Most researchers are aware that a spoken message has a linguistic (i.e., the words used) and a paralinguistic (e.g., pitch, formants, etc) component. While the linguistic

component transmits the content of the message, the paralinguistic component conveys information about the speaker's affective states, cognitive states, urgency, and other informative features. What might come as a surprise to some is that the linguistic component of a message itself can provide a window into the mind of the speaker, above and beyond the explicit content of the message. This possibility was explored in the current paper as a solution to the important problem of evaluating student responses to tutor questions.

Although most language-based model tracing algorithms focus on domain-related content words (e.g., velocity, speech RAM, computer) to evaluate student responses, we hypothesized that important information can be extracted from function words (e.g., I, you, and, with, should, could) and domain-independent content words (e.g., hope, mistake, regret, consider). The results indicate that word categories extracted from LIWC were moderately effective (i.e., accuracy rates yielded an 81% improvement over the baseline) in discriminating between error-ridden, vague, partially-correct, and correct answers, thereby providing some evidence to confirm this hypothesis.

Our findings have important implications for dialogue-based ITSs that use computational linguistic techniques to evaluate student responses. Many of the answer assessment algorithms implemented in these ITSs perform some sort of word weighting where the domain-independent words are effectively filtered out. For example, the inverse word frequency weighted overlap (IWFOW) algorithm used in conjunction with LSA in AutoTutor weights each word relative to its inverse frequency in the English language [26]. LSA, itself, uses log-entropy weighting to emphasize low-frequency words in the corpus. As a consequence, higher frequency words such as closed-class function words (e.g., and, but, a, the) have comparatively low weights, while lower frequency words (e.g., RAM, system, speed) have higher weights.

We do not claim that ITSs should exclusively focus on the domain-independent function and content words in lieu of critical terms related to the tutorial domain. Obviously, the 45.2% accuracy score obtained by the discriminant analysis is not sufficiently accurate to model students' knowledge levels. However, it should be noted that the correlation between the predicted answer categories from the discriminant model and the human coded answer categories was .329 ($p < .05$). This is quantitatively similar to the .29 sentence-level correlation obtained by LSA and the .39 correlation obtained by word overlap [6]. These analyses use weighting metrics to hone in on the less-frequent domain-dependent content words, while overlooking the abundant domain-independent function and content words.

Our claim is that a composite evaluation algorithm that considers both of these perspectives is the most defensible position. Such an algorithm could use IWFOW or LSA to evaluate the substantive content in the student responses as well as the discriminant functions derived in the present paper. There is the question of whether the combination of these two evaluation methods will result in superadditive, additive, or redundant effects. When there are superadditive effects, the classification performance from both approaches will be superior to an additive combination of the individual methods. Simply put, the whole will be greater than the sum of the parts. An alternative hypothesis would be that there is redundancy between the methods. When there is redundancy, the combination of the two methods yields negligible incremental gains. Answers to this question will require further research and development.

Acknowledgements. This research was supported by the by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A080594. The opinions expressed are those of the authors and do not represent views of the funding agencies.

References

1. Anderson, J., Corbett, A., Koedinger, K., Pelletier, R.: Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences* 4(2), 167–207 (1995)
2. VanLehn, K., Lynch, C., Taylor, L., Weinstein, A., Shelby, R., Schulze, K.: Minimally Invasive Tutoring of Complex Physics Problem Solving. In: Cerri, S.A., Gouardères, G., Paraguaçu, F. (eds.) *ITS 2002. LNCS*, vol. 2363, pp. 367–376. Springer, Heidelberg (2002)
3. Koedinger, K., Anderson, J., Hadley, W., Mark, M.: Intelligent Tutoring Goes to School in the Big City. *Journal of Artificial Intelligence in Education* 8, 30–43 (1997)
4. Kurup, M., Greer, J.E., McCalla, G.: The Fawly Article Tutor. In: Frasson, C., McCalla, G.I., Gauthier, G. (eds.) *ITS 1992. LNCS*, vol. 608, pp. 84–91. Springer, Heidelberg (1992)
5. Heift, T., Schulze, M.: Error Diagnosis and Error Correction in Computer-Assisted Language Learning. *CALICO* 20(3), 433–436 (2003)
6. Graesser, A., Penumatsa, P., Ventura, M., Cai, Z., Hu, X.: Using LSA in AutoTutor: Learning Through Mixed-Initiative Dialogue in Natural Language. In: Landauer, T., McNamara, D., Dennis, S., Kintsch, W. (eds.) *Handbook of Latent Semantic Analysis*, pp. 243–262. Erlbaum, Mahwah (2007)
7. Graesser, A., Lu, S., Jackson, G., Mitchell, H., Ventura, M., Olney, A., Louwerse, M.M.: AutoTutor: A Tutor with Dialogue in Natural Language. *Behavioral Research Methods, Instruments, and Computers* 36, 180–193 (2004)
8. Landauer, T., McNamara, D., Dennis, S., Kintsch, W. (eds.): *Handbook of Latent Semantic Analysis*. Erlbaum, Mahwah (2007)
9. Campbell, R.S., Pennebaker, J.W.: The Secret Life of Pronouns: Flexibility in Writing Style and Physical Health. *Psychological Science* 14, 60–65 (2003)
10. D'Mello, S., Dowell, N., Graesser, A.: Cohesion Relationships in Tutorial Dialogue as Predictors of Affective States. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Graesser, A. (eds.) *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, pp. 9–16. IOS Press, Amsterdam (2009)
11. Hancock, J.T., Curry, L., Goorha, S., Woodworth, M.T.: On Lying and Being Lied to: A Linguistic Analysis of Deception. *Discourse Processes* 45, 1–23 (2008)
12. Mairesse, F., Walker, M.A., Mehl, M.R., Moore, R.K.: Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. *Journal of Artificial Intelligence Research* 30, 457–500 (2007)
13. Pennebaker, J.W., Stone, L.D.: Words of Wisdom: Language Use Over the Life Span. *Journal of Personality and Social Psychology* 83(2), 291–301 (2003)
14. Pennebaker, J.W.: Writing About Emotional Experiences as a Therapeutic Process. *Psychological Sciences* 8, 162–166 (1997)
15. Pennebaker, J.W., Francis, M.E., Booth, R.J.: *Linguistic Inquiry and Word Count (LIWC)*. Erlbaum, Mahway (2001)
16. Stiles, W.B.: *In Describing Talk: A Taxonomy of Verbal Response Modes*. Sage, Newbury Park (1992)

17. Pennebaker, J.W., King, L.A.: Linguistic Styles: Language Use as an Individual Difference. *Journal of Personality and Social Psychology* 77, 1296–1312 (1999)
18. Boroditsky, L.: Does Language Shape Thought?: Mandarin and English Speakers' Conception of Time. *Cognitive Psychology* 43, 1–22 (2001)
19. Whorf, B.: In Language, Thought, and Reality: Selected Writings of Benjamin Lee Whorf. MIT Press, Cambridge (1956)
20. Zhang, T., Hasegawa-Johnson, M., Levinson, S.E.: Cognitive State Classification in a Spoken Tutorial Dialogue System. *Speech Communication* 48, 616–632 (2006)
21. Person, N., Lehman, B., Ozbun, R.: Pedagogical and Motivational Dialogue Moves Used by Expert Tutors. In: 17th Annual Meeting of the Society for Text and Discourse, Glasgow, Scotland (2007)
22. Groom, C.J., Pennebaker, J.W.: Words. *Journal of Research in Personality* 36, 615–621 (2002)
23. Pennebaker, J.W., Lay, T.C.: Language Use and Personality During Crises: Analyses of Mayor Rudolph Giuliani's Press Conference. *Journal of Research in Personality* 36, 271–282 (2002)
24. Kitayama, S.: Interaction Between Affect and Cognition in Word Perception. *Journal of Personality and Social Psychology* 58(2), 209–217 (1990)
25. Cohen, J.: A Power Primer. *Psychological Bulletin* 112(1), 155–159 (1992)
26. Chipman, P.: An Analysis and Optimization of AutoTutor's Student Model (Unpublished Master's Thesis). The University of Memphis, Memphis (2008)

KSC-PaL: A Peer Learning Agent

Cynthia Kersey¹, Barbara Di Eugenio², Pamela Jordan³, and Sandra Katz⁴

¹ Lewis University

² University of Illinois at Chicago

³ LRDC and Biomedical Informatics, University of Pittsburgh

⁴ LRDC, University of Pittsburgh

Abstract. We have developed an artificial agent based on a computational model of peer learning we developed. That model shows that shifts in initiative are conducive to learning. The peer learning agent can collaborate with a human student via dialog and actions within a graphical workspace. This paper describes the architecture and implementation of the agent and the user study we conducted to evaluate the agent. Results show that the agent is able to encourage shifts in initiative in order to promote learning and that students learn using the agent.

Keywords: Peer Agent, Knowledge Co-construction, Initiative.

1 Introduction

Research shows that collaboration promotes learning, potentially for all of the participants [2,7,12]. Similarly, studies in peer tutoring demonstrate that there are cognitive gains for both the tutor and the tutee [1,5,11]. However, the study of peer learning from a computational perspective is still in the early stages. Although some researchers have attempted to develop simulated peers [3,14], there is very little research on what constitutes effective peer interaction to guide the development of effective peer learning agents.

In our previous work we derived a model of peer interactions that was suitable for incorporation in an agent. This model operationalizes *Knowledge Co-construction* [8] via the notion of initiative shifts in dialogue. We have incorporated this model in an innovative peer learning agent, KSC-PaL, that is designed to collaborate with a student to solve problems in the domain of computer science data structures.

This paper presents the details of the implementation and evaluation of KSC-PaL. We start by summarizing the computational model of peer learning that is incorporated into the agent, followed by a description of the system design and architecture. We conclude with the results of the user study we conducted to evaluate the agent.

2 Computational Model

We have performed an extensive corpus analysis [10] in order to derive a computational model of Knowledge Co-construction (KCC). This construct explains

the effectiveness of peer learning by postulating that learning is enhanced when students work together to construct knowledge. An earlier study by Hausmann et al. [8] had extended the analysis of KCC by incorporating relations, such as elaborate and criticize, within KCC episodes. However, our analysis found that these relations were not only difficult to identify but did not correlate with learning in our corpus. Hence, we looked for simpler but principled correlates of KCC. We found those in the linguistically motivated notion of *initiative shifts* in dialogue. Our analysis found a strong relationship between initiative shifts and KCC episodes. A paired t-test showed that there were significantly more initiative shifts in the annotated KCC episodes compared with the rest of the dialogue ($t(57) = 3.32, p = 0.0016$). The moderate effect difference between the two groups (effect size = 0.49) shows that there is a meaningful increase in the number of initiative shifts in KCC episodes compared with problem solving activity outside of the KCC episodes. Additionally, we found moderate correlations of learning with both KCC ($R^2 = 0.14, p = 0.02$) and with initiative shifts ($R^2 = 0.20, p = 0.00$).

Since the corpus analysis showed a correlation between initiative and KCC and between initiative and learning, the next step was to identify ways for KSC-PaL to encourage such shifts in initiative. We explored two different methods to do so. One method is based on the observation that student uncertainty (hedging) may lead to a shift in initiative. The other is based on related literature [4,15] which shows that certain conversational cues, other than hedging, lead to shifts in initiative. Our analysis showed that the following cues were most likely to lead to initiative shift and to increase knowledge score (which was computed using the student model described in section 3.3): hedging, using prompts, making mistakes intended to incite student criticism and requesting feedback.

3 KSC-PaL

Based on this analysis, we developed a peer learning agent, KSC-PaL. The core of KSC-PaL is the TuTalk system [9]. TuTalk is a dialogue management system that supports natural language dialogues for educational applications and allows for both tutorial and conversational dialogues. In developing the agent we extended TuTalk by adding a graphical user interface, replacing TuTalk’s student model and augmenting TuTalk’s planner to implement the model discussed above.

The interface manages communication between TuTalk and the student. Students communicate with the agent using typed natural language and graphical actions within a graphical user interface. The student input is processed by the interface and its related modules into an appropriate format and passed to TuTalk. Since TuTalk’s interpretation module is not able to appropriately handle all student utterances and we wanted to avoid interpretation issues impacting our results, a human interpreter assists in this process. Additionally TuTalk requests assistance from the Student Model/Dialogue Planner (SMDP) to manage the dialogue in order to appropriately shift initiative and encourage learning. These modules are described below in more detail.

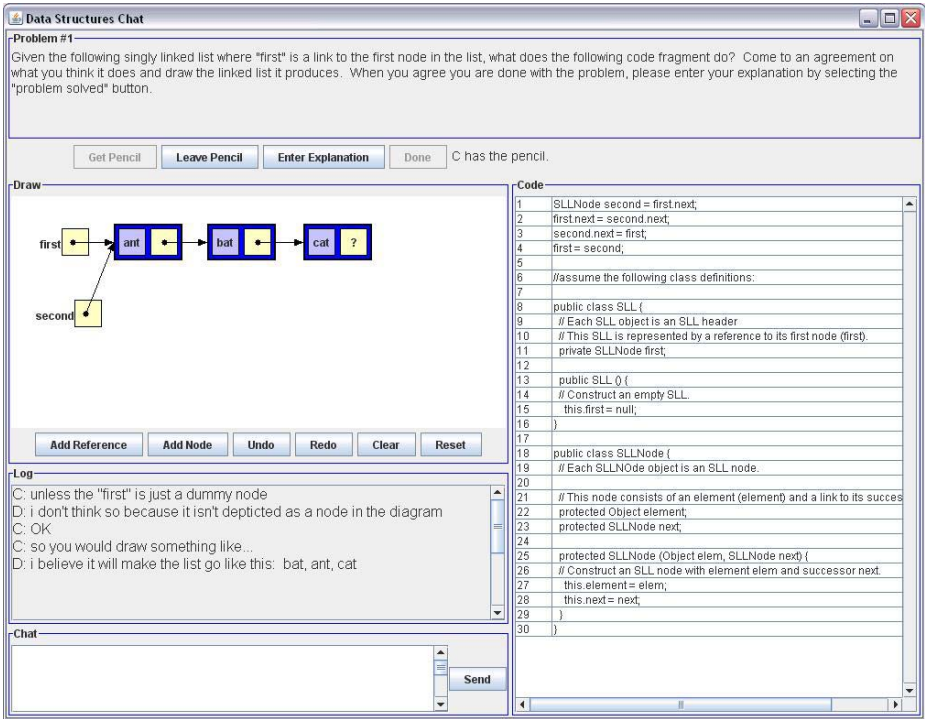


Fig. 1. KSC-PaL user interface

3.1 Interface

The user interface consists of four distinct areas (see figure 1):

1. Problem display: Displays the problem description.
2. Code display: Displays the code from the problem statement.
3. Chat Area: Allows for typed user input and an interleaved dialogue history of the student and the agent.
4. Drawing area: Here users can diagram data structures to aid in the explanation of parts of the problem being solved. The drawing area has objects representing nodes and links that can be used to build lists.

The interface includes a preprocessor module which takes as input a student's utterances and actions and modifies them so that they can be recognized by TuTalk. This preprocessor consists of a spell corrector and a graphical actions interpreter that interprets the student's drawing and coding actions and passes them to TuTalk as natural language utterances.

3.2 Human Interpreter

Given the limitations of current technology for natural language understanding, a human interpreter was incorporated to assist in the disambiguation of

student utterances. The interpreter receives a student utterance along with a list of possible matching concepts from TuTalk. The interpreter then selects the most likely matching concepts from TuTalk, thus assisting in natural language interpretation. If the student utterance doesn't match any of these concepts, a second list of concepts, containing student initiative utterances, are presented to the interpreter. If none of these match then all known concepts are presented to the interpreter for matching. Note that the interpreter has a limited, predetermined set of choices, corresponding to the concepts that TuTalk is aware of. In this way, his/her intervention is circumscribed.

The interpreter also plays a role in the interpretation of graphical actions. The natural language interpretation of the drawing or coding action is first sent to the interpreter. He/she then verifies that the interpretation is valid or selects an alternate interpretation from a list of known graphical actions and sends it on to TuTalk for processing.

Additionally, since interpreting student input of a solution would require extensive natural language processing, the interpreter also matches the solutions entered by the student to a limited range of possible solutions, such as *correct*, *incomplete* or *incorrect*.

3.3 Student Model/Dialogue Planner (SMDP)

KSC-PaL's planner selects scripts and responses to student initiative to manage initiative shifts. TuTalk uses *scenarios* for guiding the dialogues. These scenarios contain both the *recipe* (script) and *concepts*, which are linguistic concepts used to realize the dialogue. Scripts are hierarchical in nature and consist of a sequence of goals for addressing a topic. Goals usually involve multiple steps where each step consists of an initiation followed by one or more responses. Generally, the initiation is an agent utterance and responses are possible ways in which a student can respond. However, when using mixed-initiative, the initiation could represent a student utterance while the responses are potential agent replies to the student's utterance. Additionally, TuTalk allows for alternative recipes to achieve a goal.

In drafting the scripts for KSC-PaL, we authored goals that would encourage shifts in initiative as well as goals that would not encourage initiative shifts. Similarly in drafting responses to student initiative, we drafted both initiative-shifting responses as well as responses that would not likely shift initiative. The agent encourages initiative shifts by using prompts, hedging, requesting feedback from the student and encouraging student criticism by intentionally making errors in problem solving. TuTalk's planner does not manage these options to the level required by the agent, so a planning module was added to make choices on goal implementation and agent response with the objective of managing shifts in initiative.

This planner was combined with the student model to create the Student Model/Domain Planner (SMDP). The SMDP consists of a server that manages communication with TuTalk, a student model, an initiative module that tracks

initiative shifts and a planner that makes decisions based on the current state of initiative and student knowledge.

Student Model. The agent requires a student model to track the current state of problem solving as well as estimate the student’s knowledge of concepts involved in solving the problem in order to guide its behavior. Since TuTalk’s student model does not provide these capabilities, a student model which incorporates problem solution graphs [6] was added to the agent. Solution graphs are Bayesian networks where each node represents either an action required to solve the problem or a concept required as part of problem solving. A user’s utterances and actions are then matched to these nodes. This provides the agent with information related to the student’s knowledge as well as the current topic under discussion.

Initiative Tracker. On receiving a student or agent utterance or action from the SMDP server, the initiative tracker codes the turn with either student initiative or agent initiative. The tracker contains a classifier for natural language utterances and a separate classifier for drawing and coding actions. Natural language utterances are parsed using the Stanford Maximum Entropy Tagger [13] to provide the appropriate features for use by the initiative classifier. When classifying a drawing or coding action, the initiative tracker retrieves the student knowledge score for use by the classifier. Once the turn is classified, it is determined whether a shift in initiative has occurred by comparing the current classification with the classification of the previous turn.

When requested by the planner, the initiative tracker returns the average level of initiative shifts. This is computed by dividing the number of initiative shifts by the total number of turns.

Planner Module. Requests for goal implementation and requests for agent response are managed by the planner module. Two factors determine whether a goal implementation or response that encourages an initiative shift will be selected: (1) the current level of initiative shifts and (2) the change in the student’s knowledge score. Initiative shifts are tracked using the initiative tracker module described above and knowledge levels are maintained in the student model. Goals or responses are selected to encourage initiative shifts when the average level of initiative shifts is less than 0.2117 (mean initiative shifts in KCC episodes as calculated from corpus data) and the student’s knowledge level has not increased since the last time a request for goal implementation or response was requested.

If the planner has determined that an initiative shift should be encouraged, it selects among alternatives based on the holder of initiative in the previous utterance/action and a label associated with each of the potential goal implementations or responses. For example, to encourage an initiative shift when the initiative holder for the previous utterance/action was the student and the agent has the choice of goal implementations labeled *correct*, *partial-correct* and *incorrect*, the agent will select the goal implementation labeled *correct* because it is likely to result in a shift of initiative.

4 Evaluation

We developed two versions of KSC-PaL to test the effectiveness of the model of KCC described above. In the *experimental* version of the agent (PaL), goal versions and responses to student utterances are selected by the planner to maintain a high level of shifts in initiative. In the *control* version (PaL-C), the planner is not consulted for goal versions or responses. Additionally, the script was modified to remove those utterances that were identified as likely to shift initiative: incorrect statements, hedges, prompts and requests for feedback.

4.1 User Study

We collected interactions of 25 students, where 13 interacted with PaL and 12 interacted with PaL-C. At the beginning of the session, each student was given a five question pre-test to evaluate his or her knowledge prior to interacting with the agent. Prior to problem solving, the students were given a short tutorial on using the interface. They then solved two linked list problems with the agent. At the conclusion of problem solving, students were given a post-test, identical to the pre-test. Additionally, they were asked to fill out a questionnaire to assess their satisfaction with the agent.

4.2 Effect on Learning

In order to investigate whether students learned using KSC-PaL, we first performed a paired t-test of pre-test and post-test scores. This analysis showed that overall students learn using KSC-PaL (see table [1](#)). T-test analysis also shows that there is a significant difference between pre-test and post-test in the experimental condition and a trend toward a significant difference in the control condition. However, there is no significant difference between the gains in the two groups.

4.3 Initiative Shifts and Learning

In both conditions, the agent tracks the initiative holder in each utterance using the classifiers described above. A manual annotation of these utterances showed

Table 1. Learning using KSC-PaL

Condition	N	Pre-test M	Post-test M	gain	<i>t</i>	<i>p</i>
KSC-PaL (all students)	25	0.61	0.68	0.07	2.90	0.01
PaL	13	0.60	0.66	0.06	2.55	0.02
PaL-C	12	0.62	0.69	0.07	2.03	0.06
PaL plus upper quartile						
PaL-C subjects	18	0.61	0.68	0.07	3.29	0.00
PaL-C less upper quartile						
PaL-C subjects	7	0.66	0.66	0.00	-0.96	ns

that the level of accuracy of the classifiers was as expected. 747 of 937 of utterances and drawing actions (80.15%) were correctly classified. However, in order to evaluate the effectiveness of initiative shifts, the following analysis uses the utterances manually annotated for initiative.

To examine the impact of initiative shifts on learning, we used two measures of shifts: (1) the number of shifts and (2) normalized initiative shifts, calculated by dividing the number of initiative shifts by the total number of utterances and drawing actions for the session.

Given that the control condition does not encourage initiative shifts but neither does it prevent them, we combined the experimental subjects with those control subjects whose interactions showed high levels of initiative shifts, i.e. where the amount of initiative shifts falls in the upper quartile of the number of initiative shifts for the combined group. As shown in table 1, when compared with the remaining control condition subjects there is a difference in learning between the two groups. A t-test performed on the gains between the two groups showed the difference is significant ($t = 2.35$, $p = 0.03$). The effect size (d) is 0.18 which is considered a moderate difference.

Additionally, using multiple linear regression, the measures described above were used as predictors of post-test score after regressing out the impact of pre-test score. Table 2 shows that while the correlations are significant or trending toward significance, the impact is relatively small. If this same analysis is applied to those subjects with a pre-test score below the mean, there is a larger impact of initiative shifts on post-test score. Analysis of high pre-test subjects showed no significant correlation of post-test score with initiative shifts or normalized initiative shifts.

Table 2. Impact of Initiative Shifts on Learning

Predictor of Post-test	β	R^2	p
Initiative shifts	0.24	0.03	0.06
Normalized initiative shifts	0.28	0.01	0.02
Low pre-test subjects (n=14)			
Initiative shifts	0.45	0.07	0.09
Normalized initiative shifts	0.49	0.16	0.04

4.4 Agent’s Ability to Shift Initiative

In the experimental condition, KSC-PaL attempts to shift initiative in order to maintain a certain level of initiative shifts. In retrospect, this threshold appears to be set too low, since KSC-PaL rarely selected responses or goal implementations that would encourage a shift in initiative. Only 34 of the 200 requests for response or goal implementation (17%) resulted in a selection to shift initiative.

Therefore, in order to examine the effectiveness of encouraging initiative shifts, we used an alternative method. As mentioned above, the script for the experimental condition included agent utterances that encourage initiative shifts, including instances where no request for agent response or goal selection would

be made. These types of utterances were generally excluded from the script for the control condition. Thus, the students in the experimental condition were more likely to encounter those utterances that encourage initiative shifts. To examine the impact of this difference, the dialogues in the user study were semi-automatically annotated with the following encouragers of initiative shifts:

- hedge
- request for feedback
- incorrect statements
- prompts

This was accomplished by collecting all of the agent responses and identifying those responses that fall into one of the categories listed above. Since the agent has a limited set of responses, the transcripts were queried for matching utterances and automatically coded with the appropriate labels.

First we investigated whether the number of shifts encouraging utterances had an impact on learning by using multiple regression to predict post-test score using pre-test score + initiative shift utterances. This was not statistically significant.

We then ran a t-test to see if the number of utterances tagged as shift encouragers differed between control sessions and experimental sessions. An unpaired t-test showed that they were significantly different ($t = 3.28$, $p = 0.0036$). We then used linear regression to see if there was a relationship between the number of these shift inducing utterances and number of initiative shifts that occurred. This was also significant ($\beta = 0.40$, $R^2 = 0.16$, $p = 0.04$). This result suggests that these shift encouragers do have an impact on the number of initiative shifts that occur during a problem solving session.

4.5 Student Satisfaction

At the conclusion of problem solving, students were asked to complete a short survey related to their satisfaction using KSC-PaL. The survey consisted of statements to which the students were asked to rate their level of agreement

Table 3. Student Survey - Average Responses

Statement	Control Condition		Experimental Condition	
	M	sd	M	sd
The agent helped me learn about linked lists	3.54	0.97	3.08	1.38
Working with the agent is like working with a classmate	3.23	1.23	3.38	1.26
I would use the agent on a regular basis, for other topics (like trees)	3.92	0.86	3.31	1.11
The agent understands what I am saying	3.77	1.16	3.23	1.09
The agent responds appropriately to what I am saying	3.54	1.33	3.46	1.26
I found what the agent said repetitive	3.00	0.91	3.08	1.32
I felt like I had control over solving the problems, and the agent wasn't trying to take charge too often.	1.49	3.31	3.69	1.38

with. Responses were on a 5 point Likert scale, with 1 representing strongly disagree and 5 representing strongly agree. The statements on the survey are shown in Table 3.

There were no significant differences between the responses to these questions for those in the control condition versus those in the experimental condition suggesting that attempting to shift initiative does not have a negative impact on student satisfaction with the agent.

5 Conclusion and Future Work

We implemented a peer learning agent, KSC-PaL, based on the results of an extensive corpus analysis that showed that KCC episodes could be identified from shifts in initiative. KSC-PaL is an innovative peer learning agent in that it attempts to shift initiative between itself and the student. Therefore, unlike other peer learning agents, it shifts roles from more experienced peer to less-experienced peer within a single problem-solving episode. Our evaluation of KSC-PaL found that students learned using the agent. Although there was no significant difference between the conditions, we found those students whose interactions with the agent had higher normalized initiative shifts, regardless of condition, learned more. We also found that this effect was more pronounced for students who began with a lower level of initial knowledge regarding linked lists. Additionally, in the experimental condition, KSC-PaL was successful in encouraging shifts in initiative using the identified shift encouraging cues and these attempts to shift initiative did not have a negative impact on student satisfaction with the agent.

Since in the current implementation of KSC-PaL, the agent chooses to shift initiative based on a fixed level of average initiative shifts, future work will explore varying the threshold for initiative shifts. There may be some ideal level of initiative shifts that encourages learning without decreasing student satisfaction with the agent. Additionally, we plan to incorporate more sophisticated natural language understanding technology into KSC-PaL. With improved NLU, the human interpreter could be removed from the system. This would allow the system to be deployed in classrooms or potentially on the Internet.

References

1. Birtz, M.W., Dixon, J., McLaughlin, T.F.: The effects of peer tutoring on mathematics performance: A recent review. *B. C. Journal of Special Education* 13(1), 17–33 (1989)
2. Brown, A.L., Palincsar, A.S.: Guided, cooperative learning and individual knowledge acquisition, pp. 226–307. Lawrence Erlbaum Associates, Hillsdale (1989)
3. Chan, T.-W., Baskin, A.B.: Studying with the prince. In: *Proceedings of the ITS-88 Conference*, pp. 194–200 (1988)
4. Chu-Carroll, J., Brown, M.K.: An evidential model for tracking initiative in collaborative dialogue interactions. *User Modeling and User-Adapted Interaction* 8(3–4), 215–253 (1998)

5. Cohen, P.A., Kulik, J.A., Kulik, C.C.: Education outcomes of tutoring: A meta-analysis of findings. *American Education Research Journal* 19(2), 237–248 (1982)
6. Conati, C., Gertner, A., van Lehn, K.: Using Bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction* 12(4), 371–417 (2002)
7. Fisher, E.: Distinctive features of pupil-pupil classroom talk and their relationship to learning: How discursive exploration might be encouraged. *Language and Education* 7, 239–257 (1993)
8. Hausmann, R.G.M., Chi, M.T.H., Roy, M.: Learning from collaborative problem solving: An analysis of three hypothesized mechanisms. In: Forbus, K.D., Gentner, D., Regier, T. (eds.) *26th Annual Conference of the Cognitive Science Society*, Mahwah, NJ, pp. 547–552 (2004)
9. Jordan, P.W., Hall, B., Ringenber, M.A., Cue, Y., Rosé, C.P.: Tools for authoring a dialogue agent that participates in learning studies. In: *Artificial Intelligence in Education, AIED 2007*, pp. 43–50 (2007)
10. Kersey, C., Di Eugenio, B., Jordan, P., Katz, S.: Ksc-pal: a peer learning agent that encourages students to take the initiative. In: *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 55–63. Association for Computational Linguistics (2009)
11. Rekrut, M.D.: *Teaching to learn: Cross-age tutoring to enhance strategy instruction*. American Education Research Association (1992)
12. Tin, T.B.: Does talking with peers help learning? the role of expertise and talk in convergent group discussion tasks. *Journal of English for Academic Purposes* 2(1), 53–66 (2003)
13. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: *NAACL 2003: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Morristown, NJ, USA, pp. 173–180. Association for Computational Linguistics (2003)
14. Vizcaíno, A.: A simulated student can improve collaborative learning. *International Journal of Artificial Intelligence in Education* 15(1), 3–40 (2005)
15. Walker, M., Whittaker, S.: Mixed initiative in dialogue: an investigation into discourse segmentation. In: *Proceedings of the 28th annual meeting on Association for Computational Linguistics*, Morristown, NJ, USA, pp. 70–78. Association for Computational Linguistics (1990)

Transforming a Linear Module into an Adaptive One: Tackling the Challenge

Jonathan G.K. Foss and Alexandra I. Cristea

Department of Computer Science, University of Warwick,
Coventry, CV4 7AL, United Kingdom
{J.G.K.Foss,A.I.Cristea}@warwick.ac.uk

Abstract. Every learner is fundamentally different. However, few courses are delivered in a way that is tailored to the specific needs of each student. Delivery systems for adaptive educational hypermedia have been extensively researched and found promising. Still, authoring of adaptive courses remains a challenge. In prior research, we have built an adaptive hypermedia authoring system, MOT3.0. The main focus was on enhancing the type of functionality that allows the non-technical author, to efficiently and effectively use such a tool. Here we show how teachers can start from existing course material and transform it into an adaptive course, catering for various learners. We also show how this apparent simplicity still allows for building of flexible and complex adaptation, and describe an evaluation with course authors.

Keywords: authoring of adaptive hypermedia, adaptive hypermedia, MOT3.0.

1 Introduction

Learners are individuals, and it is important to cater to their specific needs and requirements. Although this is a statement usually widely agreed upon, especially in the case of learner-centered teaching [1], we don't yet see a wide number of courses delivered in an adaptive fashion. Adaptive educational hypermedia has been around for almost 20 years, and adaptive delivery systems have been extensively researched. The bottle-neck remains in the domain of authoring for such systems, despite a recent body of consistent research [2]. Part of it is due to the (real or assumed) complexity of (using) such systems. Previously we have built and described an enhanced adaptive hypermedia authoring system MOT3.0 [3]. The main focus of this effort was on adding and extending the type of functionality that allows the 'lay person', the non-technical author, to efficiently use such a tool. In this paper we show how, in a realistic case, a teacher can start from any course she is already teaching, and transform it, in a number of steps, into an adaptive course, thus targeting various learners and moving away from the 'one-size-fits-all' approach. We then discuss how this apparent simplicity still permits for the building of flexible and complex adaptation, and finally present evaluation results with designers and authors of the tool.

2 Scenarios

This paper considers the authoring process from the point of view of two types of authoring, as illustrated by the two scenarios below.

2.1 Content Authoring

Professor Smith is a lecturer in Computer Science, and has presented a ‘Web Development’ course for the last five years. The resources she currently uses are: 30 lecture presentations (written in PowerPoint); 5 videos (each 5 minutes long) and 1 online quiz (authored in Moodle). Although the Professor is keen to embrace the advantages of adaptive hypermedia, she does not want to spend a long time rewriting all of her course material. Nor does she wish to learn a new programming language. Thus she uses the MOT3.0 tool, which will allow her to structure her existing content in a way that can be integrated into an adaptive course. Her students have previously taken an ILS (Index of Learning Styles [4]) test and have shown clear preferences for two types of learning styles: some of her students are *visual*, some *verbal*. She would also like to classify her students into *beginner*, *intermediate* and *advanced* groups. She then selects two adaptation strategies from a pool of strategies (created by her colleague, Professor Jones) that cater for the two types of adaptivity she is envisioning. From the natural language description of the strategies, without reading the code she finds out what type of labeling and annotation she needs to add to the material she has imported into MOT3.0. Because the content has been automatically separated into many reusable pieces, she finds the annotation process simple and fast. Finally, she applies the adaptation strategies to her content and deploys the result in the adaptation engine which will display it to her students, in a personalized way.

2.2 Adaptation Authoring

Professor Jones is another Computer Science lecturer, and a colleague of Professor Smith. He understands the pedagogical benefits of adaptive hypermedia, and has recently learnt the syntax of the LAG [5] adaptation programming language. Professor Jones has been appointed by his department to create a pool of adaptation strategies that will be used by his colleagues. He has both pedagogical knowledge and programming knowledge.

However, Professor Jones has not yet had much experience of authoring LAG adaptation files. The web-based PEAL editor [6] will assist Professor Jones, providing syntax highlighting and code completion. He then creates a good number of relevant strategies in a relatively short amount of time. Importantly, he adds good natural language descriptions to each of the strategies, so that his colleagues may use them without needing to read any of his code.

In the following sections, we will explain, from a technical point of view, how Professor Smith and Professor Jones can collaborate on an adaptive course, utilizing the two scenarios above.

3 Importing the Linear Content in MOT3.0

3.1 Importing Presentation Slides

Professor Smith starts by using MOT3.0's presentation *importer* to upload one of her existing PowerPoint files on "PHP" to the MOT server. The import script analyzes the presentation content, and creates a new domain structure (called a *domain map*) to store her lecture (see Fig. 1). As adaptation means conditionally displaying or removing content fragments, depending on the learner's needs, the first task for the system is to separate the existing content into reusable fragments (called *attributes*).

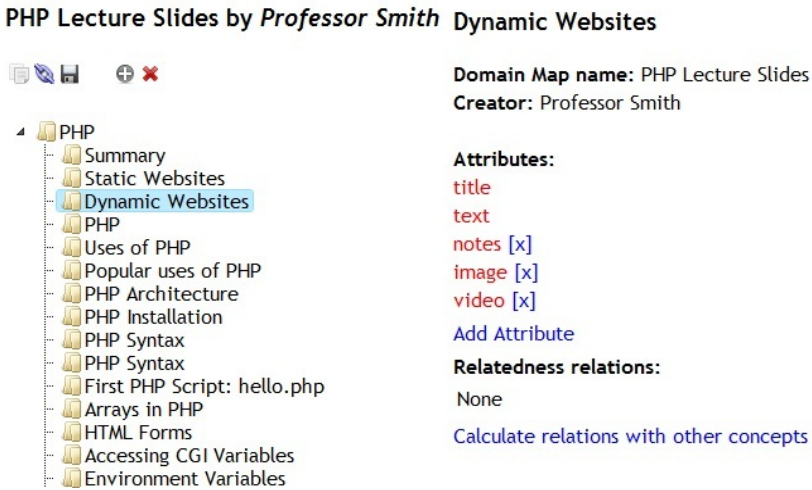


Fig. 1. Domain Model of a PowerPoint presentation on the "PHP" topic

Concretely, for each slide in her presentation, the import script creates a new concept in the domain map hierarchy (Fig. 1, left side), and a number of attributes assigned to this concept (Fig. 1, right side). The importer generates a slide *image*, and also automates OpenOffice.org¹ to export an HTML representation of the slide. From the latter, MOT3.0 extracts the *title* of the slide, the *text* content, and Professor Smith's slide *notes*. These attributes are the various information representations for each slide, and ensure thus various adaptations (e.g., slide notes can be used to create an overview; titles can be used to generate a 'Table of Contents'). The actual strategies she will be using are created by Professor Jones, and will be introduced in section 4. The extracted format allows Professor Smith also to add additional information to her module, either from HTML content stored previously on MOT3.0 - by simply copying a concept across from a previously authored domain map; or from additional material - e.g., she can upload one of the videos she was using in her class. She does this by creating another attribute for the concept 'Dynamic Websites', to which she uploads the *video* file (Fig. 1, right side, attribute 'video').

¹ <http://www.openoffice.org>

3.2 Importing Wikipedia Content

Professor Smith is keen to enhance her lectures by providing information about related topics from Wikipedia². She is aware of the issues surrounding the reliability of Wikipedia content; however she would like her students to be able to read about the module topics from other sources. She simply types the name of a Wikipedia article (here, “PHP”) into MOT3.0’s Wikipedia importer, which then downloads the WikiText source code of the article. Headings in WikiText are denoted by placing ‘=’ signs on both sides of the heading text. The number of signs denotes the level of the heading (e.g. 2 signs for a level 1 heading, 3 signs for a level 2 heading etc.), which allow the import script to divide the article’s content into sections, thus inferring the structure of the article. For each section of the article, a concept is created in the *domain map* (Fig. 2, left side). Each concept is assigned two attributes; the *title* of the section, and the *text* of the section (converted to HTML). The import clearly generates a good number of reusable, separate concepts, grouped in hierarchies, each with at least two attributes. All these will constitute the alternatives that will be available to the adaptation strategies she will apply. As with the previous domain map, Professor Smith is able to add more content to the newly created domain map.

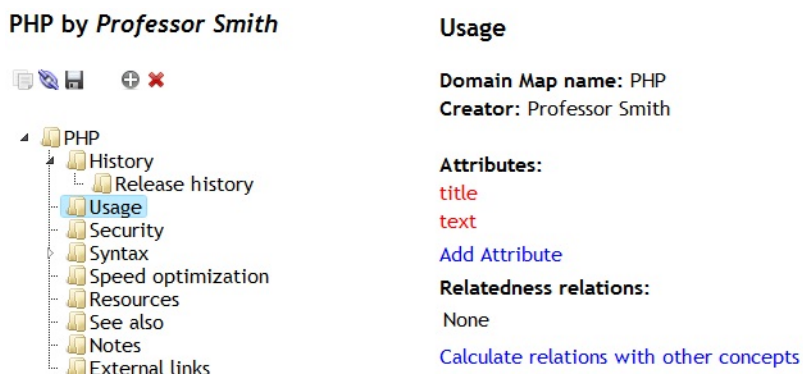


Fig. 2. Domain model of the imported Wikipedia PHP article

3.3 Importing Moodle Content

Another import script Professor Smith can use concerns content from other Learning Management Systems, such as Moodle or Sakai. Professor Smith has already created an online quiz using Moodle, so she exports this content to an IMS-QTI file. She can then upload the IMS-QTI file to MOT3.0, where her content will be converted into another domain map. For each question in the quiz, a concept is created. Each of these concepts contains a *question* attribute, and an *answer* attribute. These questions and answers can be used within an adaptive course. For instance, it would be simple to create an adaptive course that hides all answers until the user has read all questions.

² <http://www.wikipedia.org>

4 Creating Adaptation Strategies

Professor Jones uses PEAL to create a series of adaptation strategies.

4.1 Beginner-Intermediate-Advanced Strategy

One strategy he creates divides students into 3 groups: *beginner*, *intermediate* and *advanced*, hiding content from learners until they have reached the appropriate level (Fig. 3 shows an editing snapshot). LAG is able to update the user model, to allow the user to progress from *beginner* to *intermediate*. PEAL suggests *automatic completion* for the current program line (pop-up window). The available library *code fragments*, which can be inserted directly into the current code, appear in the right frame. Also, Fig. 3 shows *color and formatting coding* and *recognition of programming instructions*, as well as *code line numbers* to help the author to program in the LAG adaptation language, which is new for Professor Jones. Additionally, PEAL gives access to previously stored strategies (created by someone else and marked for sharing), allows parts of programs to be created directly via a Wizard, and thus overall represents a simple way for Professor Jones to accomplish his task in a short amount of time.

```

17 initialization (
18
19     PM.next = true
20     PM.ToDo = false
21     PM.
22     PM.GM
23     // PM.GM.Concept          general (unlabeled) concept readable; mark every
24     // PM.GM.Concept.show    id yet"
25     PM.menu
26     while TRUE (
27         PM.GM.Concept.show = true
28         UM.GM.Concept.beenthere = 0
29     )
30     UM.GM.begnum = 0
31     UM.GM.intnum = 0
32     UM.GM.advnum = 0
33     while GM.Concept.label == beg (
34         UM.GM.begnum += 1
35     )
36     while GM.Concept.label == int (
37         PM.GM.Concept.show = false
38

```

Fig. 3. Editing with the PEAL tool

4.2 Visual-Verbal Strategy

Another strategy Professor Jones creates differentiates between learners who are visual learners and those who prefer text. He defines a variable representing if the user prefers visual or verbal content. Pieces of content are labeled 'visverb', and given a weight to indicate whether the content is visual or verbal. The user's preference variable is compared with the weight of the content, and if the result is above a predefined threshold, the content is shown.

When Professor Jones has completed his strategies, he publishes them on the university website. He has added a comment to the top of each strategy that states the purpose of the strategy, and the labels the strategy uses.

5 Combining and Enhancing Linear Input

5.1 Adding Adaptive Behavior to Linear Content

After importing and enriching her imported material via domain maps, as shown in section 3, Professor Smith can now export them into a *goal map*, by clicking on an icon in MOT3.0. A goal map allows her to add pedagogical labels and weights (Fig. 4, right side), according to the adaptation strategy that she will be employing. She could import the domain maps to various goal maps and add different labels, thus using the same content for different pedagogical personalization strategies. However, she decides to create only one lesson for now, based on the content from one of her presentations. The goal model environment also allows her to combine content from different domain maps. She uses this to add information from her Wikipedia domain map. Then she labels the image version of the slide as ‘visverb’, and gives it a weight of 30 (representing visual content) and the text version of the slide as ‘visverb’ with a weight of 70 (for verbal content). These labels and weights correspond to the ones prescribed by the ‘Visual-Verbal’ strategy created by Professor Jones. The MOT3.0 system allows her to apply the same label and weight to many goal model concepts at once, thus saving her time, as most of her material is either of a visual or a verbal nature.

The screenshot displays the MOT3.0 goal model interface. On the left, a tree structure shows the imported content under the heading "PHP Lecture Slides by Professor Smith". The tree includes a root node "[PHP] (, 0)" with sub-nodes for "[Summary] (, 0)", "[PHP Syntax] (, 0)", and "[PHP Installation] (, 0)". Under "[PHP Syntax] (, 0)", there are sub-nodes for "title (, 0) (PHP Syntax)", "text (, 0) (PHP SyntaxThe PHP preprocessor)", "image (visverb, 30) ()", and "notes (, 0) ()". Under "[PHP Installation] (, 0)", there are sub-nodes for "title (, 0) (PHP Installation)", "text (visverb, 70) (PHP InstallationBinaries and s)", "image (visverb, 30) ()", and "notes (, 0) ()". On the right, a configuration panel titled "Multiple sublessons selected:" shows a "Label:" field with the value "visverb" and a "Weight:" field with the value "70". Below these fields is an "Update" button.

Fig. 4. Goal model of the imported PowerPoint presentation

5.2 Delivering Adaptive Courses

Professor Smith can now export her goal map and upload it to the AHA! delivery tool, together with one of Professor Jones’s strategy files, and deploy it. This will create a course combining the educational content with the adaptation strategy. Professor Smith’s students can then visit the adaptive course.

6 Evaluation and Discussion

An evaluation was performed at the University of Warwick with six volunteer course authors and designers. They were asked to explore the system, and answer 45 questions, which we grouped into 10 categories of basic functions, as below:

1. ... *browsing other author's materials*
2. ... *editing with MOT3.0*
3. ... *changing hierarchies of material via drag&drop*
4. ... *copying and linking functionality*
5. ... *editing HTML using the editor*
6. ... *importing Wikipedia content*
7. ... *importing Presentation content*
8. ... *functionality of importing content*
9. ... *authoring for adaptation as supported by MOT3.0*
10. ... *Semi-Automatically Creating and Linking Content for adaptation*

Fig. 5 shows that the designers found most of the basic functions 'Easy' (or 'Very Easy') to use.

To establish the statistical significance of these results, we have mapped the answers {'Very Easy', 'Easy', 'Difficult', 'Very Difficult'} onto the values {2, 1, -1, -2}. This assumes equidistance between these labeled values, as well as monotonicity, an assumption which is widely used in literature, and also conforms to the natural language use of these words. We have then applied a one-sample T-test to compare the answers against the average of 0, corresponding to 'Neither Easy nor Difficult', to establish if the positive average is statistically significant.

An analysis of the data showed that *browsing, editing, changing hierarchies* (Q1,2,3), and *editing HTML* (Q5) are statistically significantly easy with 95% confidence ($P < 0.05$). Also *importing Wikipedia content, presentation, and (semi-)automatically creating content and linking* are significantly useful.

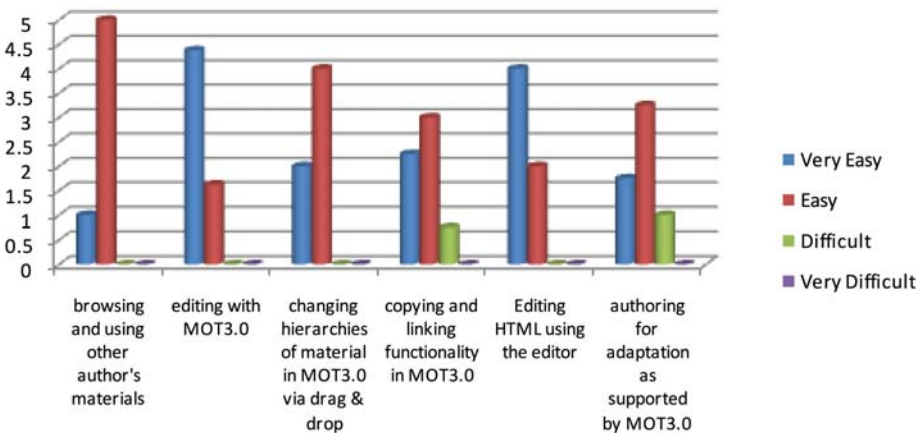


Fig. 5. Evaluation results for the basic functionality of MOT3.0

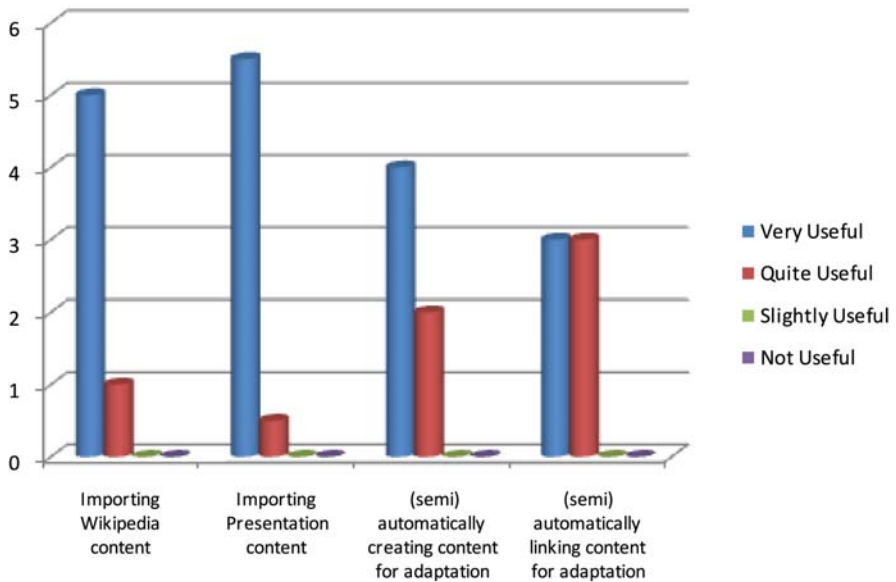


Fig. 6. Evaluation results for the importing features

However, the *copying and linking functionality* (Q4), *importing content* (Q8) and *general authoring* (Q9) are only statistically significant with 90% confidence. To analyze the reasons for this, Table 1 shows the p-values for each of the sub-questions within these questions.

Table 1 shows that within the area of copying and linking functionality, whilst *dragging domain concepts* is significantly easy, *inserting goal maps from domain concept attributes* or *other sublessons* is not. Looking at the qualitative comments, the experts noted that: “Inserting of domain map attributes needs improvement [...] partial goalmaps cannot be inserted” and “it is easy, but a bit inconsistent: for GM you have to click add, for DM you have to drag & drop. I would like it not to refresh back, as I may want to add more than 1 attribute”.

With regard to the importing scripts, Table 1 shows that although the functionality was appreciated by the experts, the speed of the scripts were unsatisfactory. Some of the comments were “It will be good to see how long the article/presentation is before the import.”, and “The speed could become an issue if several presentations are imported simultaneously.”

The general authoring questions showed that the experts felt that *creating adaptive presentations with MOT3.0* is preferred (in a statistically significant way) to programming adaptation from scratch, and also *using graphical drag & drop interfaces* in authoring for adaptation is considered beneficial. Looking at why the experts are not convinced about *(semi-)automatically creating and linking content*, the comments were as follows: “The physical manipulation is easy, but you have to understand what you are doing”, “Linking automatically is only possible in a hierarchical way. It would be interesting to see different types of automatic linking.”

Table 1. Sub-questions for questions 4, 8 and 9

<i>Question</i>		<i>p-value</i>
4a	Dragging domain concepts between trees when copying/linking	0.001
4b	Inserting goal map sublessons from domain concept attributes	0.363
4c	Inserting other goal map lessons as sublessons	0.093
8a	The content of the imported Wikipedia article	0.001
8b	The number of attributes extracted from an article	0.001
8c	The type of attributes extracted from an article	0.001
8d	Speed of importing an article	0.465
8e	The content of the imported Presentation	0.001
8f	The number of attributes extracted from a Presentation	0.001
8g	The type of attributes extracted from a Presentation	0.001
8h	Speed of importing a Presentation	0.465
9a	Being able to create adaptive presentations with MOT3.0 (as compared with programming adaptation from scratch)	0.001
9b	Being able to (semi) automatically create content for adaptation	0.286
9c	Being able to (semi) automatically link content for adaptation	0.363
9d	Using graphical drag & drop interfaces in authoring for adaptation	0.001

Thus, whilst clearly some improvements can be done (and the experts have given us some very good pointers towards this), the overall evaluation shows that people like our imaginary Professors Smith and Jones can expect to be able to author with a reasonable degree of ease personalized courseware with a system such as MOT3.0.

7 Conclusions

Most research into adaptive hypermedia has focused on the delivery of the content rather than the authoring side. Interbook [8] is an example of a system which uses a more familiar authoring interface, and allows authors to create content based on Microsoft Word documents. Still, such documents entail annotation to create adaptivity rules. AHA! [7] also provides a set of authoring tools. However, it requires the author to manually create concepts in (X)HTML.

This paper has documented and evaluated the process that will allow educators to create adaptive courses from some of their existing resources. Specifically, we have introduced methods of generating domain models based on presentation slides and Wikipedia articles. It is hoped that authoring systems with import facilities such as those provided by MOT3.0 will encourage more educators – from a wide variety of subject areas – to author for adaptive hypermedia.

References

1. Nunan, D.: *The learner-centred curriculum: a study in second language teaching*. Cambridge University Press, Cambridge (1988)
2. Brusilovsky, P.: Developing adaptive educational hypermedia systems: From design models to authoring tools. In: Murray, T., et al. (eds.) *Authoring Tools for Advanced Technology Learning Environment*, pp. 377–409. Kluwer, Dordrecht (2003)
3. Foss, J., Cristea, A.: Adaptive Hypermedia Content Authoring using MOT3.0. In: *EC-TEL* (2009)
4. Soloman, B., Felder, R.: Index of Learning Styles Questionnaire. In: College of Engineering, North Carolina State University, <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>
5. Cristea, A., de Mooij, A.: Adaptive Course Authoring: My Online Teacher. In: *International Conference on Telecommunications 2003*, Papeete, French Polynesia, pp. 1762–1769 (2003)
6. Cristea, A., Smits, D., Bevan, J., Hendrix, M.: LAG 2.0: Refining a reusable Adaptation Language and Improving on its Authoring. In: Cress, U., Dimitrova, V., Specht, M. (eds.) *EC-TEL 2009*. LNCS, vol. 5794, pp. 7–21. Springer, Heidelberg (2009)
7. De Bra, P., Smits, D., Stash, N.: Creating and Delivering Adaptive Courses with AHA! In: Nejdil, W., Tochtermann, K. (eds.) *EC-TEL 2006*. LNCS, vol. 4227, pp. 22–33. Springer, Heidelberg (2006)
8. Eklund, J., Brusilovsky, P.: Interbook: An Adaptive Tutoring System. *UniServe Science News* 12, 8–13 (1999)

An Authoring Tool to Support the Design and Use of Theory-Based Collaborative Learning Activities

Seiji Isotani¹, Riichiro Mizoguchi², Sadao Isotani³, Olimpio M. Capeli⁴,
Naoko Isotani⁴, and Antonio R.P.L. de Albuquerque⁵

¹ Human-Computer Interaction Institute, Carnegie Mellon University, USA

² The Institute of Scientific and Industrial Research, Osaka University, Japan

³ Physics Institute, University of Sao Paulo, Brazil

⁴ Foundation Osasco Institute of Technology, Brazil

⁵ Paulista University, Brazil

isotani@acm.org, miz@ei.sanken.osaka-u.ac.jp,

sisotani@if.usp.br

Abstract. Design of pedagogically sound collaborative learning (CL) activities is a complex task, but necessary if the goal is to support learning. Through the design of CL scenarios, a designer can define structures that increase the chance for learning to occur. It means that the effectiveness of the collaboration depends on the transformation of the designer's intentions into elements that will constitute the learning scenario. To support the creation of CL scenarios this paper presents an intelligent authoring tool that is equipped with the knowledge about different pedagogies and practices related to collaboration. Through the use of this information, the tool can provide intelligent guidance that support designers to create more effective CL scenarios. The results of an experiment suggest that our tool helps teachers to more easily introduce CL activities in classroom and creates favorable conditions for students to perform collaboration improving their overall learning performance throughout the year.

Keywords: Collaborative learning, intelligent authoring tool, ontology.

1 Introduction

To design an effective and pedagogically sound collaborative learning (CL) scenario, a teacher can rely on learning theories (such as Peer Tutoring) to assist with the assignment of roles, selection of activities, definition of learning strategies, formation of groups, and so on [7]. However, to select an appropriate learning theory (or any other type of structure to help collaboration) it is necessary to consider learners' conditions (e.g., previous knowledge/skills), the learning goals, and other variables that may influence and promote good interactions among group members [3].

This flexibility in the choice of different learning theories, and in the consideration of different variables, can therefore provide us with a wide range of options for designing and conducting CL processes. It also suggests the difficulty of selecting an appropriate set of learning theories during the instructional design process, to ensure learners' benefits and the consistency of the learning processes. Creating

well-thought-out CL scenarios requires experience and knowledge about different pedagogies and practices related to collaboration. Inexperienced designers/teachers, who may not have all the necessary knowledge to formulate pedagogically sound collaborative learning plans, may have difficulties in designing CL activities.

Thus, to help users design CL scenarios based on theories, we need an elaborate authoring system. While the number of technologies that support collaboration have increased considerably [13], only a few CL authoring systems have been developed to deal with multiple theories [8]. As pointed out by Laurillard [10], the technology itself does not support good CL environments. Creating or using mechanisms for collaboration without pedagogical considerations does not necessarily improve learning outcomes and may possibly harm learners' development.

In this paper we will briefly present previous works related to CL design. Then, we will give an overview of our intelligent authoring tool, referred to as CHOCOLATO. And finally, we will present the results of an experiment carried out by one teacher and 133 students during the period of 2008 and 2009 showing that our tool helps the teacher to more easily design theory-based CL activities that can be applied in classroom and creates good conditions for students to interact more effectively.

2 Related Work

New technologies have been developed to explore the use of pedagogies in order to create better learning scenarios. Nowadays, there are many authoring tools that focus on supporting the design of learning scenarios [5]. However, there are few authoring tools that have been created specifically to support the designing of complex CL scenarios, such as LAMS [2], Cool Modes [11], Collage [6], and Learning Design Palette [7]. These tools help teachers to create structured CL activities for students by enabling the designing of the interaction flows.

One of the reasons for the limited number of tools that are "aware" of pedagogies (e.g., learning theories) is due to the difficulty of representing pedagogical knowledge and principles in a computer-understandable way. Because of that, none of the cited authoring systems has the desired functionality to retrieve appropriate learning theories for selecting methodologies that "match" a specific situation (e.g. students' current knowledge state) automatically, or to provide pedagogical principles, based on multiple theories, for structuring collaborative learning environments.

3 CHOCOLATO: A Theory-Aware Authoring Tool for CSCL

To address the problems presented in the previous section we have been developing a theory-aware authoring tool for CL called *CHOCOLATO – a Concrete and Helpful Ontology-aware Collaborative Learning Authoring Tool*. To allow for the application of multiple theories during the authoring process, CHOCOLATO uses the results of previous achievement in representing learning theories and collaboration formally through the use of ontologies [8]. Some of the benefits of this approach are: (a) to prevent unexpected interpretations of the theories while designing CL scenarios; (b) to provide common vocabulary to describe these scenarios; and (c) to offers enough information for computational semantics to allow for "intelligent" guidance with theoretical justifications during the authoring process.

CHOCOLATO is composed of different sub-systems that aim at supporting the design of CL scenarios based on learners' conditions, desires, and requirements. Some of the sub-systems used in this work provide help for group formation, the design of CL activities and the selection of learning materials.

Currently, CHOCOLATO utilizes technologies for the Semantic Web to reason on ontologies and support intelligent guidance during the authoring process. The ontology utilized by CHOCOLATO is described in previous work and is known as **CL ontology** [8]. To reason on the CL ontology, we use a set of procedures to deal with RDF referred to as ARC2¹. The MySQL database has been utilized to store the data of learners, ontologies, learning resources, and other information. The queries to retrieve information follow the SPARQL² format. The user interface utilizes an extension of an open source learning management system referred to as Claroline³. And finally, to develop the functionalities of CHOCOLATO we have been using standards languages for programming on the Web such as HTML, AJAX, and PHP.

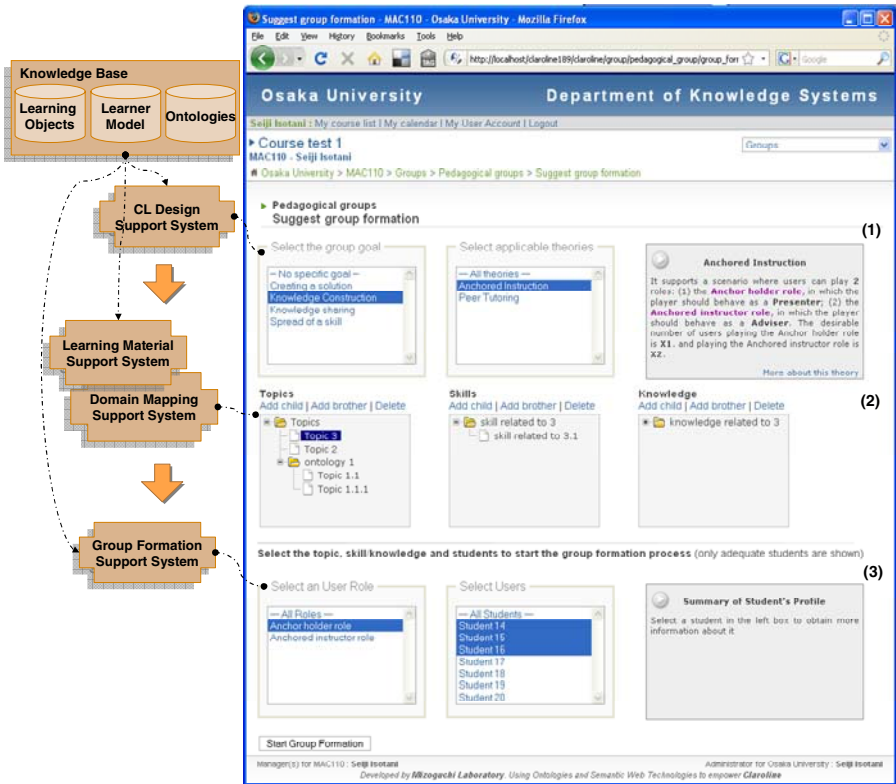


Fig. 1. The right side is a screenshot of the main interface of CHOCOLATO. The left side includes the elements in the CHOCOLATO architecture that run on the background.

¹ <http://arc.semsol.org/>
² <http://www.w3.org/TR/rdf-sparql-query/>
³ <http://www.claroline.net/>

One of the interfaces of CHOCOLATO is shown in Figure 1. Each block in CHOCOLATO’s interface, numbered from (1) to (3), has three boxes of dynamic information. The first block provides support for the selection of goals and theories to design CL scenarios; the second block helps to connect the domain content into our ontologies; and, the third block facilitates the group formation process.

The first block of the interface is connected with the CL design support system which enables users to select appropriate theories to design theory-based CL scenarios. Within this block, the first box (labeled as “Select the group goal”) is connected with the group goal concept in the CL ontology. Thus, this box shows the possible group goals that are currently represented in our ontology (e.g. knowledge construction or spread of a skill). If the ontology is updated, then automatically the system updates the list of group goals presented in this box.

The second box (labeled as “Select applicable theories”) shows the theories that users can utilize to create CL scenarios. As same as the first box, this information is extracted from the CL ontology. Furthermore, when the user selects a goal in the first box the system will only show the theories that can support the development of the chosen goal in the second box. This functionality is implemented by using the semantic connection between concepts in the ontology as shown in Figure 2.

The user’s selection in the first box (Figure 2-1) generates a constraint that needs to be satisfied. To do that, our algorithm works as follows:

1. Identify the concept in the ontology which corresponds to the user’s selection (Figure 2-2);

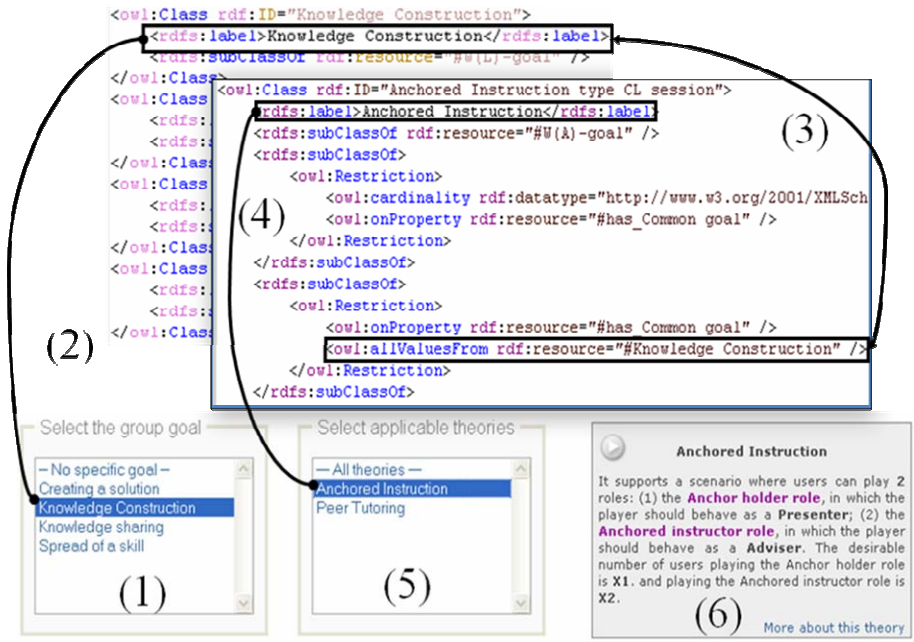


Fig. 2. Elements in the interface and their connections with concepts in the ontology (top)

2. Search values of the CL scenario concept verifying if the property “common goal” has the same value as the concept identified in step 1 (Figure 2-3);
3. Return the name of the theory utilized by the CL scenario that satisfies the constraint presented in step 2 (Figure 2-4).

With the theories presented in the second box (Figure 2-5) the user can select which theory he/she wants to use to create theory-based CL scenarios. However, this choice is not easy, especially for inexperienced teachers who may not have the knowledge of learning theories or collaborative learning. Thus, when a user selects a theory in the second box, the third box (Figure 2-6) gives a summary about how to apply the theory and some recommendations such as the number of roles that should be attributed to the students, the behavior that each student should externalize, the desired number of participants in each paper, besides other various information that can be visualized when clicking in the link “more about this theory.” It is worth to note that all information is obtained from our ontologies.

The second block of the interface (Figure 1-2) facilitates the mapping of domain specific content into the ontology. This process follows the idea of creating a decomposition tree to connect ontologies and domain content [9]. The first box corresponds to the topics (or domain-dependent learning goals). Each topic can be decomposed into sub-topics. For each topic the user needs to separate the skills to be developed from the knowledge to be acquired. Furthermore, each knowledge/skill can be connected with some learning resources to be utilized. Such a simple interface to create trees hides the complexity of mapping content into ontologies and facilitates the use of our system in real situations.

Finally, the third block of the interface supports the group formation process. In our tool, group formation is part of the CL design process. Thus, in our interface, when the user selects the group goals and a theory, then the information in the box labeled as “Select a user role” (bottom-left of Figure 1) is automatically updated presenting only the roles that the selected theory utilizes.

Furthermore, by clicking in one of the roles, the second box labeled as “Select users” (bottom-center of Figure 1) shows the learners registered in the environment who are able to *adequately* play the selected role. To verify adequacy, CHOCOLATO checks in the ontology the necessary and desired requirements to play a role. And then, by using the stages of knowledge/skills in the learner model it identifies the learners who are able to play the selected role in the domain specified previously.

Finally, after these steps are completed all necessary information to create a theory-based CL scenario is set up. Thus, the system will run in the background an algorithm to recommend interaction patterns that aid the user to create effective CL activities. To provide better user experiences while using our tool, the complexity of ontologies and reasoning processes are completely hidden from the user. Thus, through a simple interface, CHOCOLATO offers a more effective, intelligent, and structured guidance that helps users during the designing of CL scenarios.

4 Experiment

The experiment of using CHOCOLATO was accomplished at a public school referred to as FITO – Osasco Institute of Technology. It maintains a quite traditional fundamental course in the city of Osasco, Brazil. Due to its conservative philosophy, the

teaching method is strongly based on the traditional model of instruction where the teacher transmits his/her knowledge to students who passively *absorb* the content.

Through a partnership with the teacher of mathematics the author introduced in the beginning of 2008 the use of the collaborative learning in four classrooms of 5th grade (133 students). Because the school did not allow different methods of teaching/learning for each classroom, the use of control groups (classes using the traditional teaching) and experimental groups (classes using collaborative learning) was not utilized. Thus, to establish a basis for comparison between the traditional method of instruction and the method using collaborative learning, we analyzed the data of students who attended the 5th grade at FITO from 2000 to 2007. This data is composed by assessments made by the teacher to evaluate her students such as tests, homework, extra activities, grades for behavior in class, and etc.

We compare the result of the analysis obtained from 2000 to 2007 with the results obtained in the year of 2008 and 2009 when collaborative learning was introduced in the classroom supported by CHOCOLATO. Such comparison is feasible, because 2000 to 2009 all classes have similar average score performance in the beginning of the school year and the teaching method, tests, and other activities were accomplished by the same teacher, with the same learning materials during the period of 2000 to 2007. In the next section the method for analysis and its results are presented.

4.1 Data Analysis Method: Principal Components Analysis (PCA)

The objective of this data analysis is to offer a rough idea about what students knew in the beginning of the year and compare it with what they have learned by the end of the year. To accomplish that, initially all scores obtained by students who completed the 5th grade were gathered during the period of 2000 to 2007. Then, for each student we compared the score obtained in the first test of the year with the average score in the same year to create a graph. Because of the strong similarity between the graphs for each year, we show only three of them in Figure 3. Each point in the graph represents a student and his/her scores; the x -axis represents the score in the first test and the y -axis represents the average score of all activities in the same year.

All the graphs from 2000 to 2007 seem to follow a pattern that forms a cloud of points with an angle of 45 degrees in relation to x -axis. Such a pattern gives an indication that there might be a correlation between the score obtained in the first test and student development throughout the year.

To analyze the possible correlation among the students' scores, a method for identifying patterns in the data was utilized. This method is referred to as PCA – Principal Component Analysis Method [1]. It expresses the data in a way that similarities and differences are more perceptible. According to Smith [12], PCA not only helps to find patterns in data, but also shows the pattern by reducing the number of variables without much loss of information. This means that PCA aims at mapping possible correlated variables into another smaller number of variables that are referred to as principal components. In the case of a two-dimensional analysis, which is our case, the 1st principal component (PCA-1) shows the projection of the data into a straight line that follows the greatest variance of the points. In other words PCA-1 is a line that passes through the middle of the cloud of points in our data. The 2nd principal component (PCA-2) gives us the variance of the data in relation to the 1st principal component. It shows the pattern of the points that do not follow the PCA-1.

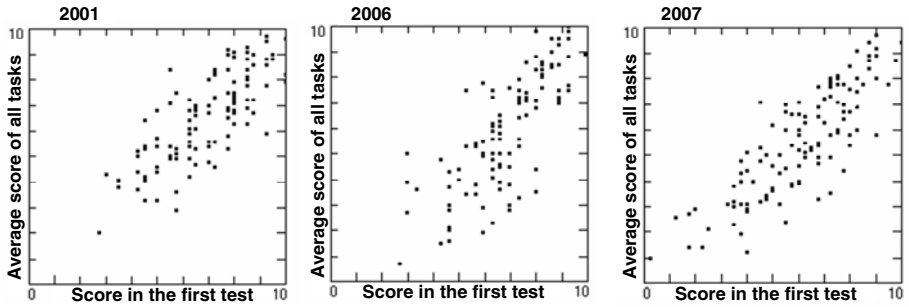


Fig. 3. Graphs showing the correlation between students’ score in the beginning of the year and the average score obtained throughout the year

We use the PCA to analyze students’ scores during the period of 2000 to 2007, where the teacher used the traditional teaching method (basically, individual learning). Figure 4 shows the result of the application of the PCA method in our data. As same as Figure 3, the x -axis represents the score in the first test and the y -axis represents the average score of all tasks. The points marked with “+” are the students’ data, the bold points are the projection of students’ data into the principal components, and the full line shows the linear regression (expected value of each point) of the data. The most important information is obtained from the bold points that represent the principal components. The correlation between the initial and the average scores is shown by the PCA-1 which if composed by the bold points following similar path to the linear regression line. The PCA-2 is shown by the bold points crossing *almost* perpendicularly the PCA-1.

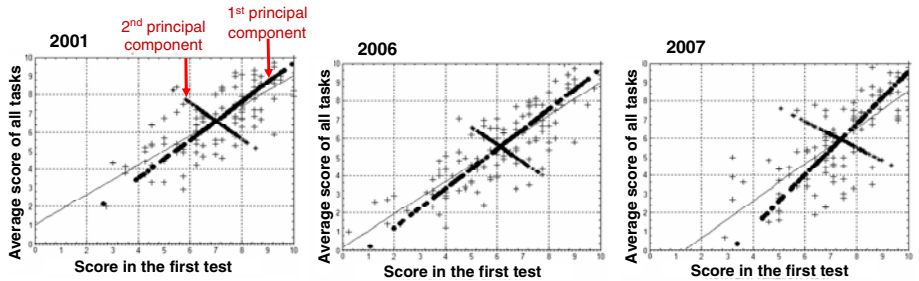


Fig. 4. Result of the application of the PCA method in the students’ data to check the correlation between the initial and average scores

The PCA-1 is a line that follows almost the same inclination of the linear regression (approximately 45 degrees in relation to x -axis) which indicates that there is a linear correlation between the first score and the average score. The PCA-2 indicates that the first score of students is not equal, but proportional to their average score in the year. This result suggests a pattern where *students with low scores in the beginning of the semester will have low scores in the end of the semester in comparison to*

his/her peers. In other words, we could say that a student acquires the content proportionally to his/her knowledge in the beginning of the course.

The ideal graph would be when the linear regression line and the PCA-1 have their initial values starting from six in the y -axis and for any point $P = (x, y)$, the value of y should always be equal or higher than the value of x . Furthermore, the PCA-2, that shows the variance of the data, should increase its size. Such a situation means that students who get a low score in the beginning of the semester has chances to obtain a score equal or higher than six in the end of the year, which is the minimum requirement to pass the final exam. Furthermore, students who get a good score in the beginning will continue getting good scores until the end of the year.

4.2 Introducing Collaborative Learning with CHOCOLATO

The 5th grade math teacher at FITO was willing to test new methods of teaching-learning because of two problems she had to deal with everyday. For example, one problem that the teacher had been dealing with was to teach the same content for all students at the same time. It means that less knowledgeable students and more knowledgeable students are treated in a homogeneous way. In such a situation, the teacher cannot interrupt her explanations to aid those who have difficulties in following the content. Furthermore, according to Freire [4] while teachers are exposing the content, usually, less knowledgeable students feel uncomfortable to ask questions; and more knowledgeable students who already understand the content have to wait and follow a slow pace.

The teacher's observation is complete in line with our findings during the data analysis using PCA. Because she could not give adequate support for students using the traditional teaching method, what happened is that *students with low scores in the beginning of the semester had few chances of improving his/her performance along the course*.

Thus, to provide the teacher with new pedagogical methods to support learning, in the beginning of 2008 CHOCOLATO was introduced as a mean to utilize collaborative learning in the classroom. The experiment with CHOCOLATO was carried out until the end of the school year (about 9 months) and then again in 2009. The initial goal of this experiment was to improve the quality of teaching and thereby improve the performance of students.

To create CL scenarios that could be applied in classrooms, the teachers had several questions. Among these, the most important questions were the following: (1) how should I group students? (2) How should I plan the group activities? (3) How can I support students while they are working in groups? (4) How can I assess students' learning?

The current version of CHOCOLATO could help the teacher to answer the first two questions. With structured guidance using CHOCOLATO's interface, the teacher did not have difficulties to set up a goal for groups and choose a theory to work with. The teacher did not have any knowledge about learning theories. Therefore, the information that CHOCOLATO offered was fundamental to help the teacher to identify a good theory-based CL scenario for specific situations. Regarding group formation, the system could suggest the learners who were able to play a role satisfactorily and automatically form groups. Finally, CHOCOLATO also provides interaction patterns

that help the teacher to design CL activities following pedagogical requirements. The question related to how to support students while they are working in groups and how to evaluate them cannot be answered with the current version of CHOCOLATO. The reason is because the system has been developed to be an authoring tool that facilitates the creation of theory-based CL scenarios. However, in future versions of CHOCOLATO an extension that deals with teacher support for conducting and analyzing CL activities is under development.

In spite of some difficulties faced by the teacher to carry out CL activities in classroom, the general results were positive. The main gain according to the teacher was to open a pedagogical procedure that at the same time allows for helping less knowledgeable students and more knowledgeable ones. Furthermore, it was observed that CL activities had a positive effect in students' learning and behavior. The teacher pointed out that there was a considerable decrease of unnecessary noise (parallel conversations not related to the content) during the classes because students were engaged while doing their group work. Finally, after the end of the school year we analyzed the results using the PCA method as the same as we have done in previous years. Thus, we compare the score obtained by the student on the first test with the average score obtained in 2008. Figure 5 shows the PCA after the introduction of CL using CHOCOLATO in 2008. Although, many difficulties have occurred while students worked in groups, the inclination of PCA-1 in relation to x -axis is much smaller if compared with the other graphs shown in Figure 4. Furthermore the linear regression line starts from the value four in y -axis and the PCA-2 is much larger than the previous graphs as well. It signifies that students who had a poor performance in the first test could recover and learn better using theory-based CL supported by CHOCOLATO if compared with the previous seven years where the traditional method was utilized. A similar result was obtained during the year of 2009.

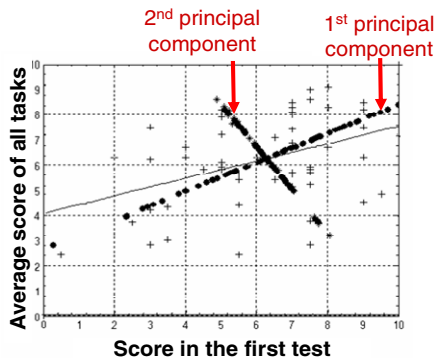


Fig. 5. Result of the application of the PCA method for 2008. The inclination of the PCA-1 is smaller than the previous years which indicate a better performance of students.

In summary, the results of the experiment suggest that with a principled CL design process, CHOCOLATO creates favorable conditions for learners to perform collaboration, while helping instructors to more easily create a sequence of activities that support the achievement of learning goals and thereby improving the performance of students throughout the year.

5 Conclusions

Developing, deploying and evaluating a theory-aware tool for collaborative learning has been especially challenging given the context of group learning where the synergy among the learners' interactions affect the learning processes, and hence, the learning outcome. We believe that the well-thought-out development of pedagogically compliant authoring tools is a step toward bridging the gap between the theoretical understanding of collaborative learning and the practical designing of effective CL activities.

The tool CHOCOLATO presented in this work is an example that shows some interesting results of using ontologies to create more intelligent systems with theoretical knowledge to support teachers during the design of CL activities. The experiments conducted with a math teacher and 133 students in a public school in Brazil for 2 years suggest that CHOCOLATO helped the designing of theory-based CL scenarios that support the achievement of desired learning goals. Furthermore, the implementation of these CL scenarios created favorable conditions for students to perform meaningful interactions and thereby improving their learning performance.

It is worth to note that the current version of CHOCOLATO does not have the capability to update automatically the learner model. This means that, for each set of CL activities, the teacher had to evaluate the students and semi-manually updated the changes of the learners' states in our tool. The automatic update of the learner model is a complex process and we are developing such functionality to be included in future versions of our tool.

References

1. Cooley, W.W., Lohnes, P.R.: *Multivariate Data Analysis*. Wiley, New York (1971)
2. Dalziel, J.: *Implementing Learning Design: the Learning Activity Management System (LAMS)*. In: 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education, pp. 593–596 (2003)
3. Dillenbourg, P., Hong, F.: *The Mechanics of CSCL Macro Scripts*. *International Journal of Computer-Supported Collaborative Learning* 3(1), 5–23 (2008)
4. Freire, P.: *Pedagogy of the Oppressed*. Continuum, New York (1993)
5. Griffiths, D., Blat, J., Garcia, R., Vogten, H., Kwong, K.: *Learning Design Tools*. In: Koper, R., Tattersall, C. (eds.) *Handbook on Learning Design: modelling and implementing network-based education & training*, pp. 109–135. Springer, Heidelberg (2005)
6. Hernandez-Leo, D., Villasclaras-Fernandez, E.D., Jorriñ-Abellan, I.M., Asensio-Perez, J.I., Dimitriadis, Y., Ruiz-Requies, I., Rubia-Avi, B.: *Collage: a Collaborative Learning Design Editor Based on Patterns*. *Educational Technology & Society* 9(1), 58–71 (2006)
7. Inaba, A., Mizoguchi, R.: *Learning Design Palette: An Ontology-Aware Authoring System for learning design*. In: 12th International Conference on Computers in Education, pp. 597–607 (2004)
8. Isotani, S., Inaba, A., Ikeda, M., Mizoguchi, R.: *An Ontology Engineering Approach to the Realization of Theory-Driven Group Formation*. *International Journal of Computer-Supported Collaborative Learning* 4(4), 445–478 (2009)

9. Isotani, S., Mizoguchi, R.: Adventure in the Boundary between Domain-Independent Ontologies and Domain-Specific Content for CSCL. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part III. LNCS (LNAI), vol. 5179, pp. 523–532. Springer, Heidelberg (2008)
10. Laurillard, D.: The Pedagogical Challenges to Collaborative Technologies. *International Journal of Computer-Supported Collaborative Learning* 4(1), 5–20 (2009)
11. Pinkwart, N.: A Plug-In Architecture for Graph Based Collaborative Modeling Systems. In: 11th International Conference on Artificial Intelligence in Education, pp. 535–536 (2003)
12. Smith, L.I.: A Tutorial on Principal Components Analysis (2002), http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
13. Soller, A., Martínez-Monés, A., Jermann, P., Muehlenbrock, M.: From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. *International Journal of Artificial Intelligence in Education* 15(4), 261–290 (2005)

How to Build Bridges between Intelligent Tutoring System Subfields of Research

Philip Pavlik Jr. and Joe Toth

Human Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA 15213
ppavlik@andrew.cmu.edu, jtoth@cs.cmu.edu

Abstract. The plethora of different subfields in intelligent tutoring systems (ITS) are often difficult to integrate theoretically when analyzing how to design an intelligent tutor. Important principles of design are claimed by many subfields, including but not limited to: design, human-computer interaction, perceptual psychology, cognitive psychology, affective and motivation psychology, statistics, artificial intelligence, cognitive neuroscience, constructivist and situated cognition theories. Because these theories and methods sometimes address the same grain size and sometimes different grain sizes they may or may not conflict or be compatible and this has implications for ITS design. These issues of theoretical synthesis also have implications for the experimentation that is used by our various subfields to establish principles. Because our proposal allows the combination of multiple perspectives, it becomes apparent that the current “forward selection” method of theoretical progress might be limited. An alternative “backward elimination” experimental method is explained. Finally, we provide examples to illustrate how to build the bridges we propose.

Keywords: Intelligent tutoring systems, situated cognition, cognitive psychology, design, perception.

1 Introduction

A very old story tells of 6 blind men who were asked to describe an elephant after feeling a part of it. One described it as a wicker basket (ear), one as a ploughshare (tusk), one as a plough (trunk), one as a granary (body), one as a pillar (leg) and one as a mortar (back) and one as a brush (tip of the tail). An ITS researcher trying to be faithful to basic theories of how people develop their capacity to think and act often feels like he is reading research from these blind men describing an elephant. The ITS researcher turns to the fields of artificial intelligence, statistics, cognitive psychology, cognitive neuroscience, perceptual psychology, all fields of design, human computer interaction, computational modeling, constructivist learning theories, behaviorism, embodied cognition and situated cognition, to name several, and finds many alternative ways to characterize the development in proficiency that we commonly call learning. Unfortunately, these many different sources often result in more confusion than explanation because these multiple theories are difficult to integrate into a mental model of learning that is consistent. It seems the only alternative for the ITS researcher is to have

several different inconsistent mental models of learning that must compete when it comes time to plan tutorial interventions to improve student learning.

2 The Problem

We are not the first to notice this problem with sciences of behavior as we know them. Donald Davidson in his influential paper “On the Very Idea of a Conceptual Scheme” [1] has addressed this problem by laying the blame at the feet of philosophers such as Thomas Kuhn. Kuhn has made the case that sciences progress by large paradigm shifts in which new ideas and worldviews replace old conceptual schemes [2], which are subsequently unintelligible from the perspective of the new paradigm. In criticizing the absurdity of this, Davidson says:

Suppose that in my office of Minister of Scientific Language I want the new man to stop using words that refer, say, to emotions, feelings, thoughts and intentions, and to talk instead of the physiological states and happenings that are assumed to be more or less identical with the mental riff and raff. How do I tell whether my advice has been heeded if the new man speaks a new language? For all I know, the shiny new phrases, though stolen from the old language in which they refer to physiological stirrings, may in his mouth play the role of the messy old mental concepts. The key phrase is: for all I know. What is clear is that retention of some or all of the old vocabulary in itself provides no basis for judging the new scheme to be the same as, or different from, the old. So what sounded at first like a thrilling discovery – that truth is relative to a conceptual scheme – has not so far been shown to be anything more than the pedestrian and familiar fact that the truth of a sentence is relative to (among other things) the language to which it belongs. Instead of living in different worlds, Kuhn's scientists may, like those who need Webster's dictionary, be only words apart. [1, p. 10-11]

Davidson's goes on to argue that the strong dualism of “scheme and content, of organizing system and something waiting to be organized”, implied in Kuhn's theory of how ideas shift, is unintelligible. He makes the case that there is no duality of theory neutral reality and relative theory, but rather he argues sensory experience provide all our evidence for the acceptance of sentences as true or false. This being so he appeals to this basis of truth as a way of making meaningful disagreement possible between supposedly irreconcilable perspectives. He focuses on enlarging the basis of shared belief by using individuals' common beliefs and opinions as a starting point from which to improve translation between perspectives.

We argue here that similar bridges of translation should be built between the various education research subfields so that ITS will be more able to utilize multiple perspectives in creating educational interventions. To further this goal we offer here an integration of several of the major education research related subfields. We take a balanced approach in this integration, emphasizing how to achieve harmony between these sometimes combative, competitive subfields. This is not the same sort of project that Allen Newell advocated in finding a unified model of cognition because that project was a call for theoretical integration within a subfield [3], while this call is for a valid method to integrate disparate perspectives on learning for more practical reasons.

3 A Proposed Solution

Normally, a subfield of ITS can be partially defined by the units of analysis [4-5] it chooses to study. Thus, behaviorists study behaviors, cognitive psychologists focus on skills, situated cognition researchers study interactions, design specialists focus on form and function, and cognitive neuroscientists focus on the brain but attach cognitive labels to describe the actions of brain tissue. As we can see, one thing that these ITS-relevant behavioral scientists agree on is that a unit of analysis is necessary to analyze a scientific problem. This assumption helps in translating and aligning the subfields because, across each of the various perspectives, grain sizes are based on sensory evidence. It seems that we should be able to align the related sensory bases of each subfield (essentially agreeing that observation is a valid method) and thereby see how the disparate subfields' theoretical explanations map to each other. It seems in such cases you have three essential ways such subfield alignments may turn out.

3.1 Combination

The first possibility is that two perspectives do not explicitly make scientific conclusions about the same units of grain size. In this case we have two perspectives that are essentially orthogonal to one another, and it seems that the reasonable thing for an ITS designer to do is to incorporate both perspectives in the ITS being designed. At first his might not seem intuitive, but this incorporation may be crucial in both experimental and development ITS projects. This is because in ITS research, unlike research in a field like experimental cognitive psychology, we are ultimately interested in how educational research plays out in practical application. In other words, we are more interested in the ecological validity of our results (that they work in a real situation) because this utility within an ecological placement of the tutor is the most important result we wish to generalize in showing that our ITS will benefit real students in real classrooms. For example, in contrast, in cognitive psychology we are often looking for pure main effects and interactions unconfounded by manipulations of other variables. In cognitive psychology work, an assumption is that one wants to remove confounding variables so that main effects can show through and so that that the result that is found is not actually a hidden interaction with some unmanipulated variable. While this is a valid method given the goals, it is likely to be ineffective in establishing that our design has validity in ecological placements.

3.2 Integration

Our second situation is similar, where we have two perspectives, which, while not the same, nevertheless have clear isomorphism. The work of the Jilk, Lebiere, Anderson and O'Reilly illustrates how this is done in the case of aligning a cognitive theory with neuroscience theory [e.g. 6]. Essentially, while there may be some differences between the perspectives (see below on how to resolve conflict) in this situation the perspectives complement and strengthen each other and it becomes more clear that ITS research needs to pay attention to the learning related conclusions of such hybrid perspectives. In their work, these authors have attempted to identify how the ACT-R symbolic model can be mapped to connectionist architecture. They argue that this

process is mutually informative and constraining and propose that no single perspective can capture the full richness of cognition. For instance, the authors note that their collaboration has led to a realization that neither theory answers the question of where symbols come from [6].

3.3 Resolution

Our final situation is when the ITS researcher encounters two perspectives that differ in what they predict is best for learning at a grain size. While this disagreement between perspectives may be hard fought in many cases, the ITS researcher might note that often times the common sense resolution admits some truth in both perspectives. Often this disagreement centers on issues of balance along a continuum. For example, constructivists often argue that learning is most effective when the student is able to participate in the building of understanding while direct instruction advocates argue that clear communications of information with some repetition are the most effective way of causing learning. In a case like this, most people's sensory experience probably supports some aspects of both theories, and this leads the ITS researcher to suspect a case where balancing the perspectives is most appropriate.

The assistance dilemma is one way to frame this competition between competing factors since the assistance dilemma describes considerations in balancing factors when one side of a continuum between factors can be described as more assistance and the other side of the continuum can be described as greater assistance [7-8]. For instance, the resolution between the dispute about the effectiveness of direct instruction compared to discovery learning [9] can be described as an assistance question because direct instruction is typically characterized as providing more assistance than is discovery learning. By characterizing the tradeoff along a continuum we seemingly convert what was a dichotomous theoretical question into a question about tradeoffs. Once characterized as a question about tradeoffs, it quickly becomes clear that the endpoints of the continuum (completely unassisted discovery and fully scaffolded problem solving) are both unlikely to be very useful. Rather it seems clear that some balance between telling information to and withholding information from the student will necessarily be optimal.

As we can see, by proposing that theoretical differences are defined by variation along a continuum, we avoid diametrical arguments that are insoluble. Of course, not all perspectives will hold up when their validity is inspected relative to other perspectives. We won't present examples of this here, but basically this admits that this method of combining perspectives does allow falsification. Falsification is a state where the ITS researcher has identified the predictions of 2 theories relative to some instructional decision and has good experimental evidence that one of the perspectives is incorrect in its conclusions relative the instructional design decision. If one accepts the original notion that it is plausible to suppose multiple educational factors moderate the influence of each other, then it is only in these cases where one perspective's predictions are probably incorrect that the predictions of that perspective are safe to exclude.

3.4 Implications for Experimentation

When considering this proposal to unify subfields, we can note that the combinatorics of exploring the space of the multiple principles becomes untenable if we wish to use

a bottom up strategy of assuming the traditional null hypothesis (an unmanipulated, “vanilla” state of nature where everything except the tested principle is left out). However, when considering how to proceed with ITS research, it helps to think back to the full meaning of the null hypothesis. In many subfields the null hypothesis is typically considered to be a situation that is as “unmanipulated as possible” and taken to be the default state of nature. It is this typically barren backdrop against which we observe main effects when we manipulate a variable as our alternative hypothesis. However, this bias for simple experiments might simply be taken as a bias about our assumptions of the true default state of nature. If we instead assume a true state of nature that includes multiple “confounders” we may be creating a null hypothesis that is much closer to the default state of nature. Because of this, we might expect that any conclusions we might make will generalize more easily to other ecological valid states of nature (which would likely share many features).

Therefore, it seems that in ITS research we might validly consider the null hypothesis as using as many “confounding factors” as seem plausible from a naturalistic standpoint, determined from a review of the multiple ITS subfields. In this sort of situation, ITS research would begin with a many component ITS system well integrated into coursework and the classroom, and then each of the many factors could be varied experimentally, using typical experimental design to see if their removal reduces performance in the system. For example, while cognitive psychology experimentation answers questions like what is the effect of spaced practice given no other confounding variables using a simple memory task, this new experimental method would ask what is the effect of spacing and its interaction with the retention interval given its use in a complex context of learning items that will need to be used productively. In both examples then, our experiment might manipulate the spacing interval.

The key difference in this approach is that we assume that unexplored factors are present (and unmanipulated in our ecological null hypothesis) so that any conclusions we make about the investigated factors will have validity in a natural situation where we might expect some level of all plausible factors. This approach seems further justified in the case of education research specifically because the plausible threat is not that we will not include some highly effective components in each of our respective systems, but rather the threat is that the highly effective aspects of our systems will be nullified by factors that we have not included in our experiments (perhaps to reduce possible confounds). In such cases, it is possible that what we have considered confounding may actually be necessary to include so as to get valid results. Essentially, we are describing a likelihood of positive moderating variables that effectively gate the effect of the investigated variable. In such a situation, which we propose to be closer to the default classroom situation, it seems logical to do one’s best to include all the possible moderators so as not to preclude an effect of the manipulated variable.

An analogy to the two most common ways statisticians have found to select terms in regression equations is useful to help see the structure of the argument we have made thus far. Unless you have time to search the entire combinatoric space of possible terms, forward selection and backward elimination are the main methods of trying to establish a best regression model. As we may recall, forward selection starts out with an empty model and tests the significance of adding each predictor to the model, adding the terms one at a time in the order of which is most significant. As we may note, this is very similar to how the experimental method is used in behavioral

sciences. Like forward selection, behavioral science methods tend to begin with few variables being manipulated so that the other confounders or moderators should not have any effect. Obviously, like forward selection, our behavioral science methods produce many successes, and allow incremental additions to our progress toward ITS design that uses multiple principles to improve student learning in intelligent tutoring systems.

An alternative to forward selection is backward elimination. Backward elimination begins with all the variables and then begins to remove the ones that are discovered not to influence the predictions. Backward elimination may actually be a superior way to search for the best ITS because it allows us to retain all the potential moderators from the start, unlike forward selection. As was noted by in a review of variable selection, "It is often argued that forward selection is computationally more efficient than backward elimination to generate nested subsets of variables. However, the defenders of backward elimination argue that weaker subsets are found by forward selection because the importance of variables is not assessed in the context of other variables not included yet." [10]

4 Specific Resolutions

Our goal of providing specific examples of the combination of subfields is depicted in Table 1. Table one is organized with a partition of grain sizes in the left column and a subset of the relevant subfields along the top. For reasons of space, we have been forced to group subfields and ignore many findings to focus on some core principles in each subfield. Within each cell then, are the principles that apply at the grain size for the subfield. Using this chart we can then analyze cases where there is either no explicit dispute at the grain size, cases where the perspectives offer complementary approaches and finally cases where there is an explicit conflict to be resolved between the subfields.

4.1 Example Resolutions

Examination of table 1 can be used to help identify places where theories are making potentially conflicting predictions at matching grain sizes. From there, the theories can be broken down further to more closely identify any grain size isomorphisms. After this sensory process of object matching is completed, predictions relative to the objects can be compared for inconsistencies, which can then be resolved as described above.

However, this more clear analysis allows us to notice that many of the disputes between the theories seem to be issues where either side would admit some truth to the other. For instance, while situated/constructivist theorists specify the importance of authentic contexts, there is very little work in the cognitive literature which challenges this notion directly. While much cognitive work might be described as ignoring the importance of authentic contexts, this is very different than proposing that authentic contexts have a negative effect on student learning. Indeed, digging deeper into the cognitive literature allows us to unearth many specific findings that support the importance of context. [e.g. 11]

Table 1. Map of subfield integration for each grain size

	<i>Cognitive, Social, Motivational Psychology</i>	<i>Situated Cognition, Constructivism, Discovery Learning</i>	<i>Artificial Intelligence, Statistics, Computational Modeling</i>	<i>Neuroscience</i>	<i>Perceptual Psychology, HCI and Design</i>
Societal Learning Context		Interaction across levels[14]	SEM models[15]		Interaction design[16] Activity centered design[17]
Individual	Cognitive load[12] Self-explanation[18] Analogy[19] Worked examples[20-21] Scheduling[22] Testing[23] Theory of intelligence[24] Efficacy[25] Goal orientation[26] Vicarious Modeling[32]	Scaffolding[27] Affordances[28]	Models of individual differences[29]	Working memory training[30] School lunches[31]	
Peers		Modeling[14]			
ITS software		Authentic contexts[33] Exploration contexts[34]	Question generation[35] User modeling[36]		Organization of gestalt[37] Pleasure[38]
Teachers		Modeling and Coaching[14]			

Further, there appears to be no explicit reason why cognitive phenomenon would not be important in situated learning. For example, consider cognitive load [12] in real life situations. There seems to be no reason why the putatively cognitive mechanism of cognitive load would not affect students in authentic tasks with real world contexts. Indeed, because authentic contexts often include more details, it seems that an integration of situated theory and cognitive load theory offers advantages. By integrating these theories it would allow us to examine how much authentic context is useful and how much causes extraneous cognitive load. While this sort of synthetic approach is not always simple, it helps to explicitly reveal the best resolution to any contradiction when examining different perspectives on an issue important to ITS development.

5 Conclusions

What we are advocating here has much in common with design based research methods [13]. We are also advocating doing experiments in naturalistic settings and we also agree that “scientists must draw connections to theoretical assertions and claims that transcend the local context” [p.8]. However, unlike design based research advocates, we do not place a special emphasis on a theoretical starting point (e.g. situated cognition theory) for our design based analysis, but rather acknowledge that regardless of the perspective of the researcher (e.g. behaviorist, design theory, etc.) there are distinct advantages of using a backward elimination method that begins with a complex state of nature (the existing ITS system and its interrelations with students, peers and teachers) and then makes experimental variations. In contrast, we have argued that by beginning with a simple system and incrementally adding intervention characteristics the ITS designer will face challenges to ecological validity and have difficulty detecting effects that might be moderated by other factors. Since these moderating factors (e.g. student motivation controls learning even in a cognitively excellent ITS) may block effects that would show in a more natural design, it may even be necessary to use such a backwards procedure depending on the intervention being investigated.

Acknowledgments. This research was supported by the U.S. Department of Education (IES-NCSER) #R305B070487 and was also made possible with the assistance and funding of Carnegie Learning Inc., the Pittsburgh Science of Learning Center, DataShop team (NSF-SBE) #0354420 and Ronald Zdrojkowski.

References

1. Davidson, D.: On the Very Idea of a Conceptual Scheme. *Proceedings and Addresses of the American Philosophical Association* 47, 5–20 (1973)
2. Kuhn, T.: *The structure of scientific revolutions*. University of Chicago Press, Chicago (1970)
3. Newell, A.: *Unified theories of cognition*. Harvard Univ. Pr., Cambridge (1994)

4. Altman, I., Rogoff, B.: World views in psychology: Trait, interactional, organismic, and transactional perspectives. In: *Handbook of environmental psychology*, vol. 1, pp. 7–40 (1987)
5. Matusov, E.: Applying a Sociocultural Approach to Vygotskian Academia: Our Tsar Isn't Like Yours, and Yours Isn't Like Ours'. *Culture & Psychology* 14, 5 (2008)
6. Jilk, D.J., Lebiere, C., O'Reilly, R.C., Anderson, J.R.: SAL: an explicitly pluralistic cognitive architecture. *J. Exp. Theor. Artif. Intell.* 20, 197–218 (2008)
7. Koedinger, K.R., Pavlik Jr., P.I., McLaren, B.M., Alevan, V.: Is it Better to Give than to Receive? In: Sloutsky, V., Love, B., McRae, K. (eds.) *The Assistance Dilemma as a Fundamental Unsolved Problem in the Cognitive Science of Learning and Instruction*, Washington, D.C. (2008)
8. Koedinger, K.R., Alevan, V.: Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review* 19, 239–264 (2007)
9. Kirschner, P.A., Sweller, J., Clark, R.E.: Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist* 41, 75–86 (2006)
10. Guyon, I., Andre, E.: An introduction to variable and feature selection. *Journal of Machine Learning Research* 3, 1157–1182 (2003)
11. Titone, D.: Contextual memory in a virtual world. *Trends in Cognitive Sciences* 5, 376 (2001)
12. Kalyuga, S., Sweller, J.: Measuring Knowledge to Optimize Cognitive Load Factors During Instruction. *Journal of Educational Psychology* 96, 558–568 (2004)
13. Barab, S., Squire, K.: Design-based research: Putting a stake in the ground. *Journal of the Learning Sciences* 13, 1–14 (2004)
14. Collins, A.: Design issues for learning environments. *International perspectives on the design of technology-supported learning environments*, 347–361 (1996)
15. Ramayah, J.: A SEM investigation of e-learning: an illustrated perspective of a course website acceptance model among business students. *International Journal of Information and Operations Management Education* 2 (2007)
16. Kolko, J.: Abductive Thinking and Sensemaking: The Drivers of Design Synthesis. *Design Issues* 26, 15–28 (2010)
17. Gifford, B.R., Enyedy, N.D.: Activity centered design: towards a theoretical framework for CSCL. In: *Proceedings of the 1999 conference on Computer support for collaborative learning International Society of the Learning Sciences*, Palo Alto, California, vol. 22 (1999)
18. Chi, M.T.H., de Leeuw, N., Chiu, M.-H., LaVancher, C.: Eliciting self-explanations improves understanding. *Cognitive Science: A Multidisciplinary Journal* 18, 439–477 (1994)
19. Gentner, D.: Structure-mapping: A theoretical framework for analogy. *Cognitive Science: A Multidisciplinary Journal* 7, 155–170 (1983)
20. Atkinson, R.K., Derry, S.J., Renkl, A., Wortham, D.: Learning from examples: Instructional principles from the worked examples research. *Review of Educational Research* 70, 181–214 (2000)
21. Carroll, W.M.: Using worked examples as an instructional support in the algebra classroom. *Journal of Educational Psychology* 86, 360–367 (1994)
22. Pavlik Jr., P.I., Anderson, J.R.: Using a model to compute the optimal schedule of practice. *Journal of Experimental Psychology: Applied* 14, 101–117 (2008)
23. Karpicke, J.D., Roediger III, H.L.: The Critical Importance of Retrieval for Learning. *Science* 319, 966–968 (2008)

24. Dweck, C.S.: *Self-theories: their role in motivation, personality, and development*. Psychology Press, Philadelphia (2000)
25. Bandura, A.: Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review* 84, 191–215 (1977)
26. Elliot, A., McGregor, H.: A 2 x 2 achievement goal framework. *Journal of Personality and Social Psychology* 80, 501–519 (2001)
27. Wood, D., Wood, H.: Vygotsky, tutoring and learning. *Oxford review of Education*, 5–16 (1996)
28. Greeno, J.G.: Gibson's affordances. *Psychological Review* 101, 336–342 (1994)
29. Navarro, D.J., Griffiths, T.L., Steyvers, M., Lee, M.D.: Modeling individual differences using Dirichlet processes. *Journal of Mathematical Psychology* 50, 101–122 (2006)
30. Jaeggi, S.M., Buschkuhl, M., Jonides, J., Perrig, W.J.: Improving fluid intelligence with training on working memory. *Proceedings of the National academy of Sciences of the United States of America* 105, 6829–6833 (2008)
31. Kleinman, R.E., Hall, S., Green, H., Korzec-Ramirez, D., Patton, K., Pagano, M.E., Murphy, J.M.: Diet, Breakfast, and Academic Performance in Children. *Annals of Nutrition and Metabolism* 46, 24–30 (2002)
32. Bandura, A.: *Principles of behavior modification*, New York (1969)
33. Lave, J.: *Cognition in practice: Mind, mathematics, and culture in everyday life*. Cambridge Univ. Pr., Cambridge (1988)
34. Bicknell-Holmes, T., Hoffman, P.: Elicit, engage, experience, explore: discovery learning in library instruction. *Reference Services Review* 28, 313–322 (2000)
35. Li, T., Sambasivam, S.: Question Difficulty Assessment in Intelligent Tutor Systems for Computer Architecture. In: *Proceedings of ISECON 2003* (2003)
36. Desmarais, M.C., Maluf, A., Liu, J.: User-expertise modeling with empirically derived probabilistic implication networks. *User Modeling and User-Adapted Interaction* 5, 283–315 (1996)
37. Chang, D., Dooley, L., Tuovinen, J.E.: Gestalt theory in visual screen design: a new look at an old subject. In: *Proceedings of the Seventh world conference on computers in education conference on Computers in education: Australian topics*, vol. 8, pp. 5–12. Australian Computer Society, Inc., Copenhagen (2002)
38. Green, W., Jordan, P.: *Pleasure with products: Beyond usability*. Taylor & Francis, Abington (2002)

Recognizing Dialogue Content in Student Collaborative Conversation

Toby Dragon, Mark Floryan, Beverly Woolf, and Tom Murray

University of Massachusetts Amherst
140 Governors Dr. Amherst, MA USA
{dragon,mfloryan,bev,tmurray}@cs.umass.edu

Abstract. This paper describes efforts to both promote and recognize student dialogue in free-entry text discussion within an inquiry-learning environment. First, we discuss collaborative tools that enable students to work together and how these tools can potentially focus student effort on subject matter. We then show how our tutor uses an expert knowledge base to recognize (with 88% success rate) when students are discussing content relevant to the problem and to correctly link (with 70% success) that content with an actual topic. Subsets of the data indicate that even better results are possible. This research provides solid support for the concept of using a knowledge base to recognize content in free-entry text discussion. The paper concludes by demonstrating how this content recognition can be used to support students engaged in problem-solving activities.

Keywords: knowledge base, ill-defined domains, collaboration, inquiry learning.

1 Introduction

One of the major challenges facing developers of intelligent tutoring systems for ill-defined domains is maintaining student focus on appropriate content. Students can easily drift from the topic on which they should focus when exploring open-ended environments. A tutor should provide tools to help students maintain proper focus and center their work on the most vital domain content. The tutor should also recognize the content of student work in order to provide appropriate feedback. The introduction of collaboration creates a greater chance that students become sidetracked, but also provides novel opportunities to automatically recognize whether students are engaged in useful learning activities. Here we present research into these concepts of promoting and recognizing discussion of domain content in the collaborative inquiry learning system, Rashi.

Rashi is an inquiry learning system that provides tools and environments necessary for students to have authentic learning experiences by considering real-world problems. The system provides case descriptions for students to investigate, along with information about how to approach each problem [1]. Various data collection methods (interactive images, interview interfaces, video and dynamic maps) provide open-ended spaces for student exploration and acquaint students with methods

commonly used by professionals to access and organize information. In the *Human Biology Tutor* (the domain used in our current research), students evaluate patients and generate hypotheses about their medical condition. Patients' complaints form an initial set of data from which students begin the diagnostic process. Students move opportunistically from one inquiry phase to another as they sort, filter, and categorize data in order to form hypotheses about the patient's illness. Students can interview the virtual patient (Figure 1 Top), perform a physical examination, or run lab tests.

We have added collaborative tools to this system and demonstrate how we can leverage the information provided by student use of these tools to recognize the domain content on which students are focusing.

There is a large amount of work present in the field of intelligent tutoring systems that is relevant and informative to our current efforts. First, several research groups have explored the potential for collaborative work to improve use of tutoring systems for ill-defined domains. Most prominent among these are the COLER system [2] and COLLECT-UML [3]. Both systems use similar work-sharing techniques to those that we present. These systems give students the ability to view/share work with a team, and offer some form of coaching on both content and collaboration. We have also used as reference Soller's research into collaboration that does not attempt domain level support [4]. This work focuses on monitoring the collaborative efforts of students in an attempt to improve on their skills as team members.

Specific to our new line of research in recognizing the content of chat, we must recognize two areas of research that go beyond our current work. First, there is a large body of research that investigates how adding structure (e.g. sentence starters, dialog phases, etc) to dialog tools can improve student work [5]. Our research investigates recognition in a simpler, more open-ended dialog space. Second, there is the large body of research into natural language understanding that is particularly relevant to our work. These researchers attempt to support students by retrieving related content generated by prior users to provide support. Ravi et al. mine prior student work to find adequate matches [6], while Bernhard and Gurevych analyze the wikiAnswers data store in order to support students [7]. The major challenge to these efforts are

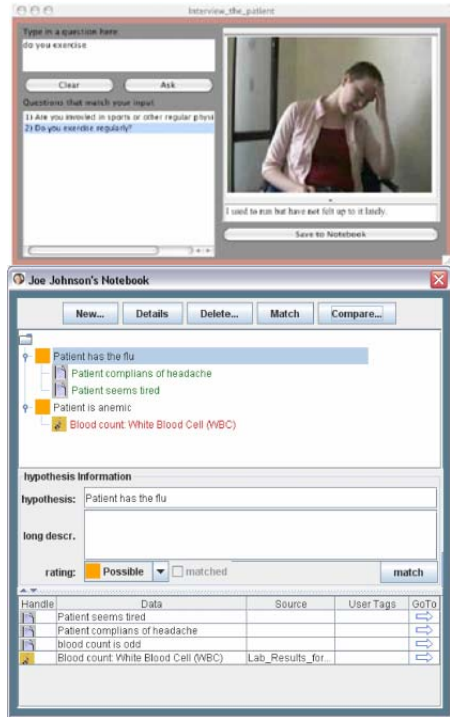


Fig. 1. Students “interview” the patient in the Rashi system (top) and record their hypotheses for a diagnosis along with evidence supporting and refuting that evidence in the notebook (bottom)

data-mining issues; sorting and filtering through large data sets for relevant information. The work presented here differs in that we use a succinct expert knowledge base specifically tailored for the domain and case at hand. This makes the matching of student work tractable and viable without complex NLU techniques, as can be seen by our empirical results. In addition, having ties to our own expert knowledge base can provide new and interesting intervention techniques.

We now describe the collaborative tools added to the Rashi system (Section 2), our approach to recognizing content through use of our knowledge base (Section 3), and the empirical studies used to test this system (Sections 4-5). We conclude with a description of future work (Section 6).

2 Collaborative Features

The Rashi system uses collaborative capabilities to enable students to view and share work within a group. These tools allow users to view each others' notebooks (Figure 1 bottom), and to drag and drop both data and hypotheses from others' notebooks to their own. This supports a variety collaborative activities ranging from students working in tightly knit groups, where each student takes on a role and contributes in a specific manner, to students working mostly independently but sharing ideas and thoughts when reaching an impasse. The system also provides a chat facility that enables students to discuss issues with members of their group, Figure 2. Several features, including text coloring, filtering, and new message notifications increase the usability and quality of the discussion tool.

Students can create a subject for each message, which allows the team to focus on a specific topic, Figure 2, bottom panel. Chat messages can be filtered by these topics and students can easily respond to the subject by clicking on it. In addition, Rashi allows users a one-click method of automatically setting the subject of a new conversation to the contents of an existing Rashi notebook item. This creates an internal link between the conversation and the notebook item, allowing a confused group member to click on the chat subject and be quickly taken to related work in a group member's notebook.



Fig. 2. Collaborative work in Rashi includes student dialogue through chat. Student conversations carry a label (e.g., *Regarding Hyperthyroidism*, middle panel) based on subject (e.g., Graves Disease, bottom panel) typed by a student.

2.1 Effects of Basic Collaborative Features

We evaluated these collaborative features throughout 2007 and 2008 in both high school and college classrooms and found they caused an increase in the amount of

work students completed in the system [8]. Students created more hypotheses, collected more data, and made more connections between their data when collaborative tools were in use. However, we were not able to collect information on whether the increased amount of work was indicative of increased performance. In other words, were the students doing better work, or just more work?

A large study in the Spring of 2009 further investigated whether the amount of work completed within Rashi was indicative of improved performance. We looked for correlations between grade information from teachers and usage statistics from Rashi and found that the amount of work (hypotheses generated, data collected) completed in Rashi was in *no way indicative* of their performance on their final write-ups and overall grade. In other words, seeing an increase in the amount of work students completed was not necessarily an indication of success. We concluded that the major confounding factors involved the issue of students completing a high volume of work, rather than high quality work. Even students who completed a significant amount of work within the system often engage in work that was off-task, repetitive, and/or tangential to the problem at hand, as is often seen when working in ill-defined domains [9]. This result has driven the team's current effort. We built tools to better focus students' collaborative work and use our expert system to automatically recognize the content on which students are currently focusing.

2.2 Content-Focused Collaborative Tools

The chat tool we provided up to this point displays all posts chronologically like other standard chat software. This interface provides an open venue for discussion, but can lead to several forms of noise including the ability to “drown out” other users with a large number of unnecessary intermediate posts [10]. One way to avoid this noise is to provide tools that help students focus on content rather than just providing an open forum. Rashi contains a unique critique-rebuttal feature that fulfills this need and supports students' engagement in topic-oriented discussions, Figure 3. Built into the notebook, this feature enables students to select any item or topic in a group member's notebook and to offer critiques about them. When a critique is given, the owner of the notebook item is notified, and they can respond with a rebuttal, a defense for his or her position, Figure 3, middle panel. This back and forth discussion is by definition focused around subject matter. The two parties can continue to update and

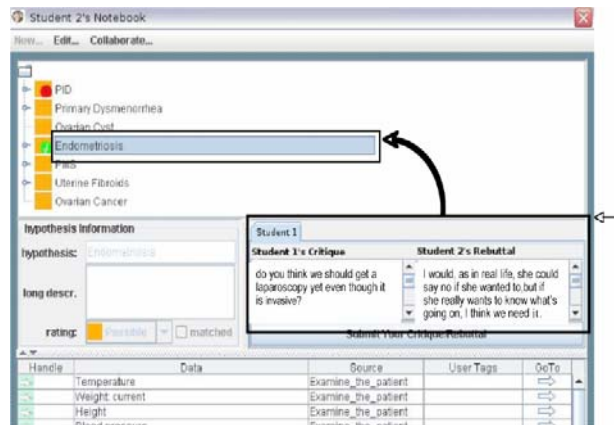


Fig. 3. Student 1 offers a critique of a notebook item (Endometriosis, top panel). The creator of the hypothesis, Student 2, responds to the critique (middle box).

resend their critiques and rebuttals, but only the most recent version of each is shown on the interface. This helps to avoid the drowning out of another user. Since this feature is embedded directly into the notebook, it more tightly couples conversations to students' work, thus helping students engage in constructive criticism and organize discussions around content.

3 Recognizing Content

A key feature of the Rashi system is that it provides deep domain understanding by employing the expert knowledge base [1]. Domain experts created knowledge bases for each Rashi tutor using an external authoring tool, thus enabling Rashi to remain domain-independent yet offer content feedback over many different subjects (e.g., biology, forestry, geology). The Expert Knowledge Base is a directed, acyclic graph of domain-related concepts connected with supporting and refuting relationships, Figure 4. At the top of the graph are hypotheses: high-level possibilities of reasonable explanations of the phenomena presented within the given domain. At the bottom of the graph are data: facts and low-level observations about the case at hand. Relationships connect hypotheses and data, sometimes directly and sometimes through mid-level inferences.

Our current work focuses on human biology, presented through differential medical diagnosis. This knowledge base is generalized across all the cases of the domain (e.g., hyperthyroidism, food poisoning, diarrhea) and has been supplemented to suit cases individually, which plays an important role in recognizing content. Using this knowledge base, various types of support can be offered by an automated coaching agent. When the coach recognizes which diagnosis or evidence the student is discussing, it can show supporting or refuting evidence for a student who is stuck, help students create hypotheses that are consistent with their data, or support students to create correct relationships between their hypotheses and data [1].

In order to provide this help, the system must match student statements to these expert knowledge base elements. This is accomplished by using the search engine library Lucene (lucene.apache.org) to index the individual elements from the knowledge base along with their associated keywords. Student statements are matched by analyzing the result set and scores returned by the search engine. In the past, this process was manually recognized and checked by the student when coaching was to be offered.

The addition of collaboration offers the unique opportunity to recognize situations where students might be helpful to each other. We hope to allow students to support each other at crucial points in time rather than relying on the coach to handle

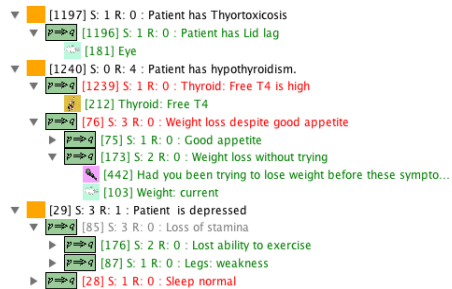


Fig. 4. The Expert Knowledge Base for three diagnoses (thyrotoxicosis, hypothyroidism, and depression) and the evidence supporting (green) and refuting (red) each diagnosis

the entire burden of student support [8]. This can be achieved using this same recognition scheme, but it introduces the need for more precise and constantly updated matching of student work to the knowledge base.

Automated content recognition becomes paramount when supporting collaborative efforts, as the tutor needs information about multiple students to provide support for any one. With individualized coaching, the system needed only to reason about a single student at the time support was requested. However, when attempting to intelligently encourage collaboration, the tutor must reason about all students work at the time when any one student needs support. This drives our current research into automated matching of student content. We apply the same matching algorithm to the chat messages between students to as we do to recognize and respond to content for a single student.

4 Empirical Studies of Automated Chat Analysis

To test the validity and usefulness of our automated matching techniques as applied to chat, we analyzed data from two classroom uses of the software. Both groups used the Rashi collaborative system with chat and critique/rebuttal facilities. The first class included approximately 44 middle-school students from a science summer camp at Hampshire College, Summer 2009, and the second group 14 students in an introductory Hampshire human biology course, Fall of 2009. In both studies, students worked with the system for five days and could use the system on their own computers at any time. During these two sessions, a total of 796 non-blank individual chat messages were sent. Each student, however, could only view the chat happening within his or her group of three to five students. Groups were generally self-selected, with teacher support when individuals were without a group.

An independent judge (a member of the software team not involved or familiar with the matching scheme or the knowledge base) created a set of comparison data over which the efficacy of an automatic matcher could be assessed. The judge rated all 796 chat messages according to a simple metric: a *content* score of 1 if the message specifically referenced knowledge relating to the case and the student constructively discussed the problem in an effort to perform the desired work. Messages received a score of 0 if they did not contain this material. Through this process, the judge found that twelve percent of the Summer 09 messages and 50 percent of the Fall 09 messages were considered on target in terms of content¹.

The first task was to see how often the recognition algorithm could identify that a given chat message referred to domain content. Thus the automated system gave each message a rating of 1 or 0 for content. We found that overall the system had an 88% success rate identifying messages containing domain content, meaning in 88% of the cases, the automated matching decision and independent judgment agreed, Table 1.

The success rates for identifying the presence of domain content indicates the potential for identifying content within chat messages. However, the system can be more useful if it can identify precisely what content is being discussed in order to provide

¹ Several students in summer group diluted the chat by saturating conversations with meaningless messages. The summer content score rose to 31% with these messages removed.

Table 1. Data comparison to demonstrate the efficacy of the automated chat-matching algorithm, which reasoned about whether chat content was appropriate for the domain

Data Set	Total Messages	Judge / Automated Agreement	% Correctly Identified
Summer 2009	496	461	93%
Fall 2009 Case 1	93	84	90%
Fall 2009 Case 2	207	153	73%
Total	796	698	88%

the most useful automated help for students. Therefore, rather than converting the match results to a Boolean value, we set the matching algorithm to return the “best fit” content for that message. Once again, the same independent judge compared the student’s statement with this matched content and marked whether the knowledge base entry was appropriate, Table 2.

As seen in the table, in the best case, 75% of the matches found by the algorithm correctly matched student content with exact concepts in the database and the matching success for all messages was 70%.

Table 2. Results of the automated chat-matching algorithm in correctly identifying the specific content of each student message. The judge’s evaluation of the topic automatically identified by the algorithm.

Data Set	Automated Content Matches	Judged Correct Content Matches	% Content Match
Summer 2009	63	44	70%
Fall 2009 Case 1	69	52	75%
Fall 2009 Case 2	45	25	56%
Total	177	121	70%

5 Discussion of Results

A key result of this research is the development of an automatic recognition algorithm that recognizes domain content in student dialogue, a significant milestone in understanding and evaluating student dialogue. This effort, especially the results in Table 2, accomplish the difficult task of matching an infinite space of input (student’s typed responses) with a set of hundreds of knowledge base statements, rather than attempting to predict a binary decision (content or not) as was shown in Table 1. The information provided in Table 2 is significantly more useful for a coaching agent because the tutor can actually understand the precise content that students are discussing. This provides numerous opportunities for coaching, which we discuss in the following section. These results show that a tutoring system can accurately determine the domain content of student dialogue in the majority of cases by using an expert knowledge base. This is a powerful result that implies building knowledge bases is a viable option for the intelligent tutoring community at large when researchers seek to understand the domain content of student discussion.

Another key result of this work is the realization that development of the knowledge base plays a critical role in the success of concept identification. In Rashi, an over-arching knowledge base exists about human anatomy that operates for the domain of differential diagnosis in general. However, the knowledge base is incrementally built on a case-by-case basis. We put considerable effort into enhancing the expert knowledge base for the cases used in Summer and Fall Case 1. The data from these runs were more useful for dialogue recognition than were the data for Fall Case 2 (as can be seen in Tables 1 and 2). While this leaves us with less-than-ideal results for case 2, it reinforces the idea that our knowledge base structure and creation process are working successfully, since added effort led to direct improvement of matching capabilities.

6 Conclusions and Future Work

Our current work showed success (70-88%) in recognizing content in student discussions. While recognizing content of these scenarios is useful, the content is not recognized correctly in the rest of the cases. We also do not currently have a system for identifying when we have potential mismatches, and any such system would not be perfect. Here we come up against an essential issue of tutoring systems in ill-defined domains. Whenever students are working on authentic, realistic problems and given freedom to work in open-ended environments, coaching systems will have to operate under uncertainty. Therefore, we need to take precautions to avoid an intervention that would be disruptive or counter-productive if the tutoring system were to be mistaken about the content recognition. We keep this in mind as we go beyond recognition and use this content information to guide students with several different resources available in Rashi. The coach will present both passive information sources relevant to current work, and more active interventions where students are interrupted from potentially wasteful behavior and prompted to re-focus on content by discussing with group members.

An example of our more passive support techniques is a “Suggested Topics” list adjoining the chat window. The system will populate the list with items that are related to a group’s current work according to the Expert Knowledge Base. The group can then see the connections and gaps in their collective work. Clickable links will be automatically generated to quickly show students what elements within the tutor are relevant to the domain knowledge associated with their current conversation. By connecting the system in this way, students can quickly move through tutor elements that are related to the chat, and thus students will have swifter access to resources relevant to their discussions. These suggested topics and related links will be updated dynamically as the system continuously watches the collaborative effort of the students.

Content recognition can also provide active interventions in Rashi. Active interventions require that the system detect a specific moment at which an intervention should be given. What opportunities will be recognized as appropriate times for prompting students to discuss the argument with one another? Some likely candidate situations include:

- One student is missing supporting or refuting arguments that another student has identified.
- Two students have partially completed different parts of the same argument structure.
- One student has a hypothesis, and another student has data to support the hypothesis, but neither has formed an argument yet.
- One student has a supporting argument and another student has a refuting argument for the same hypothesis.

A pilot study of this type of intervention was conducted along with the Fall 2009 experiments. The system did not yet have automated matching capability, so students were asked to match their statements manually to the knowledge base in order to receive feedback. Use of the system was so limited that we cannot make any real generalizations. We did recognize anecdotally the potential for such a technique, and the obvious need for an automated matching scheme that would allow for such interventions without the interruption of manual matching.

In order to make these interventions as accurate as possible, future work will also include improvements to the content recognition algorithm by including context in the search process. The automatic recognizer often finds a list of possible subjects for a given chat message. In our experiments we simply used the top match. However, context clues can inform an automatic process when the top match may not be the best choice. The first and foremost of these context clues is *temporal proximity*. Currently, each message is considered independently when seeking a link with the Expert Knowledge Base. However, from our data we see that it is likely the subject of a given chat message is the same or similar to the messages around it. An automatic recognizer can weight potential matches by considering the matched content around the given message.

Another major context clue that has not been exploited in the current matching algorithm is *case-specific proximity*. We noted in our data analysis that large portions of the chat refer to knowledge that is unique to the current case (e.g. a patient-specific allergy). Again, we can use this type of information to weight our matching choices, helping us more accurately pinpoint the specific knowledge base components related to student discussion.

Acknowledgements

This research was funded by an award from the National Science Foundation, NSF 0632769, IIS CSE, Effective Collaborative Role-playing Environments, (PI) Beverly Woolf, with Merle Bruno and Daniel Suthers. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

References

1. Dragon, T., Woolf, B.P., Marshall, D.: Coaching within a domain independent inquiry environment. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 144–153. Springer, Heidelberg (2006)

2. Constantino-Gonzalez, M.A., Suthers, D.D., Escamilla de los Santos, J.G.: Coaching web-based collaborative learning based on problem solution differences and participation. *International Journal of Artificial Intelligence in Education* 13(2-4), 263–299 (2003)
3. Baghaei, N., Mitrovic, A.: From Modelling Domain Knowledge to Metacognitive Skills: Extending a Constraint-Based Tutoring System to Support Collaboration. In: Conati, C., McCoy, K., Paliouras, G. (eds.) *UM 2007. LNCS (LNAI)*, vol. 4511, pp. 217–227. Springer, Heidelberg (2007)
4. Soller, A., Martinez-Monez, A., Jermann, P., Muehlenbrock, M.: From mirroring to guiding: A review of state of the art technology for supporting collaborative learning. *International Journal of Artificial Intelligence in Education* 15(4), 261–290 (2005)
5. Ravenscroft, A., McAlister, S., Sagar, M.: Digital Dialogue Games and InterLoc: A Deep Learning Design for Collaborative Argumentation on the Web. In: Pinkwart, N. (ed.) *Educational Technologies for Teaching Argumentation Skills*. Bentham Science E-Books (2010)
6. Ravi, S., Kim, J., Shaw, E.: Mining On-line Discussions: Assessing, Technical Quality for Student Scaffolding and Classifying Messages for Participation Profiling. In: *Educational Data Mining Workshop for the Conference of Artificial Intelligence in Education*, Marina del Rey, CA, USA, July 2007, pp. 70–79 (2007)
7. Bernhard, D., Gurevych, I.: Answering learners' questions by retrieving question paraphrases from social Q&A sites. In: *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications for the Association for Computational Linguistics* Columbu, Ohio, pp. 44–52 (2008)
8. Dragon, T., Woolf, B.P., Murray, T.: Intelligent Coaching for Collaboration in Ill-Defined Domains. In: *Conference of Artificial Intelligence in Education*, Brighton, England, pp. 740–742 (2009)
9. Lu, J., Lajoie, S.P.: Facilitating medical decision making with collaborative tools. In: *Proceedings the World Conference on Education Multimedia, Hypermedia & Telecommunications*, pp. 2062–2066. AACE, Norfolk (2005)
10. Truth Mapping: A Tool To Elevate Debate: <http://www.truthmapping.com>

Supporting Learners' Self-organization: An Exploratory Study

Patrice Moguel¹, Pierre Tchounikine¹, and André Tricot²

¹ LIG, Université de Grenoble I, France

² CLLE-LTC, Université de Toulouse II, France

Patrice.Moguel@metapatrice.com, Pierre.Tchounikine@imag.fr,
andre.tricot@toulouse.iufm.fr

Abstract. Learners engaged in CSCL macro-scripts are involved in self-organization activities. We present an exploratory study that suggests that Bardram's theoretical model of collective work dynamics is a pertinent basis for both (1) designing interfaces providing a passive support that engages learners in an explicit organization activity, and (2) making learners' organization more easily detectable and analyzable within a perspective of active support.

Keywords: CSCL, Organization, Collective Challenge.

1 Introduction

CSCL macro-scripts are learning scenarios designed to enhance the probability of a group of learners engaging in knowledge-generative interactions such as conflict resolution, explanation or mutual regulation [1]. Such scripts define sequences of activities, create roles and constrain the mode of interaction among peers. Their basic principle is to structure learners' activity to make them engage in an effective collaboration whilst providing some flexibility and avoiding over-scripting (over-structuring), which could sterilize collaborative learning situations [2].

Macro-script settings are particular cases of collective work situations: learners are mutually dependent on their work [3]. CSCW emphasize that actors engaged in such interdependent processes must address an overhead activity, that of articulating (dividing, allocating, coordinating, scheduling, meshing, interrelating, etc.) their respective activities [4,5]. Organization is a meta-level activity that is not focused on the targeted output, but on setting the conditions of the production of this output.

In macro-script settings, taking organization into account is a core issue. First, organization impacts the overall process. If learners fail in building a more or less coherent organization, their engagement may diminish, and the pedagogical objective of promoting knowledge-generative interactions may not be reached. Second, consideration of organizational issues leads to interactions such as building a common ground, planning, resolving conflict resolution, or regulating processes.

By definition, macro-scripts provide learners with a certain degree of flexibility, i.e., let them decide on some aspects of the script enactment. Many experiments reported in the literature show that learners use this flexibility, e.g., in context, divide

tasks into subtasks and adjust their division of labor, define some sub-strategies, or adopt alternative ways of using the technological means provided: they engage in *self-organization* activities [3]. Self-organization is “the meta-level activity that a group of learners engaged in a CSCL script may engage in so as to maintain, within the reference frame that is externally defined by the script, a more-or-less stable pattern of collective arrangement”. In this definition, “self” is meant to highlight that, in such a context, part of the organization is externally set by the script, and part is related to emergent features of learners’ enactment of the script at run-time.

From a general view point, an intelligent/adaptive CSCL framework addressing organization issues should be able to support self-organization by (1) passive features such as offering learners tools to share their plans, and (2) active support based on a certain understanding of learners’ organization and its evolution. In the context of CSCL, active support can consist in individual or collective hints related to a lack of involvement or to the tackling of some tasks (using ITS techniques, see [6] for example). Other central issues are the dynamic adaptation of the scenario and/or of the technical framework offered to the learners [3], in particular in order to provide learners with means that continue to comply with the pedagogical objectives whilst not conflicting with their emergent activity [2,3].

Our general goal is to study how to support learners in their self-organization and prevent collaboration breakdowns. We use as a case study a pedagogical collective challenge, which is a type of macro-script that enhances the role of learners’ motivation: the scenario is less detailed than in basic macro-scripts, and emphasis is rather on introducing a challenge to enhance motivation [7]. This type of setting is particularly prone to self-organization phenomena. We have designed and implemented a computer-based system that supports learners’ self-organization. This system is based on Bardram’s theoretical model of collective work dynamics [4]. We have conducted an exploratory study to analyze the impact of this system. This study suggests the adopted approach is relevant for (1) designing interfaces providing learners with passive support by engaging them in an explicit organization activity, and (2) making learners’ organization more easily detectable and analyzable, which is a *sine qua non* condition for envisaging active support.

We first present Bardram’s model and the way in which we use it, the application used as an experimental field, and the system principles. We then present the exploratory study (6 groups, 3 of which used the system) and the lessons learned.

2 Theoretical Background: Bardram’s Model

Bardram’s model [4] (see Figure 1) introduces 3 basic notions: *co-ordination*, *co-operation* and *co-construction*. Co-ordination denotes the level where actors concentrate on the subtasks they have been assigned. Their work is related to a common goal, but their individual actions are only externally related to each other. They carry out the overall task from the point of view of their individual activity. Co-operation is a level where

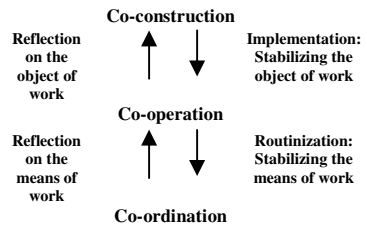


Fig. 1. Bardram’s Model

actors are active in considering the shared objective. This enables them to relate to each other and make corrective adjustments to their own and others' actions according to the overall objective. Co-construction is the level where actors focus on conceptualizing or re-conceptualizing their organization in relation to their shared objects. These 3 levels correspond to analytic distinctions: activity takes place simultaneously at all levels.

Bardram's model highlights the importance of supporting the dynamic transitions that may occur from one level to another during activity. Bottom-up transitions are related to an analysis of the object or the means of the work, which can occur in relation to a breakdown or an explicit shift of focus. Top-down transitions are related to the solving of problems and contradictions, and lead to a stabilization of the object and means of the work. In a learning context, transitions may be induced by learners themselves or by regulation actions, e.g., drawing learners' attention to the fact that they should interact about another level feature in relation to a problem encountered by a learner or by the group, an anticipation of a breakdown, or a pedagogical opportunity. Bardram's model also stresses the fact that perceiving breakdowns is an important dimension of the understanding of collective work dynamics. Breakdowns must be regarded as natural and important events, which should (if the actors are aware of them) challenge the group, and cause a reflection on the means or the object of the work, i.e., a bottom-up transition. A breakdown is solved by a stabilization of the object or means of work, and should end in a top-down transition.

To analyze learners' organization, we have elaborated a structured coding grid (a set of indicators [8]) that allows analysis of learners' organization in terms of co-construction, co-operation, and co-ordination. As examples, co-construction phases are denoted by actions categorized as "understanding of the problem", "elaboration or revision of the general strategy" or "installation of a co-operative structure"; co-operation phases are denoted by actions categorized as "proposition, negotiation, or revision of a precise planning", "decision-making about the organization" or "agreement on how to work together"; co-ordination phases are denoted by actions categorized as "adjustment / application of the adopted organization". We define a breakdown as a difficulty or a contradiction related to an organization that could endanger the dynamics of the collective problem-solving if it were to persist. We use the negation of the criteria and sub-criteria of the coding grid, reformulated as necessary, to detect breakdowns. For example, at the co-construction level, we use as a breakdown criterion "problem not collectively understood" and the two sub-criteria "common representation not clearly established" and "common language not clearly elaborated/acknowledged." Such sub-criteria are not absolute indicators, but rather "symptoms" that may lead to diagnosing a breakdown. Indeed, when considering breakdowns, time is an important issue. When a breakdown is detected, data can be further analyzed to understand if it has been solved, and how, or not solved, and why.

3 The Setting: The "Race with No Winner" Challenge

A pedagogical collective challenge is a learning situation where: (1) the problem is designed to make learners practice some target domain-related or meta-cognitive competencies; (2) a group of learners is involved, as a team, in the solving of the

problem; (3) solving requires the learners to pool their forces; (4) the problem and the setting are designed to create a positive tension that motivates learners [7].

The challenge we use, entitled “the race with no winner”, has been defined by a community of practice dedicated to the use of simulations in mathematics and physics [9]. It is based on a Flash simulation. 10 cars can be put on a track. The cars have different behaviors (e.g., speed or dynamics: some of them stop at one place and re-start after a few seconds, while others do not). Learners must first test all the cars (using the simulation) to collect the data necessary to establish a relation between the departure position of every single car and its arrival on the finishing line. This requires solving equations to determine speed, duration and distance. When learners are ready, the tutor chooses 3 cars, and places one of these cars somewhere on the track (can be anywhere). The learners have to place the 2 other cars on the track so that the 3 cars cross the finishing line at the same time. The simulation is then run to check their solution. There are thus 3 phases: (1) preparing data (measuring and calculating the data for the 10 cars; car behavior does not change from one race to another); (2) calculating where to place the 2 cars after the tutor has placed his car on the track, on the basis of car behavior as calculated at the previous step (to be completed in 30 minutes); (3) running the simulation to check whether the cars have all arrived at the same time. The amount of calculations, the required accuracy and the limited time mean that phases 1 and 2 require a collective organized effort.

Preliminary experiments showed that, although learners are aware of the necessarily collective nature of the work, they do not naturally engage in elaborating an explicit organization and, if any, adopt a very poor organization. This leads them not only to fail (failing is not necessarily an issue: learners learn through their interactions during phases 1 and 2, success in phase 3 is nice but not a *sine qua non* condition for learning) but, more importantly, to feel “in front of a wall”, lose motivation, and not engage in collaboration and knowledge-generative interactions.

4 The System: Supporting Learners’ Organization

The system (named Albatros) is designed to allow a group of distant learners to collectively build a solution (phases 1 & 2). It integrates 2 dedicated shared editors designed to support learners’ self-organization, classical communication (chat) and voting tools, and the simulation. A comprehensive description can be found in [2].

A first editor (see Figure 2), related to the co-construction level, allows the elaboration (and, in case of breakdown, the revision) of a common view and vocabulary (grounding). It allows learners (and supports them, providing a set of predefined items

Id	PRIORITY	INVOLVED NOTION	DATA NAME	DATA DESCRIPTION	TYPE OF ACTION
1	1 High	All the cars	Duration of the race		Measure
2	1 High	All the cars	Speed	Speed of the car	Calculate
3	1 High	Cars that stop	Duration of the stop	Duration of the stop is important because the speeds are constant	Measure
...

Fig. 2. Data definition and actions editor (from the experiment, translated from French)

they can use or be inspired by) to collectively define the data and actions they will need as a list of *actions to be processed*. Each action mentions the involved notion (e.g., “cars that stop”), the name adopted by the group to denote the data (e.g., “duration of the stop”), a textual description, and what is to be carried out in relation to this data (e.g., “measure”). As each action corresponds to a line in the interface, the result is a kind of general problem-solving plan. Learners' activity, as shown by the experiments, is as hoped: definition of a set of lines denoting the way they intend to act and the involved notions (which is of course not necessarily what will happen!), using the suggested items and/or creating some others, each line being elaborated, discussed and negotiated (via the chat). The interface requires each line to be collectively acknowledged via the voting tool. Any learner can come back at any time to what has been defined previously.

Learner #1	Learner #2	Learner #3	No.	PRIORITY	INVOLVED NOTION	DATA NAME	TYPE OF ACTION	CAR # 1			CAR # 2			L
								Learner #1	Learner #2	Learner #3	Learner #1	Learner #2	Learner #3	
?	OK	OK	1	1 High	All the cars	Duration of the race	Measure	5,5	5,5	5,5	27,9	27,9	27,9	
OK	?	OK	2	1 High	All the cars	Speed	Calculate	27,45	?	27,5	5,41	?	5,41	
OK	OK	?	3	1 High	Cars that stop	Duration of the stop	Measure	OK	0	OK	OK	0	OK	
?	?	?	?	?	?	?	?	?	

Fig. 3. Planning definition/execution editor (from the experiment, translated from French)

The second editor (see Figure 3), related to the co-operation and co-ordination levels, allows definition and enactment of a more precise plan. The interface is generated from the collective result of the preceding phase. For every line data/action, 3 columns by car are generated. All cells are initialized with “?”. When used in the “definition” mode, the editor allows learners to declare who will carry out each action: if “Learner #1” clicks on a cell, he/she declares he/she will carry out this task for this car; the “?” is replaced by an “OK” in “Learner #1” column. As the interface is shared, the other learners are aware of this. As each car/action pair is associated with one column per learner, learners can decide to delegate each action to just one, two or three of them. A chat allows synchronous interactions, and learners have to vote on the result to skip to the next phase, using the editor in “execution” mode: the cells marked as “OK” for a learner become editable, i.e., he/she can edit the value. Now that tasks and roles have been fixed, each learner is confronted with his/her tasks. In accordance with Bardram’s model, the experiments show that learners concentrate on the task they have been assigned (learners are individually measuring, calculating, etc.), their individual actions being externally related to each other, but related to a common goal (co-ordination level). The shared interface allows every learner to know what he/she is supposed to do and what the others are doing. Solving evolution is denoted by the fact that the “OKs” are gradually replaced by values.

The way learners can use the editors is very flexible, e.g., they can always come back to previous declarations, start editing values although tasks allocation is not complete (i.e., some “?” remain, see Figure 3), or come back to this allocation. The notion of “plan” (i.e., the succession of lines) is here to be thought of as a resource (and not a constraint) adaptable in context, see Bardram’s work on the non-contradiction between planning seen from this view point and Suchman’s

situated-action views. With respect to these editors, four important issues directly related to intertwining of Bardram's levels can be noticed: (1) the organization/execution interfaces are similar, (2) if something is modified in the organization the execution interface is automatically adapted, (3) only the items that have been changed in the organization are modified in the execution interface (modifying the organization does not mean re-starting from scratch or changing everything), and (4) learners can move from the organization to the execution interface and *vice versa*.

With respect to usability, the coherence organization level / execution level appears to make the interface easily understandable and usable by learners (the design has benefited from exploratory experiments). During action, learners do come back on what they had "planned" (in some cases in an explicit way, i.e., changing their declarations via the editors, in other cases by agreeing to do so more or less explicitly via the chat or *de facto*), and also enact partial plans.

5 Exploratory Study

The exploratory study aimed at suggesting if and how the system impacts the learners' organization and motivation, and helps detect learners' organization. We present hereafter the data and comments (due to a limited number of individuals and groups, no statistically significant results were to be expected).

Experimental setting

Participants. 18 learners (9 females, 9 males, 11th grade science) were randomly assigned to 6 groups of 3 learners.

Materials. Each learner was connected via an individual computer to the system Website. Computers were equipped with software (Camstasia) to record the learners' screens (video file). The chat and the different tools were logged in an XML format.

Protocol. 3 groups used the system before and during the challenge: specific editors (to fix the data to be collected, define the organization and collect the data) and the voting tool to acknowledge decisions (plan, change of mode), simulation, calculator and chat to discuss at all stages. 3 other groups didn't use the system and had only access to the basic tools: chat, simulation, text zone for individual edition, and calculator. The 6 groups worked in the same place, in the presence of the experimenter. They exclusively used the computer-based system to exchange within the group. The proposed scenario was as described previously, identical for the 6 groups: introduction screens explaining the problem and the tools (this part being different for the two groups); phase to prepare the data; launch of the final phase of the challenge when the experimenter perceived the group to be ready (approximately after 2 or 3 hours of preparation, depending on the group).

Data. Every action (mouse click) and message typed by the participants was recorded and associated with an author, a time-stamp, a duration, a type (e.g., "measure"), the tool used, and complementary data such as the numerical value or the tool's mode when pertinent (e.g., organization or execution). The result is a chronological reconstruction of each collective session as a 3-column table displaying the messages

and actions of the three learners of the group. Then, we used the coding grid [8] to identify the co-construction, co-operation and co-ordination phases. To analyze the evolution of motivation, we used the SAL instrument [10].

Results and discussion related to learners' organization

Engaging learners in constructing an explicit organization could represent an obstacle for them: it is a meta-level additional activity they are not used to, they are not naturally convinced of its interest, and the system may have appeared difficult to use. The SAL questionnaire (Table 1) and the fact that groups using the system engage in longer sessions (Table 2: learners decide themselves when to face the final simulation, and groups using the system asked for additional preparation time) suggest this is not the case. Learners' motivation seems to increase. However, this needs to be confirmed statistically using SAL with more subjects. As a matter of fact, learners using the system seem to come closer to the result (Table 4). It can be thought that feeling the group has good chances of success impacts motivation and engagement.

Table 1. Scores, SAL questionnaire

Motivation scores, 6 questions (24 points max.)																		
	Groups using the system									Groups not using the system								
	GR1			GR3			GR5			GR2			GR4			GR6		
	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3
Pré	21	17	17	10	18	19	22	16	17	19	20	20	15	16	10	13	N/A	20
Post	20	20	12	11	19	21	24	15	23	15	19	17	14	12	16	12	N/A	21

Table 2. Duration (and % / total session) at each Bardram's model activity level

Duration / level	Groups using the system			Groups not using the system		
	GR1	GR3	GR5	GR2	GR4	GR6
Co-construction breakdowns	0' (0%)	57'30" (23%)	0' (0%)	105' (62%)	25' (14%)	100' (64%)
Co-construction with no breakdowns	45' (20%)	50' (20%)	72'30" (30%)	27'30 (16%)	30'00 (16%)	0' (0%)
Co-operation breakdowns	5'00" (2%)	10'00" (4%)	2'30" (1%)	0' (0%)	2'30" (1%)	0' (0%)
Co-operation with no breakdowns	10'00" (5%)	32'30" (13%)	7'30" (3%)	5'00" (3%)	25'00" (14%)	12'30" (8%)
Co-ordination breakdowns	77'30" (34%)	50'00" (20%)	37'30" (15%)	12'30" (7%)	0' (0%)	27'30" (18%)
Co-ordination with no breakdowns	77'30" (34%)	32'30" (13%)	37'30" (15%)	0' (0%)	67'30" (36%)	7'30" (5%)
Individual activity	10'00" (5%)	17'30" (7%)	90'00" (36%)	20'00" (12%)	35'00" (19%)	7'30" (5%)
Total session and %	225' (100%)	250' (100%)	245' (100%)	170' (100%)	185' (100%)	155' (100%)

Table 3. Number of chat messages by learner

Messages	Groups using the system									Groups not using the system								
	GR1			GR3			GR5			GR2			GR4			GR6		
Learners	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3	#1	#2	#3
Messages	66	70	49	100	116	116	56	52	67	110	90	78	61	141	47	47	39	8
Average	61,7			110,7			58,3			92,7			83,0			31,3		
Std Dev	11,2			9,2			7,8			16,2			50,7			20,6		

Table 4. Distance to the solution

Delta/solution	Groups using the system			Groups not using the system		
	GR1	GR3	GR5	GR2	GR4	GR6
Delay car #1	0,5	1,1	0	14,3	6,9	0
Delay car #2	2,2	8,2	0	5,2	0,2	0
Total	2,7	9,3	0	19,5	7,1	0
Total (3 groups)	12			26,6		
Average	4			8,9		

Delay groups' cars / tutor's car on the arrival line (success is denoted by delay = 0)

The number of chat messages by learner tends to be balanced for groups using the system, which is not the case for the other groups (Table 3). A “co-construction coherent phase with some subsequent minor revisions” pattern appears within groups using the system: a continuous phase (average duration: 46’40’’) when preparing the challenge and then during the challenge (average duration: 8’20’’), with short adjustment phases (average: 5 phases and 3’50’’ *per* phase). In the other groups co-construction is not a proper phase. This is an important point given the underlying learning assumptions (learning through interaction). This quantitative indication does not imply that these interactions are knowledge-generative. However, other indications appear positive by suggesting learners are also more “in line” (see *infra*).

The system appears to encourage learners to act as experts and not as beginners. Groups not using the system engage in individual problem-solving, and attempt to explain or share the work episodically, when necessary (typically: suddenly understanding there are discrepancies in the measurements or calculations). At the end of the preparation phase, they tend to have individual solutions written in different conceptual languages. Groups using the system engage in a collective definition of the data to be acquired. The analysis shows it corresponds to (and is later used as) a common language, and acts as the premises of a general solving strategy. The system is the center of the activity (average: 84% of the session duration), which makes learners constantly aware of each other’s actions and progress, and naturally engage in comparisons, discussions, revisions of their results, etc. They have no difficulty communicating on data and actions as they use a jointly established language. At the end of the phase preceding the final challenge, the 3 groups had a common solution with negligible variations, and written in a common language.

We have defined “breakdown periods” (Table 2) as periods where learners are no longer in-line: they are individually tackling issues that are not related to each other or conflicting (co-ordination issue), a learner is left on the side, etc. Such periods are not independent from each other, and characterizing them is an issue. A key descriptive feature that can be noted is the total period of time groups remain organized: 51% of total problem-solving time for the groups using the system, and 35% for the others. For instance, considering the key issue of co-construction, groups using the system are involved in a total of 167’30’’ (average 55’50’’) of co-construction with no breakdowns, and a total of 57’30’’ (average 19’10’’) of co-construction breakdowns, while for the groups not using the system the figures are 57’30’’ (average 19’10’’) and 230’ (average 76’40’’).The qualitative analysis shows that the process of the groups using the system is collective right from the beginning, while the others engage first in individual measures and solving for an average of 15 minutes.

The number of breakdowns made explicit and tackled by the learners using the system is greater (22 vs. 14). Qualitative analysis shows this is not correlated to a larger number of problems, but to the fact that these problems are made more easily detectable, and earlier on in the process. Furthermore, as soon as the learners detect a breakdown, they tackle it (11 out of 22 were explicitly solved). For groups not using the system, breakdowns are less explicit, less detectable, and less considered as issues. Unresolved breakdowns are located at the co-operation and co-ordination levels for the groups using the system, and at the co-construction level for the others; they are consequently much more difficult to manage (in particular, because detected very late). As a matter of fact, groups using the system face breakdowns at the co-operation and co-ordination levels because they succeeded in having a common ground, and are effectively working together. 2 of these groups failed in the final challenge, although they remained collective right through to the end (failure was due to learners not succeeding in completing a task, and not to collective-work issues). On the other hand, 2 groups not using the system came to collective-work deadlocks, i.e., lost the collective dimension.

The system does not enforce a particular organization: all groups vary in the strategy they adopt (co-operation and co-ordination levels). As an example, 2 of the groups using the system distributed the measurement and calculation tasks among the various individuals, while the 3rd group built a common ground (co-construction phase), but decided that a single (brilliant) learner would carry out all the calculations, only using the editor in execution mode (the other learners continue to be active, however). This is an important point as our objective is to support learners in making explicit their organization, not to impose one. However, from another point of view, this means that system use does not necessarily result in a balanced organization.

Results and discussion related to learners' organization detection

From a general point of view, analyzing the process of learners not using the system is much more difficult. Level qualification and level-change detection are made difficult because organization and resolution of the problem are heavily intertwined. Learners often change level according to their own process, without noticing or taking into account the opinion or the progress of the other members. Changes are often only understandable *a posteriori*, later on, by re-interpreting actions and chats. The system features (in particular, definition of common ground and alternation of organization and execution modes) facilitate learners' activity characterization.

To understand to what extent the system could support automated analysis, we re-examined the data to simulate an automated analysis for the transition notion. We defined an analysis grid restricted to computational events, i.e., identified what usage of the system (e.g., use of a given tool or change of mode) could be used as transition indicators. We then compared the first analysis (human analyst + general grid) and the second analysis (system-based indicators only), see Table 5. The result shows that 66% of the transitions can correctly be identified by basic automated analysis. The other transitions are not found because they can only be detected by the chat Natural Language (NL) analysis. They typically correspond to learners involved in a task related to a given level (building the plan, enacting planned actions) and using some

corresponding tools but, via the chat, episodically coming back to a feature and interacting to enhance their common understanding (co-construction to co-ordination: 5; co-ordination to co-construction: 7). Other transitions are related to emergent tasks that are not envisaged in the current system (e.g., learner organizing a kind of rehearsal before the final challenge). Erroneous detections (15%) are essentially anticipations: learners shift to a tool associated with co-operation or co-ordination and then chat (or continue chatting) about their organization or their problem-solving. A possible explanation could be that the fact that they engage in a subsequent phase (via the system) leads them to additional organization-related interactions.

Table 5. Automated vs. human analysis of transitions

Transitions		Human analysis				Automated detection (simulation)											
		GR1	GR3	GR5	Σ	correct				not detected				incorrect			
Groups		GR1	GR3	GR5	Σ	GR1	GR3	GR5	Σ	GR1	GR3	GR5	Σ	GR1	GR3	GR5	Σ
Top-down	co-con. / co-op.	1	3	1	5	1	3	1	5	0	0	0	0	0	0	2	2
	co-con. / co-ord.	6	4	6	16	3	4	4	11	3	0	2	5	0	0	0	0
	co-op. / co-ord.	2	4	1	7	2	4	0	6	0	0	1	1	0	2	1	3
Bottom-up	co-op. / co-con.	0	2	1	3	0	0	0	0	0	2	1	3	0	0	0	0
	co-or. / co-op.	1	3	1	5	1	3	0	4	0	0	1	1	0	1	0	1
	co-or. / co-con.	6	4	5	15	3	2	3	8	3	2	2	7	0	0	0	0
Total		16	20	15	51	10	16	8	34	6	4	7	17	0	3	3	6

6 Conclusions

Providing organizational support has necessarily an impact on learners' activity. Results so far suggest that our system (1) supports learners' self-organization whilst not imposing a given strategy, (2) promotes knowledge-generative interactions (co-construction of a common ground and strategies; mutual regulation; resolution of conflicts such as breakdowns), and (3) does not negatively impact motivation (although making learners engage in a meta-level additional activity, whose interest is not obvious for them), and (to be confirmed) seems rather to enhance motivation.

Regarding intelligent support, a core result is that the system does not disrupt learners in their problem-solving (on the contrary, it seems to support them). Given this usability result, it makes sense to make learners use the system, and thus to benefit from the major advantage of the adopted approach: the structural correspondence between the theoretical background, the system, and the analysis grid. This correspondence allows the system traces to be interpretable (to a certain extent) in the terms of the model. Preliminary results for transitions, however, show that a basic automated analysis can support a human tutor by drawing his attention, but remains insufficient for automated monitoring retroactions (66%) and must be completed by NL analysis techniques. The next steps in this research are to enhance the system to increase the percentage of transitions detectable by basic trace analysis, envisage the use of NL analysis techniques, address the issue of breakdown detection (which is likely to request more NL analysis capacities than transitions), and to model active support (hints to individuals and/or the group, etc.) based on these analyses.

References

- [1] Fischer, F., Kollar, I., Mandl, H., Haake, J.M.: Scripting computer-supported communication of knowledge. Springer, Heidelberg (2007)
- [2] Dillenbourg, P., Tchounikine, P.: Flexibility in macro-scripts for CSCL. *Journal of Computer Assisted Learning* 23(1), 1–13 (2007)
- [3] Tchounikine, P.: Operationalizing macro-scripts in CSCL technological settings. *International Journal of Computer-Supported Collaborative Learning* 3(2), 193–233 (2008)
- [4] Bardram, J.: Designing for the Dynamics of Cooperative Work Activities. In: *CSCW 1998 Conference Proceedings*, Seattle, pp. 89–98 (1998)
- [5] Schmidt, K., Bannon, L.: Taking CSCW Seriously: Supporting Articulation Work. *CSCW* 1(1-2), 7–40 (1992)
- [6] Walker, E., Rummel, N., Koedinger, K.: To tutor the tutor: Adaptive domain support for peer tutoring. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 626–635. Springer, Heidelberg (2008)
- [7] Moguel, P., Tchounikine, P., Tricot, A.: Supporting Learners' Organization in Collective Challenges. In: Dillenbourg, P., Specht, M. (eds.) *EC-TEL 2008*. LNCS, vol. 5192, pp. 290–303. Springer, Heidelberg (2008)
- [8] Moguel, P., Tchounikine, P., Tricot, A.: A Model-Based Analysis of the Organization of Students Involved in a Computer-Based Pedagogical Challenge. In: *CSCL 2009 Conf. Proceedings*, pp. 73–77 (2009)
- [9] http://www.patrickmoisan.net/copains/course_sans_gagnant.html
- [10] Marsh, H.W., Hau, K., Artelt, C., Baumert, J., Peschar, J.L.: OECD's brief self-report measure of educational psychology's most useful affective constructs: Cross-cultural, psychometric comparisons across 25 countries. *International Journal of Testing* 6(4), 311–360 (2006)

Exploring the Effectiveness of Social Capabilities and Goal Alignment in Computer Supported Collaborative Learning

Hua Ai¹, Rohit Kumar¹, Dong Nguyen¹, Amrut Nagasunder², and Carolyn P. Rosé¹

¹ Language Technologies Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, Pennsylvania, 15213

² National Institute of Technology Karnataka, Surathkal, India
{huaai, rohitk, cprose}@cs.cmu.edu,
{dong.p.ng, amrut.nagasunder}@gmail.com

Abstract. In this study, we describe a conversational agent designed to support collaborative learning interactions between pairs of students. We describe a study in which we independently manipulate the social capability and goal alignment of the agent in order to investigate the impact on student learning outcomes and student perceptions. Our results show a significant interaction effect between the two independent variables on student learning outcomes. While there are only a few perceived differences related to student satisfaction and tutor performance as evidenced in the questionnaire data, we observe significant differences in student conversational behavior, which offer tentative explanations for the learning outcomes we will investigate in subsequent work.

Keywords: social interaction, conversational agents, collaborative learning.

1 Introduction

Much prior work demonstrates the advantages of group learning over individual learning, both in terms of cognitive benefits as well as social benefits [1][2]. From a cognitive standpoint, a major advantage to learning in a group is that when one is exposed to an alternative perspective, it provides the opportunity to question one's own perspective, which in turn offers an opportunity for cognitive restructuring. In order to achieve this benefit, a major emphasis of work on scaffolding collaborative learning [3] has focused on drawing out aspects of an issue where there is a disagreement between students so that they will address the disagreements explicitly and benefit from that negotiation process. In line with this, work on formalizing the process of collaboration in order to identify events that are valuable for learning has in many cases focused on formalization of argumentation [4].

In this paper, we again investigate how conflict and negotiation relate to learning by characterizing spans of text as exhibiting a bias towards one stance or another. As a methodological contribution, we discuss how we use as a tool for quantifying bias a state-of-the-art topic modeling technique from the field of language technologies referred to as ccLDA [5]. On another dimension, we examine the impact of the

tutor's social behaviors on student learning. As we have seen from recent work, tutors capable of engaging in social interactions with groups can be significantly more effective than tutors that have no social capability [6]. However, there is a trade-off between spending time on social behaviors and spending more time talking about task-related content. In this study, we manipulate the extent to which the agent exhibits social behaviors designed to build solidarity between the student and the agent in order to investigate whether these solidarity building behaviors either magnify or dampen the effect of the bias manipulation.

We begin with a classroom study of collaborative engineering design where students work in pairs on the design of a power plant. This learning task involves negotiating between two competing objectives. Specifically, one student in the pair is assigned to the goal of maximizing the power output of the power plant. The other student, in contrast, is assigned the goal of minimizing the negative environmental impact of the design. Similar to our prior studies of collaborative learning [6][7], a conversational agent participates with the students in the design task in order to provide support. The unique contribution of this study is that we explore the introduction of bias in the way the agent presents information towards one student's stance or the other. In addition to investigating the effect of the manipulation on learning, we investigate the extent to which students are sensitive to displays of bias in the language of their human partner and that of the agent, to what extent the agent's displayed bias affects the bias displayed by the individual students, the interaction between the individual students and the agent, and finally the interaction between the pair of students themselves.

In the remainder of the paper we first review the literature on the connection between conflict and learning. We then describe our experimental study. Next we explain our methodology for measuring bias. We then detail our results. We conclude with discussion and directions for future work.

2 Previous Work

Previous work on building socially capable conversational agents focuses on designing social interaction strategies that fall into the category that Isbister and colleagues [8] have referred to as social interface within their taxonomy. The goal of these strategies is to enable users to interact in an intuitive and natural way with the agent to perform some intended task. For example, Morkes et al [9] implemented a task-oriented conversational agent that uses preprogrammed jokes. They show that this humor-equipped agent is rated as better and easier to socialize with by human participants. In another line of work, Wang and Johnson [10] found that learners who received polite tutorial feedback reported higher increase in self-efficacy at the learning task. Social strategies are also found to be effective in multi-party conversations, such as in computer supported collaborative learning. Higashinaka et. al. [11] found that an agent's use of emphatic expressions improved both overall user satisfaction and user rating of the agent. In general, computer agents which are friendly and helpful to users are favored.

In a multi-party conversation between students and a computer tutor, it is important to build rapport between students and the computer tutor so that students respect the

role of the tutor as a participant in the conversation and thus are more likely to engage with them in a productive way. At the same time, it is also important to ensure that rapport building activities do not detract from the students' task-related objectives. Here we draw wisdom from previous work [12] describing conversational processes through which the participants align to one another as an attempt to promote efficiency and effectiveness in their communication. Although this work pertains to human-human conversation, there is reason to apply it to human-computer interaction as well. Reeves and Nass [13] have shown that although people do not generally believe that computers have either human perceptions or human feelings, they still behave towards computers in a way that seems to assume that they do. This is displayed in the way that humans respond to what would be social cues in human-human interactions even when those cues are coming from computers. The pattern of results from the Reeves and Nass studies might suggest that in conversational interactions between humans and computer agents, we may see the same alignment strategies surfacing. And this has been confirmed by a series of studies conducted by Nass and his colleagues [14][15]. They report that human users align to conversational agents at both lexical and syntactical levels. The extent of the alignment depends on the users' beliefs about their conversational partners' competence.

Due to the restricted syntactic complexity exhibited in the conversational behavior of our tutor agents, in this study we only focus on examining lexical alignment. We explain in Section 4 how a cCLDA model is used to measure the bias of a student's stance in terms of the topics represented in the user's utterances, which are modeled as distributions of lexical items. These topics are later used to measure the alignment of student utterances at the lexical level.

3 Method

We are conducting our research on dynamic support for collaborative design learning in the domain of thermodynamics, using as a foundation the CyclePad articulate simulator [6], which allows students to implement design ideas using graphical interface widgets. In the collaborative design exercise described below, students work in pairs to struggle with trade-offs between power output and environmental friendliness in the design of a Rankine cycle, which is a type of heat engine.

106 undergraduate students from a mechanical engineering class at Carnegie Mellon University participated in the study by attending one of six lab sessions, in which we strictly controlled for time. At the beginning of each lab session, students were lead through formal training on the simulation software Cyclepad. They then practiced to optimizing some Rankine cycles in Cyclepad using information from a booklet given to them, which was developed by a professor from the Mechanical Engineering Department. Subsequent to this, they took the pre-test, immediately before the experimental manipulation. The exploratory design exercise, which followed, was where the students worked in pairs using CyclePad and the ConcertChat collaboration environment [16]. Students are randomly assigned into pairs and paired students do not sit next to each other so that their only communication is through the ConcertChat online learning environment. We assigned each student within each pair to a different competing goal, with one student instructed to increase power output as much as possible and the

other student instructed to make the design as environmentally friendly as possible. Students were instructed that they should negotiate with their partner in order to meet their own assigned design objective, namely either to maximize Power output (in the Power condition) or to minimize environmental impact (in the Green condition). This collaborative design exercise was followed by the post-test and the questionnaire and finally a closing activity in which the student was able to work independently with CyclePad to improve the design they developed with their partner.

As mentioned, during the collaborative design interaction, students use a collaboration software package called ConcertChat [16] to chat with each other in pairs as well as using the digital whiteboard associated with that environment to pass graphical information back and forth to one another. In all cases, a tutor agent participated with the students in the chat. The experimental manipulation only affected how the tutor agent behaved. In all other respects, the experience of students in all conditions was the same.

The experimental manipulation was a 3X3 between subjects design. Each student pair is assigned to one of the nine conditions randomly. For the first independent variable, we contrast 3 social conditions (High, Low, and None) where dialogue agents present different amounts of social behavior within the chat environment. Our dialog agent exhibits three different types of positive social-emotional behavior: showing solidarity, precipitating tension release, and agreeing. In most cases, these strategies are realized by prompts that appear in the chat. The frequency of social behavior in our socially capable tutors is regulated using a parameter that specifies the percentage of tutor turns that can be social prompts. Specifically, the threshold parameter is 15% in the case of the Low social tutor and 30% in the High social tutor. In the Nonsocial condition, no social behavior is realized.

For the second 3 level independent variable, we designed 3 conditions in which the dialogue agent showed alignment either towards the Green condition, the Power condition, or neither. In this way, students could be thought of as being in one of three different conditions in relation to the tutor agent, namely Match (where the student's goal orientation condition matched the alignment of the tutor), Mismatch (where the student's condition is the opposite of the goal alignment exhibited by the tutor), or Neutral (where the tutor showed no bias). In all cases, the information presented by the tutor is the same. The only difference is the bias exhibited. For example, where the Green biased tutor might say "What is bad about increasing the heat input to the cycle is that it increases the heat rejected to the environment." The neutral tutor would simply say "Increasing heat input to the cycle increases the heat rejected to the environment."

As outcome measures, we examined learning gains between Pre and Post test. 35 multiple choice and short answer questions were used to test analytical and conceptual knowledge of Rankine cycles. We also analyzed the conversational behavior in the chat logs. Finally, we evaluated answers to affective questionnaire items that measure students' self-efficacy, perceptions of task success, and assessment of the quality of the interaction with their partner and with the agent.

4 Modeling Conversational Dynamics

In this study, we measure the bias of a system/user utterance towards one stance or another by applying a topic discovery model on our tutoring dialogs [5]. Latent

Dirichlet Allocation (LDA) models have been widely used to discover topics on large collections of unannotated data [17] by modeling the word distributions represented in the data. For example, it has been used to predict responses to political webposts [18], to study the history of different research fields [19], and so on. What is unique about our application of this technology is that we apply it to conversational data for the purpose of modeling how users are interacting with each other. For each utterance, we compute a score to represent to which degree the utterance displays a bias towards one perspective or another.

In our study, we apply a cross-collection Latent Dirichlet Allocation (ccLDA) model [5], which is a variant of the LDA model that represents how the same topics might be represented differently by speakers representing different points of view. In the ccLDA approach, corpora are represented as collections of documents. ccLDA will construct a topic model for each collection, where these collection-specific topic models represent what is distinct about how those topics are expressed within that collection. A background model is also constructed to represent the commonalities across different collections. A model with this structure can be used to compare multiple text collections by capturing similarities and differences across them in terms of how the same topics are expressed. Since the two students who participate in each pair are assigned different objectives at the beginning, it is intuitive to apply the ccLDA model to model how the students in the two different conditions discuss similar topics, but express a different point of view through those topics.

Table 1. Topics Extracted from ccLDA

Topic 1			Topic 2		
Background	Green	Power	Background	Green	Power
Heat	11000	yah	power	low	generates
quality	Values	blades	decreases	500	makes
right	different	sir	nuclear	12800	85
max	makes	dunno	Make	sort	different
decrease	larger	kk	85	1	7000
possible	graphs	x85	cycle	tutors	12000
goes	bit	rejected	work	effeciency	qdot

To use the ccLDA model, we first separate our dialog data into three collections: those turns that were contributed by the student in the Green condition, those turns that were contributed by students in the Power condition, and those turns contributed by the tutor agent. Thus, for every dialogue, we produce two documents, one containing the concatenation of all the contributions from the student assigned to the Green condition, and the other containing the concatenation of all of the contributions from the student assigned to the Power condition. Our ccLDA model has two collections, namely, a Green collection and a Power collection. We do not include the tutor turns within either collection. When we apply ccLDA to this corpus, then, we get three different topic models, namely, one associated with the Green perspective, one

associated with the Power perspective, and one background model representing what is common between the two. When applying ccLDA, one must set a parameter for the number of topics. Because our corpus is relatively small, we set this value to 2. Thus, in all three models, we have the same 2 topics, where a topic is defined as a distribution of words, where the probabilities represent the strength of association between the word and the topic within the model. Table 1 gives an example of the top 7 words selected for each data collection for the two topics.

Table 2. Three types of topic associations

Author	Text	G_Max	P_Max	G_Avg	P_Avg	G_Wt	P_Wt
Stu1	whats ur goal?	0	0	0	0	0	0
Stu2	green as possible	1	0	0.5	0	0.5	0
Stu1	mine is generates the most power	0	2	0	2	0	2
...							
Tutor	If you increase the max temperature, what happens to the efficiency?	1	0	0.5	0	0.5	0
Tutor	Cycle Efficiency improves by increasing Tmax.	0	0	0	0	0	0

We designed three metrics for estimating bias towards either the Green perspective (G) or the Power perspective (P) using our ccLDA model. An example where these metrics are applied is presented in Table 2.

Max Topic-word association (G_Max and P_Max). In the Max Topic-word approach, for each collection specific model we compute a score for each topic, where we count the number of words in the list of the N most strongly associated words with that topic in the corresponding model. The largest number identified for any one topic within that collection specific model is the score for that collection. In this way, we can compute a score for each perspective, since there is one collection specific model per perspective. Hence, for a piece of text that has 2 terms matching with Topic 0 of Green and 1 term with Topic 1 of Green, we would consider 2 to be the score for Green. By averaging over all contributions for the same student within a conversation, it is possible to use this metric to get an average Max Topic-word score for each student for each perspective.

Average Topic-word association (G_Avg and P_Avg). In the Average Topic-word approach, we find the topic-word associations, as in the previous approach. But here, we average scores across topics within a collection specific model rather than choosing the maximum value.

Weighted Topic-word association (G_Wt and P_Wt). This is a heuristic approach that is similar to the previous approach but which uses the weights of topic terms

provided in the ccLDA probability model distributions. Whereas in the Average Topic-word approach, each word contributes 1 to the topic specific sum we compute for each topic in each collection specific model, here we add a weight that is computed by multiplying the weight of the term within the background model with the weight of that word in the topic within the collection specific model. We observe that the background models prioritize important, domain-specific terms by giving higher weights. Hence, in this approach, we consider the product of the weight of a word in the Background model and its weight in the specific collection so that the relative weighting of domain important terms is more important for the final weight than terms that are less important for the domain.

We validated the metrics by verifying that students in the Green condition were assigned higher Green bias scores than students in the Power condition, and vice versa. This was true in all cases, although the differences were only statistically significant for the first two metrics. All three metrics were highly correlated, with R values between .68 and .99. We further validated the metrics using data from a questionnaire where students were asked to rate their partner based on how hard they perceived that their partner attempted to build an environmentally friendly power plant. The first and third metrics showed a significant correlation in the expected direction with these answers.

5 Results

We analyze student performance from three directions. The first one is the direct outcome of the tutoring sessions – the learning gains; the second is perceived user performance – user’s subjective opinions from the questionnaire data; and the last one is the observed user performance – measures related to conversational behaviors.

5.1 Learning Outcomes

Recall that our experimental manipulation was composed of two independent factors, which we refer to here as Social (No Social, Low Social, and High Social) and Match (Yes-Match, No-Match, and Neutral). We first look at the most important evaluation standard in tutoring applications – the student learning gains. Using an ANCOVA with Objective Post-test as the dependent variable, Objective Pretest as a covariate, and Social and Match as independent variables, and Session as a random variable, we determined that there was a significant effect of the Social Manipulation ($F(2,94) = 5.27$, $p < .01$) where the Low Social condition was significantly better than the other two, with an effect size of .83 standard deviations in both cases. There was a marginal interaction between Social and Match $F(2,94) = 2.57$, $p = .08$, where Low Social is only significantly better than the other conditions in the case where Match is Yes-Match. All other combinations of Social and Match were statistically indistinguishable. In general, students learn the most in the condition with the tutor that showed a bias towards their design goal (Yes-Match) and Low Social. Based on the interaction effect between Social and Match, we believe that it is beneficial for the computer tutor to not only establish the appropriate level of social connection with the students,

but also been viewed as supportive of the students' objectives in order to maximize student learning outcomes.

5.2 Questionnaire Data

We then look into the questionnaire data to see whether the students perceive the social and goal manipulation we designed in this study. Using an ANOVA for each questionnaire question as the dependent variable and Social and Match as independent variables, we determined that there was a marginal effect of Match on rating of tutor as supporting the student's objectives ($F(1,102) = 2.77, p = .09$), where the tutor was seen as supporting students marginally more in the case where the goals matched. There was no effect of either variable on the perception of whether the tutor supported the partner's goal.

The effect of the Match manipulation was demonstrated in other aspects of the experience however, according to the questionnaire. For example, on the questions designed to assess the extent to which a student's partner influenced their perspective as a result of the conversation, we observed a significant interaction effect between the Social manipulation and the Match manipulation, such that when the tutor did not exhibit any bias, there was no significant effect of the Social manipulation, but with either agent that showed a bias, either matching the student's bias or the partner's bias, the High social condition significantly reduced the perceived influence of the partner's perspective. Using our bias detection approach, we determined that students were significantly more distinct from their partner in terms of measure of bias in the case where the tutor showed a bias towards one perspective or another, thus magnifying the contrast between the students. This could explain the pattern of behavior we see here. In the case of the neutral tutor, the polarization was less, so the dampening effect of the Social manipulation would not be felt as strongly.

5.3 Conversation Data

Apart from questionnaire data, we can observe an effect of our experimental manipulation on conversational patterns. We have already discussed effects related to bias in the conversation. Here we measure the extent to which students were sensitive to the social aspects of the tutor's behavior that we manipulated through our two independent variables. We began by manually classifying student turns into three categories:

- AboutSocial – student turns on social behaviors, including greetings, farewell, smiling faces, rude words, jokes
- Offtask – student turns talking about off-task topics, like weekend plans, etc
- AboutTutor – student turns that make negative comments about the tutor

We computed the number of AboutSocial, Offtask, AboutTutor turns for each student. Using an ANOVA for each of the three categories as a dependent variable and Social and Match as independent variables, we observe that there is a significant effect on AboutSocial ($F(2,29)=9.91, p<0.0001$), where a student's social behavior is significantly lower in the condition with the No Social tutor than with the Low Social tutor (with an effect size of 1.8) and High Social tutor (with an effect size of 2.0). Similarly, there is a significant effect on Offtask ($F(2,97) = 3.30, p < .05$), where

students engage in more off task behavior in the No social condition than in the Low and High social conditions (with effect size of .35 standard deviations in both cases). We also observe a significant effect on AboutTutor ($F(2,97) = 5.74, p < .005$), where students utter more negative comments about the tutor in the High social condition than in the Low Social condition (an effect size of 1.1) and the No Social condition (an effect size of 1.28).

Based on our results, we suggest that students will show more social behaviors as well as focus more on the task when the tutor exhibits social behaviors. However, when the tutor performs too much social behavior, the students get distracted and start to make fun of the tutor. This is in addition to the dampening effect of the influence students were perceived to have on one another in the High social condition.

6 Conclusions and Current Directions

In this paper we have described an investigation into the issue of competing biases or stances, and how their presence in a conversation, from human or computer participants, affects the learning, interactions, and perceptions of the encounter. Specifically, we describe a conversational agent that has the ability to exhibit bias towards one perspective or another as well as the ability to exhibit social-emotional behaviors that are designed to build solidarity. We describe a study in which we independently manipulate the social capability and goal alignment of the agent in order to investigate the impact on student learning outcomes, interactions, and perceptions. We observe a significant interaction effect between the social and goal alignment manipulation which suggest that the two strategies need to be considered together when designing tutoring systems. In addition, while there are less perceived differences in the student questionnaire data, we observe significant differences in student conversational behaviors in different experimental conditions. We suggest that an appropriate amount of tutor social behavior can help to engage students in the conversation. Furthermore, aligning with student's goals can improve the students' learning. In the future, we will further investigate how to design the appropriate level of tutor social behaviors and how to design tutor's dialog content to align with the learning objectives of both student partners in the conversation.

Acknowledgements

This work is funded by NSF grant number EEC 0935145. We thank the anonymous reviewers for their insightful suggestions.

References

1. Strijbos, J.W.: The effect of roles on computer supported collaborative learning. Doctoral Dissertation, Open University, The Netherlands (2004)
2. Baker, M., Lund, K.: Promoting reflective interactions in a CSCL environment. *Journal of Computer Assisted Learning* 13, 175–193 (1997)

3. Kollar, I., Fischer, F., Hesse, F.W.: Computer-supported cooperation scripts - a conceptual analysis. *Educational Psychology Review* (2006)
4. Weinberger, A., Fischer, F.: A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Journal of Computers & Education* 46(1) (2006)
5. Paul, M., Girju, R.: Cross-Cultural Analysis of Blogs and Forums with Mixed-Collection Topic Models Export. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (2009)
6. Kumar, R., Rosé, C.P., Wang, Y.C., Joshi, M., Robinson, A.: Tutorial Dialogue as Adaptive Collaborative Learning Support. In: *Proceedings of Artificial Intelligence in Education* (2007)
7. Chaudhuri, S., Kumar, R., Joshi, M., Terrell, E., Higgs, F., Aleven, V., Rosé, C.P.: It's Not Easy Being Green: Supporting Collaborative "Green Design" Learning. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 807–809. Springer, Heidelberg (2008)
8. Isbister, K., Nakanishi, H., Ishida, T., Nass, C.: Helper Agent: Designing an Assistant for Human-Human Interaction in a Virtual Meeting Space. In: *Proceedings of CHI* (2000)
9. Morkes, J., Kernal, H.K., Nass, C.: Effects of humor in task-oriented human-computer interaction and computer-mediated communication: A direct test of SRCT theory. *Human-Computer Interaction* 14(4) (1999)
10. Wang, N., Johnson, L.: The Politeness Effect in an intelligent foreign language tutoring system. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 270–280. Springer, Heidelberg (2008)
11. Higashinaka, R., Dohsaka, K., Isozaki, H.: Effects of Self-Disclosure and Empathy in Human-Computer Dialogue. In: *Proceedings of 2008 IEEE Workshop on Spoken Language Technology* (2008)
12. Brennan, S.E., Clark, H.H.: Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory and Cognition* (1996)
13. Reeves, B., Nass, C.I.: *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places*. Cambridge University Press, New York (1996)
14. Pearson, J., Hu, J., Branigan, H.P., Pickering, M.J., Nass, C.: Adaptive language behavior in HCI: How expectations and beliefs about a system affect users' word choice. In: *Proceedings of CHI conference on human factors in computing systems* (2006)
15. Branigan, H.P., Pickering, M.J., Pearson, J., McLean, J.F., Nass, C.: Syntactic alignment between computers and people: The role of belief about mental states. In: *Proceeding of the 25th Annual Conference of the Cognitive Science Society* (2003)
16. Concert Chat (2006), <http://www.ipsi.fraunhofer.de/concert/>
17. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. *Journal of Machine Learning Research* (2003)
18. Yano, T., Cohen, W., Smith, N.: Predicting response to political blog posts with topic models. In: *Proceedings of the 47th Conference of NAACL* (2009)
19. Paul, M., Girju, R.: Topic modeling of research An interdisciplinary perspective. In: *Proceedings of the International Conference on Recent Advances in Natural Language* (2009)

Virtual Humans with Secrets: Learning to Detect Verbal Cues to Deception

H. Chad Lane¹, Mike Schneider¹, Stephen W. Michael²,
Justin S. Albrechtsen², and Christian A. Meissner²

¹ Institute for Creative Technologies
University of Southern California

² Department of Psychology
University of Texas at El Paso

Abstract. Virtual humans are animated, lifelike characters capable of free-speech and nonverbal interaction with human users. In this paper, we describe the development of two virtual human characters for teaching the skill of deception detection. An accompanying tutoring system provides solicited hints on what to ask during an interview and unsolicited feedback that identifies properties of truthful and deceptive statements uttered by the characters. We present the results of an experiment comparing use of virtual humans with tutoring against a no-interaction (baseline) condition and a didactic condition. The didactic group viewed a slide show consisting of recorded videos along with descriptions of properties of deception and truth-telling. Results revealed that both groups significantly outperformed the no-interaction control group in a binary decision task to identify truth or deception in video statements. No significant differences were found between the training conditions.

Keywords: virtual humans, deception detection, intelligent tutoring systems.

1 Introduction

Animated pedagogical agents are often designed as tutors [1] or peers [2] in virtual learning environments. In these roles, the agent typically works alongside the learner to solve problems, hold conversations, and provide guidance. Recently, intelligent agents have expanded their roles to become the *object* of practice. That is, it is the interaction with the agent that is intended to be educational. For example, virtual humans [3] have been used to provide practice for intercultural communication [4-5], clinical interviewing [6-7], police officer training [8], and healthy play for children with autism [9], to name only a few examples.

In each of these cases, the virtual human acts as a role player in some social interaction with the learner. The primary goal is to simulate specific communicative patterns (verbal and nonverbal) in realistic ways to give the learner a chance to assess what they see and hear, then respond, all within a social context. Live role playing exercises have a long history in education [10] and there is strong evidence to believe that learners naturally interact with virtual humans as if they are real [11-12]. In this

paper, we present the results of a small research project to investigate the use of virtual humans for teaching deception detection. We describe two virtual humans that exhibit common traits of truth-telling and deception along with an accompanying tutoring system for teaching diagnostic cues to deception. We also present results from an initial evaluation of the system's effectiveness.

2 Deception Detection

How does one detect the difference between the truth and a lie? The ability to detect deception is a critical skill for a number of professions, including school administrators, reporters, therapists, and law enforcement officers. The results of an investigative interview or interrogation of non-cooperative suspects can have profound consequences on society. Unfortunately, research shows that lie detection is extremely difficult, and individuals tend to perform only slightly above chance levels [13]. Even law enforcement officials, who routinely encounter deception in their daily work and receive training in this task, perform similarly at chance levels [14-15].

It is possible, however, for training to improve one's ability to detect deception by helping learners identify the right *cues* on which to focus. Cues to deception are often divided into one of three categories – verbal, nonverbal, and vocal. *Verbal cues* are cues that come from the content of the speaker's statement (e.g., admitted lack of memory, textual embedding, self-references). *Nonverbal cues* can be observed solely by the behavior of the individual (e.g., eye contact, posture, hand/arm movements). Finally, *vocal cues* are behaviors that are related to speech production (e.g., speech hesitations, pitch of voice, response latency). DePaulo et al. [16] conducted a meta-analytic review of over 150 cues hypothesized to be related to deceptive or truthful statements from 116 research studies. Their results identified 23 cues with large effect sizes for significant differences between liars and truth tellers - of these 23 cues, 21 involved verbal or vocal cues to deception (e.g., level of detail, spontaneous corrections, admitted lack of detail, negative statements). On the other hand, many of the nonverbal cues, including those commonly believed by lay persons and law enforcement to be diagnostic (e.g., eye contact, posture, and blinking), were unrelated to deception.

In a meta-analytic review of 11 training studies (20 comparisons) in the deception detection literature, Frank and Feeley [17] determined that overall training showed a minimal effect (4% increase) in improving performance; however, the authors argued that reasons for the weak and inconsistent results may lie within the research designs and stimulus materials (e.g., relevance of the deception detection task, adequacy of the training materials, appropriateness of pre- and post-test). In a more recent meta-analytic review of the training literature, Hauch, Sporer, Michael, and Meissner [18] examined the effect of training on verbal, vocal, and non-verbal cues to deception 22 published and 8 unpublished studies involving $N = 3,638$ participants. Overall, there was no significant effect of training found for vocal cues ($d = .11$), while small effects were found for both training on nonverbal cues ($d = .18$) and the combined training of nonverbal and vocal ($d = .21$). In contrast, a robust, medium-sized effect was found for training involving verbal cues ($d = .62$).

3 Deceptive Virtual Humans

Most training programs for deception detection involve lecture-based seminars, recorded videos, role-playing with peers, and group discussion [19]. A frequently missing element in these approaches is the opportunity for realistic practice for investigative interviewing skills, which provides the context for eliciting and assessing interviewee responses that reveal deception when it is present. To explore the feasibility of using virtual humans for this purpose, two characters were developed to exhibit common traits related to truthfulness and deceit. Character data was augmented with information about deception cues and tutorial feedback. This allowed the creation of a simple tutoring system that supports the learner in asking the right questions and identifying common properties of truth-telling and deceit. In this section, we describe the development of the characters and implementation of the tutoring system.

3.1 Character Design

The first step in designing characters was to decide on incidents that would provide the backdrop for the interviews. For this, we chose two common (but serious) law enforcement situations: a bombing and a shooting. A character for each scenario was created (see figure 1): Victor, who was in the area during the bombing, and Amber, who witnessed the shooting. Both characters have the ability to be truthful or deceitful in their responses, thus providing four distinct practice opportunities. Both characters use basic nonverbal behaviors in their utterances and idol behaviors, but these are not intelligently modeled as distracters or indicators of truth or deceit in this version of the system.



Fig. 1. Virtual humans Victor and Amber can be truthful or deceptive in their responses. Art assets are variants of those used in the virtual patients project [6].

To build characters we relied primarily on the Tactical Questioning domain editor [20]. For other aspects (i.e., animation, speech generation, nonverbal behaviors) we leveraged the Virtual Human Toolkit.¹ The domain editor requires an author to create

¹ <http://vhtoolkit.ict.usc.edu/>

a set of domain objects that the character knows about with attributes for each object. Figure 2 shows the authoring tool for Amber: the author created the domain object *incident* with attributes such as *witnesses* and *feelings*. The domain editor automatically generates basic speech acts for these attributes, which correspond to what the character can say about them (and what he or she might want to keep secret, as shown in the upper right corner of the screenshot by the true and false values).

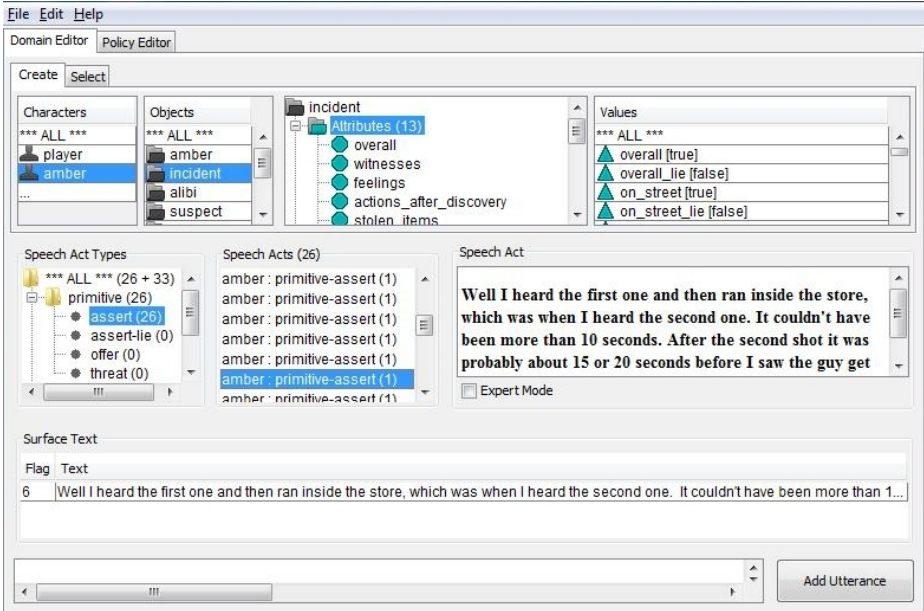


Fig. 2. Screenshot of the domain editor for Amber

In Victor's domain model includes objects and attributes related to the women's clinic and the bombing; thus, *clinic* as an object, along with an attribute *location*. The system automatically generates dialogue acts that enable him to answer questions such as "Where did the bombing occur?", however, to support robust natural language understanding, authors should provide a variety of surface forms for the questions users may ask. So, other examples for the location attribute would be "Where did the bomb go off?" and "Where did the bombing happen?" At run-time, the system uses a statistical matching algorithm to match the user's questions to those in the character's knowledge base. The matching is done in such a way that exact matches are not required to the set of pre-authored questions [21]. Below we show how the similarity measures from the classifier are used during learner interactions with the characters.

Victor and Amber each have two versions of approximately 70 responses available to answer learner questions. They can assume either a truth-telling or deceptive mode for a given session, thus providing 4 different experiences available for use during training. Utterances were carefully crafted to consist of diagnostic verbal cues based on the content. Table 1 shows several examples from Victor's utterance library along with some of the most prominent cues:

Table 1. Example statements from Victor and their underlying deception cues

Truthful	Verbal cues
T1. We were walking down Whitman road and I remember hearing a few cars screeching to stop in the middle of the street. There were a fare number of people around. A few started to run towards the building...	cognitive states, level of detail
T2. Umm, I can't remember all the details of what I have heard... I do remember them talking about "causing damage" a few times, but I could have easily heard wrong.	forthcoming, ordinary imperfections
Deceitful	
D1. I think there were a few cars that just stopped in the middle of the road.	uncertainty
D2. Well I was arrested once for just some minor offense. But, you know, that was just kids getting into mischief. I don't think that that's on my record anymore.	distancing

- In T1, Victor brings up his state of mind during the event (cognitive states) along with a high level of detail. Both are suggestive of truth telling.
- In T2, he is honest about his lack of memory (forthcoming) and suggests that he may have remembered wrong (ordinary imperfections). Both are common indicators of truth-telling.
- D1 is the deceptive version of T1. "I think" indicates uncertainty, which is often present in deceptive statements.
- Victor distances himself in D2 by suggesting the incident is no longer on his record. Even though not directly related to the event, it still suggests deception.

A primary goal of deception training is to help the learner identify such cues in utterances, and understand what they imply for the deception judgment. Although nonverbal behaviors also play a critical role in deception, both as indicators and distracters (e.g., nervousness is not a reliable sign of deception [16]), our prototype currently uses them to a limited degree.

3.2 Interaction

For learners to pose questions of the characters, we chose to use a typed interface along with suggested question matches. Although automated speech recognition and understanding would increase fidelity, our focus was primarily on helping the learner (1) ask the best questions (the skill of investigative interviewing) and (2) identify diagnostic cues (the skill of deception detection). Thus, we decided a typed interface with animated and oral responses from the characters would be sufficient. In addition, to reduce the frustration of asking a question that was not recognized, we provide the learner with the top five matches from the statistical matcher for the student to select (see figure 2). A dialogue history was also maintained in a window below the question asking area.

Enter your question here:

What did you see this morning?

Send

Get a Hint

Select the best match for your question:

- Tell me about what you witnessed this morning.
- Tell me about yourself.
- What do you know about the man you suspect?
- Where were you this morning at the time of the bombing?
- Who do you remember seeing around the clinic this morning?

Send

Fig. 3. Learners type in their question for the character then select the best match

3.3 Guidance and Feedback

A simple tutoring system provides help to the learner during their interview with the character. The system responds to help requests from the learner by clicking on the hint button shown in figure 3. It also provides unsolicited feedback on the first occurrence of every deception cue when they are present in character responses (about ¼ of all utterances do not contain cues). As mentioned, the utterances were authored and tagged with their relevant cues (see table 1). In addition, the utterances are also tagged with with phases indicating when particular questions should be asked. When a hint is requested, the current phase of the interview is consulted in order to suggest an appropriate action. The phases are: (1) greetings, (2) background on character, (3) information about the incident, (4) identifying responsible party, and (5) closing. Hints are associated with each phase. Feedback about cues present in character utterances is based on the tags described earlier that describe properties of the utterance and how they relate to the character's veracity. Table 2 shows several examples of tutor messages. All tutor messages are delivered via a pop-up window and they require that the student click to OK to close them.

Table 2. Examples of tutor utterances

Tutor hints	Relevant Phase
You may want to ask the individual about certain beliefs he has that might give you an idea of how he feels about the incident.	background
If he was involved, he probably would have been busy for the last day or so, not just during the time of the explosion. You should ask about this.	alibi
Tutor feedback	Relevant Cue
People telling the truth will often use a greater level of detail in their descriptions.	level of detail
Admitted lack of memory for details actually occurs more commonly in truthful statements	ordinary imperfections

4 Method

The objective of the current study was to determine the utility of two deception detection training programs that differed with respect to the training approach (rather than the content of the training materials). The performance of these two training approaches was also compared with a no-training control condition. Participants completed both a pre- and post-test assessment of their deception detection performance on Days 1 and 5. Day 3 involved interaction with the training program for those in the training conditions, or completion of an innocuous (irrelevant) task for those in the control condition. Given research on the validity of various cues to deception [16], training focused on the most valid verbal and vocal cues to deception.

4.1 Participants

One hundred and five undergraduate psychology students from the University of Texas at El Paso participated in exchange for credit in their introductory psychology courses.

4.2 Design

The experiment employed a 3 x 2 mixed factor design in which participants were randomly assigned to one of two training conditions or a no-training control condition (between-subject factor, $N = 35$ per condition), while all participants completed both a pre- and post-test assessing their deception detection accuracy (repeated measure).

Videotaped stimuli. Videotaped alibi statements were collected by the authors for the purposes of developing a pre- and post-test measure of deception detection accuracy. Individuals were randomly assigned to conditions in which they were instructed to provide either a truthful or deceptive account of their whereabouts three nights prior to the interview. Participants were interviewed on video regarding their statement. Interviews ranged from approximately one and a half minutes to seven minutes ($M = 2.47$ minutes). Forty-two videos (21 truthful and 21 deceitful) were collected in total. Videos were then pilot tested by a separate group of participants ($N = 18$), and 20 videos were selected for use in the pre- and post-test (10 deceitful, 10 truthful). The pre- and post-test stimuli were shown to have equivalent discriminability, and presentation of the stimuli were randomized and counterbalanced across participants.

Training programs. To contrast with the Virtual Human based system described above (VHuman), a non-interactive, didactic presentation was created (Didactic). This program involved training on the same cues to deception, but presented the information through a less immersive format. In the presentation, each cue to deception is explained via both text and auditory information. Following each explanation, a video example of the cue is shown. Statements were taken from the scenarios developed for the VHuman condition to provide comparable exposure. The presentation of cues was automated so that time on task would be comparable across participants. Participants in the VHuman condition were also presented with an abbreviated didactic presentation prior to interviewing the virtual characters that provided participants with a very basic introduction

to the various cues to deception (excluding any video depictions of the cues). Learning and exposure to the cues were then subsequently reinforced by the VHuman system as each participant interacted with both characters who were randomly assigned to lie or tell the truth regarding their given scenario. Time on task for both the Didactic and VHuman training conditions were comparable (i.e., approx. 40 min).

4.3 Procedure

The experiment was conducted across three sessions involving a pre-test, a training session, and a post-test, with sessions separated by 48 hours. The pre- and post-test sessions involved presentation of 10 videotaped statements (described previously) for which participants had to determine veracity (i.e., truth vs. lie) and provide a confidence estimate (i.e., 0 to 100% scale). Following each deception detection task, participants also completed a questionnaire assessing their knowledge of the relative value of various cues to deception – this measure served as an assessment of content learning for the two training conditions. Participants were randomly assigned to one of the two training conditions, or completed an innocuous filler task (i.e., participants completed an unrelated face recognition study in which they studied a series of faces and were later tested in their recognition performance) in the control condition.

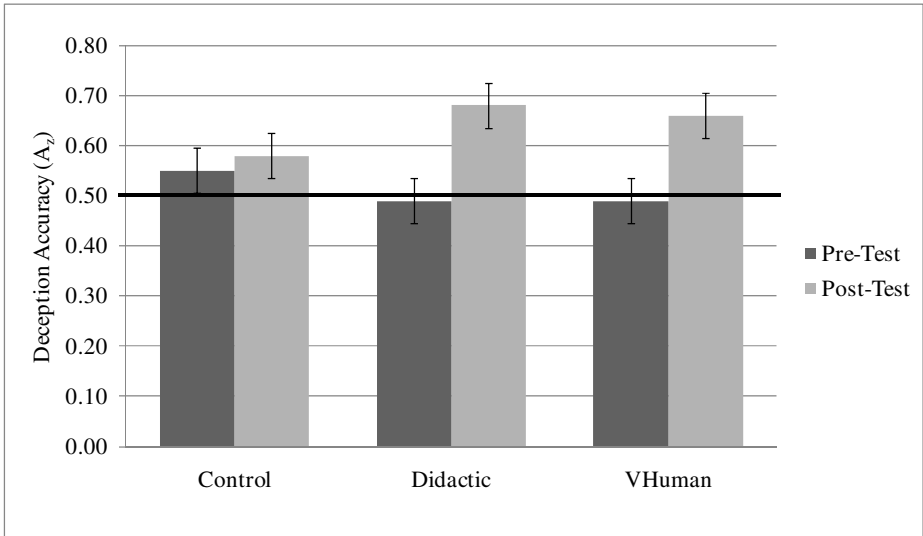


Fig. 4. Deception detection performance (A_z) as a function of training condition across pre-test and post-test. Bolded line represents “chance” performance.

5 Results

A measure of discrimination accuracy was computed via signal detection theory [22]. Specifically, A_z was computed via the following formula:

$$A_z = (d' / \sqrt{2})$$

A_z provides an estimate of discrimination accuracy that ranges from 0 to 1, with .50 being equivalent to chance performance in the present study. A_z is computed as the difference between the Z-score for the “hit” (i.e., proportion of deceptive statements correctly identified) and “false alarm” (i.e., proportion of truthful statements incorrectly identified as deceptive) estimates. Results of the experiment are displayed in Figure 3. A 3 x 2 mixed Analysis of Variance (ANOVA) was used to assess the influence of training across the pre- and post-tests. A significant interaction was observed, $F(2,102) = 3.31, p < .05, \eta_p^2 = 0.06$. Follow-up pairwise comparisons indicated that both the Didactic, $t(34) = 3.88, p < .001, d = 0.88$, and the VHuman, $t(34) = 3.37, p < .01, d = 0.65$, training conditions significantly improved detection performance. In contrast, the control condition showed no significant learning effect across the pre- and post-test, $t(34) = 0.39, ns., d = 0.08$.

Participants estimates of confidence were similarly analyzed using a 3 x 2 mixed ANOVA, resulting in a significant interaction, $F(2,102) = 3.37, p < .05, \eta_p^2 = 0.06$. Pairwise comparisons demonstrated that both the Didactic, $t(34) = 2.10, p < .05, d = 0.27$, and the VHuman, $t(34) = 4.98, p < .001, d = 0.66$, conditions significantly increased their confidence as a function of training, while the control condition showed no significant effect, $t(34) = 1.19, ns., d = 0.13$.

Finally, we also assessed participants’ learning of the validity of various cues to deception as a function of training condition. A 3 x 2 mixed ANOVA demonstrated a significant interaction, $F(2,102) = 18.28, p < .001, \eta_p^2 = 0.26$. Once again, pairwise comparisons showed that both the Didactic, $t(34) = 8.08, p < .001, d = 1.95$, and the VHuman, $t(34) = 7.34, p < .001, d = 1.83$, conditions significantly improved their knowledge of diagnostic cues to deception, while the control condition showed no significant learning effect, $t(34) = 1.78, ns., d = 0.29$.

6 Conclusions and Discussion

Based on these results, it appears that both forms of training had an equivalently positive effect on ability to detect deception when compared to a baseline control (with no training). This suggests that teaching learners about cues is an effective method for enhancing their ability to detect deception. It also suggests there is no apparent value of interactive practice with feedback for the skill of detecting deception in recorded statements.

It is common practice in deception studies to use recorded statements as pre- and post-tests. They are passive experiences that are easy to repeat and compare since participants simply watch the video and make a decision. In this study, the pre- and post-tests are similar in structure to the didactic condition (which consisted of a 40 minute presentation on cues to deception with recorded examples), just without the instructional content. This may explain why the didactic condition was sufficient to achieve a significant learning gain. In many ways, the didactic condition has higher fidelity than the VHuman condition since recorded statements show real people, using intonation, facial expressions, and so on. Even though virtual humans also simulate these aspects of human communication, in our prototype, they were identical between truth-tellers and deceptive versions of Victor and Amber. Given this, our results can

even be considered positive since virtual human-based training was able to produce equivalent learning to a very strong didactic condition closely aligned to the test.

A weakness of using recorded videos is that they only tap recognition skill; they do not evaluate a learner's ability to conduct an investigative interview. This represents a difference between the training conditions in our study. Specifically, learners who interacted with virtual humans were required to generate questions for the characters (assuming they didn't game the system). They also received hints on what to ask about, if requested. If a positive difference is to be found between conditions, it may be revealed in having learners conduct an independent interview.

The virtual characters and tutoring system were developed in about 4 months, and so there are many opportunities for improving and extending it. Perhaps most importantly, as virtual human technology matures, it will become easier to simulate human behavior with higher fidelity which will enable our system to address a greater range of novice misconceptions. For example, nervousness is commonly interpreted as a sign of deception when it is, in fact, not a reliable indicator [16]. A nervous, but truthful, virtual human would provide an interactive example to demonstrate that there are often many causes of nervousness, such as being asked questions. Beyond this, Victor and Amber fall into the category of question / answer virtual humans, meaning they do not possess a realistic model of the interaction with the user beyond the simple phase markers used by the tutor (see section 3.2). Although sufficient for some learning goals, to tap more deeply into investigative interview process, learning would likely benefit from models of emotions, proxemics, and consequences of what is being said (some of these aspects appear in other virtual human research [3, 12]).

In summary, virtual human technology is in the early stages of being applied to the problem of teaching social interaction skills. Our prototype system and evaluation suggests a virtual human-based system can increase learners' deception detection skills. The question of whether it is necessary, and if there are other benefits beyond detection, depends on new measures of investigative interviewing skill and richer models of virtual human behavior.

Acknowledgements

The project or effort described here has been sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred. We extend special thanks to Patrick Kenny for sharing art assets from the virtual patient project at ICT and technical support from Sudeep Gandhe and Jina Lee.

References

1. Johnson, W.L., Rickel, J., Lester, J.C.: Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments. *Int. J. Art. Int. in Ed.* 11, 47–48 (2000)
2. Kim, Y., Baylor, A.: A Social-Cognitive Framework for Pedagogical Agents as Learning Companions. *Ed. Tech. Res. and Dev.* 54, 569–596 (2006)

3. Swartout, W., Gratch, J., Hill, R.W., Hovy, E., Marsella, S., Rickel, J., Traum, D.: Toward virtual humans. *AI Magazine* 27, 96–108 (2006)
4. Kim, J.M., Hill, R.W., Durlach, P.J., Lane, H.C., Forbell, E., Core, M., Marsella, S., Pynadath, D.V., Hart, J.: BiLAT: A Game-based environment for practicing negotiation in a cultural context. *Int. J. Art. Int.* in Ed. (in press)
5. Johnson, W.L., Valente, A.: Tactical language and culture training systems: Using artificial intelligence to teach foreign languages and cultures. In: *Proc. 20th Nat. Conf. on Inn. App. Art. Int.*, vol. 3, pp. 1632–1639. AAAI Press, Chicago (2008)
6. Kenny, P., Parsons, T., Gratch, J., Rizzo, A.: Virtual humans for assisted health care. In: *Proc. 1st Int. Conf. on Pervasive Technology Related to Assistive Environments*, pp. 1–4. ACM Press, New York (2008)
7. Johnsen, K., Raij, A., Stevens, A., Lind, D.S., Lok, B.: The validity of a virtual human experience for interpersonal skills education. In: *Proceedings SIGCHI Conf. on Human Factors in Comp. Sys.*, pp. 1049–1058. ACM Press, New York (2007)
8. Hubal, R.C., Frank, G.A., Guinn, C.I.: Lessons learned in modeling schizophrenic and depressed responsive virtual humans for training. In: *Proc. 8th Int. Conf. on Int. User Interfaces*, pp. 58–92. ACM Press, New York (2003)
9. Tartaro, A., Cassell, J.: Playing with virtual peers: bootstrapping contingent discourse in children with autism. In: *Proc. 8th Int. Conf. Learning Sciences*, vol. 2, pp. 382–389 (2008)
10. Kane, P.E.: Role playing for educational use. *Comm. Ed.* 11, 320–323 (1964)
11. Reeves, B., Nass, C.: *The media equation: How people treat computers, television, and new media like real people and places.* Cambridge University Press, Cambridge (1996)
12. Gratch, J., Wang, N., Gerten, J., Fast, E., Duffy, R.: Creating rapport with virtual agents. In: Pelachaud, C., Martin, J.-C., André, E., Chollet, G., Karpouzis, K., Pelé, D. (eds.) *IVA 2007. LNCS (LNAI)*, vol. 4722, pp. 125–138. Springer, Heidelberg (2007)
13. Bond, C.F., DePaulo, B.M.: Accuracy of deception judgments. *Personality & Social Psychology Review* 10, 214–234 (2006)
14. Meissner, C.A., Kassin, S.M.: He's guilty!: Investigator bias in judgments of truth and deception. *Law & Human Behavior* 26, 469–480 (2002)
15. Kassin, S.M., Meissner, C.A., Norwick, R.J.: I'd know a false confession if I saw one: A comparative study of college students and police investigators. *Law & Human Behavior* 29, 211–227 (2005)
16. DePaulo, B.M., Lindsay, J.J., Malone, B.E., Muhlenbruck, L., Charlton, K., Cooper, H.: Cues to deception. *Psychological Bulletin* 129, 74–118 (2003)
17. Frank, M.G., Feeley, T.H.: To catch a liar: Challenges for research in lie detection training. *J. Applied Comm. Res.* 31, 58–75 (2003)
18. Hauch, V., Sporer, S.L., Michael, S.W., Meissner, C.A.: Does training improve detection of deception? A meta-analysis (2009) (unpublished manuscript)
19. Colwell, L.H., Miller, H.A., Lyons Jr., P.M., Miller, R.S.: *The Training of Law Enforcement Officers in Detecting Deception: A Survey of Current Practices and Suggestions for Improving Accuracy.* *Police Quarterly* 9, 275–290 (2006)
20. Gandhe, S., DuVault, D., Roque, A., Martinovski, B., Artstein, R., Leuski, A., Gerten, J., Traum, D.: From domain specification to virtual humans: An integrated approach to authoring tactical questioning characters. *Interspeech* (2008)
21. Leuski, A., Kennedy, B., Patel, R., Traum, D.: Asking questions to limited domain virtual characters: How good does speech recognition have to be? In: *Proc. 25th Army Science Conf.* (2006)
22. Wickens, T.: *Elementary Signal Detection Theory.* Oxford University Press, New York (2001)

Optimizing Story-Based Learning: An Investigation of Student Narrative Profiles

Seung Y. Lee, Bradford W. Mott, and James C. Lester

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

Abstract. Narrative-centered learning environments offer significant potential for creating effective learning experiences in which students actively participate in engaging story-based problem solving. As the capabilities of narrative-centered learning environments expand, a key challenge is identifying experiential factors that contribute to the most effective story-based learning. To investigate the impact of students' narrative experiences on learning outcomes, a Wizard-of-Oz (WOZ) study was conducted with middle school students interacting with a narrative-centered learning environment. Students' experiences were examined using narrative profiles representing their type of story interaction. With narrative planning, tutorial planning, and natural language dialogue functionalities provided by wizards, the WOZ study revealed that in interactive story-based learning supported by beyond-state-of-the-art ITS capabilities, 1) students exhibit a range of learning outcomes, 2) students exhibit a range of narrative profiles, and 3) certain student narrative profiles are strongly associated with desirable learning outcomes. The study suggests design decisions for optimizing story-based learning.

Keywords: Narrative-Centered Learning Environments, Game-Based Learning Environments, Wizard-of-Oz Study.

1 Introduction

Stories provide an episodic structure that shapes our experience. By taking advantage of narrative's inherent structure, narrative-centered learning environments offer significant potential for creating story-based learning that is both effective and engaging [1,2]. These environments offer rich interactions in which students actively participate in engaging story-based problem solving tailored to their individual needs. A growing body of research has investigated narrative-centered learning environments for education and training. For example, it has been shown that narrative-centered learning environments can support science education [3,4], social behavior education [5], interactive health education [6], and training [7,8,9].

Although the capabilities of narrative-centered learning environments have expanded greatly over the last decade [3,10,11], these technologies are still in their infancy. Given the promise that narrative-centered learning environments have shown, it is important to identify the factors that contribute to the most effective

story-based learning experiences. Further, this analysis should be conducted without current ITS technology limitations to explore how an “ideal” narrative-centered learning environment should best support learning. A promising approach for this line of investigation is the Wizard-of-Oz (WOZ) methodology. A WOZ-enabled narrative-centered learning environment could be devised in which wizards provide the narrative planning, tutorial planning, and natural language dialogue functionalities of the system to ensure human-level decision making and interactivity are achieved.

This paper presents the results of a Wizard-of-Oz study conducted with middle school students collaborating with trained wizards in a WOZ-enabled narrative-centered learning environment for microbiology. It was found that students exhibited positive learning outcomes. Analysis of the students’ experiences revealed that in interactive story-based learning supported by beyond-state-of-the-art ITS capabilities, students exhibited a range of narrative profiles, and certain student narrative profiles are strongly associated with desirable learning outcomes. The study suggests design decisions for optimizing story-based learning.

This paper is structured as follows. Section 2 provides background and related work on story-based learning. Section 3 introduces the CRYSTAL ISLAND learning environment. The study design and procedure are described in Section 4, and the results and analysis are presented in Section 5. Section 6 discusses the findings and associated design implications, and Section 7 offers concluding remarks and suggests directions for future work.

2 Background

Educators have long recognized the potential of contextualizing learning within narrative [12]. Leveraging students’ innate metacognitive apparatus for understanding and crafting stories to create story-based learning experiences offers much promise. By immersing learners in captivating worlds populated by compelling characters, narrative-centered learning environments can enable learners to participate in the construction of narratives, to engage in active problem solving, and to reflect on narrative experiences [1].

A broad range of techniques have been proposed for crafting interactive story-based learning experiences that are both engaging and pedagogically effective. In FearNot!, a simulation framework drives affect-enabled agents to generate dramatic vignettes for social behavior education [5,13]. By suggesting coping behaviors for virtual agents involved in bullying incidents, students develop an empathetic relationship with the agents. In Teatrix, a director agent supports the story creation process of students collaboratively creating fairy tales [14]. In Carmen’s Bright IDEAS, an agent-based approach to interactive narrative is utilized to teach social problem-solving skills to mothers of pediatric cancer patients [6]. In the Tactical Language and Culture Training System, virtual characters teach foreign language communication skills [8,15]. Socially intelligent virtual humans are used in the Stability and Support Operations (SASO) system to develop leadership and negotiation skills in trainees [16,17,18].

Despite the promising work that has been carried out to date on narrative-centered learning environments, the capabilities of these systems remain limited when

compared to human-to-human interactions. We seek a better understanding of the factors that contribute to effective story-based learning without the limitations of current ITS technologies.

3 The CRYSTAL ISLAND Learning Environment

The classic narratological framework for analyzing story structure is the narrative arc (Figure 1). The *narrative arc* models the tension experienced by the audience as a narrative progresses through its phases of exposition, complication, escalation, climax, and resolution. In the *exposition*, the setting and situation are introduced. During the *complication*, a problem develops and tension rises. The *escalation* sees the problem intensify and a rapid rise in the tension. The tension reaches its highest level during the *climax* when the story starts to resolve itself. During the *resolution* the remaining issues are resolved and the tension diminishes.

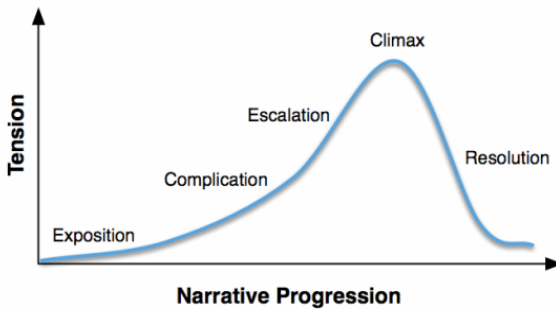


Fig. 1. A prototypical narrative arc

By examining how a student progresses through the narrative arc during an interactive story-based learning session, we can discover her *narrative profile*, which captures the relative time spent in each phase of the narrative. More formally, we define a *narrative profile* for student s as follows: Given total session time T_s for student s , an ordered sequence of n phases of the narrative arc, p_1, p_2, \dots, p_n , and a function $f_s(x)$ returning the session time of phase x for student s , the *narrative profile* for student s is:

$$\text{narrative profile}_s = \left\langle \frac{f_s(p_1)}{T_s}, \frac{f_s(p_2)}{T_s}, \dots, \frac{f_s(p_n)}{T_s} \right\rangle$$

Narrative profiles offer a window into students' narrative experiences within story-based learning environments. They provide an analytical tool for systematically investigating which types of narrative experiences are more conducive to learning.

We have been investigating narrative profiles with a narrative-centered learning environment built with Valve Corporation's Source™ engine, the technology behind Half Life® 2 and other popular computer games. CRYSTAL ISLAND (Figure 2) features a science mystery set on a recently discovered tropical island where a research station

has been established to study the island's unique flora and fauna. Underlying the science mystery is a curriculum derived from the North Carolina standard course of study for eighth-grade microbiology. Within the story, students play the role of the protagonist who is attempting to discover the identity and source of an infectious disease plaguing the research station.



Fig. 2. WOZ-enabled CRYSTAL ISLAND learning environment

The CRYSTAL ISLAND story opens by introducing the student to the island and members of the research team for which the protagonist's father serves as the lead scientist. Several members of the research team have fallen gravely ill, including the protagonist's father. It is the student's task to discover the cause and source of the outbreak. Throughout the mystery, the student is free to explore the world and interact with other characters while forming questions, generating hypotheses, collecting data, and testing hypotheses. The student can pick up and manipulate objects, view posters, operate lab equipment, and converse with non-player characters to gather clues about the source of the disease. During the course of solving the mystery, the student uses an in-game *diagnosis worksheet* to organize her thoughts regarding the patients' symptoms, the likelihood of potential diseases (based on their expected symptoms, incubation period, and transmission source), and her final diagnosis. Upon completing the diagnosis worksheet, the student verifies its contents with the camp nurse and develops a treatment plan for the sickened CRYSTAL ISLAND researchers.

To investigate the impact of students' narrative experiences on learning outcomes without the limitations imposed by current ITS technologies, a WOZ-enabled version of CRYSTAL ISLAND was developed. In the WOZ-enabled CRYSTAL ISLAND learning environment, a wizard provides narrative planning, tutorial planning, and natural language dialogue functionalities. The wizard assumes the role of the camp nurse and collaboratively works with the student to solve the science mystery. Together in the virtual environment they carry on rich conversations using voice chat and observe one another's actions (e.g., picking up objects, gazing at objects and non-player characters, operating testing equipment) while performing problem-solving activities (Figure 2).¹ In addition to attending to the navigation, spoken communication, and manipulation behaviors of the nurse's character in the virtual environment, the wizard guides the student's inquiry activities and controls the progression of the story through the narrative arc. To support these activities, the wizard's display includes

¹ The facial expressions of the characters were not synchronized with the communication between the student and wizard via voice chat.

detailed information regarding students' activities in the environment (e.g., reading books, testing objects, updating the diagnosis worksheet), as well as access to a narrative dashboard. The *narrative dashboard* enables the wizard to initiate key narrative events in the environment (e.g., introducing new patient symptoms, having a non-player character bring in additional items for testing).

In addition to the wizard functionalities, the learning environment was modified to focus on the rich interactions between the student and wizard as well as to reduce the time spent navigating through the environment. This was accomplished by confining the learning scenario to a single virtual building on the island that houses both the camp's infirmary and laboratory. Within this environment the student and wizard gain access to all of the materials needed to solve the science mystery (e.g., sickened researchers, background books and posters, potential sources of the disease, lab equipment). The learning scenario, student and wizard controls, and wizard display were refined throughout a series of pilot studies with college students prior to the study reported in this paper.

To illustrate the behavior of the WOZ-enabled CRYSTAL ISLAND, consider the following scenario. A student has been collaborating with the nurse character, whose behaviors are orchestrated by the wizard. The student has learned that an infectious disease is an illness that can be transmitted from one organism to another, often through food or water. Under guidance from the nurse, the student has examined the patients' symptoms and conducted lab tests on food and water items. Through this exploration, the student has come to believe that the source of the illness is a water-borne disease and that it is likely cholera or shigellosis. Although she believes cholera is more likely, she is unable to arrive at a final diagnosis. Through her conversation with the nurse character, "Yeah, hum, well, they both can come from water, but cholera is mostly water, I believe," the wizard determines that the student is having difficulty ruling out shigellosis and decides that this is an opportune moment to introduce a narrative event. The wizard uses the narrative dashboard and activates the *Observe Leg Cramps Symptom* plot point, which results in one of the patients moaning loudly in the infirmary. The student examines the patient, updates her diagnosis worksheet with the new information, and informs the wizard, "He has leg cramps." The student decides to consult the reference material regarding disease symptoms and says, "Ok, I am going to check the disease symptoms again." After checking a virtual book, the student exclaims, "That means it is cholera." The wizard asks the student to update her diagnosis worksheet with her new hypothesis and explain why she believes this. The student then provides a detailed explanation justifying her diagnosis and the nurse congratulates the student for successfully solving the science mystery.

4 Study Design and Procedure

Using a wizard protocol that was designed to maximize learning gains in story-based interactions, a study was conducted with middle school students.

4.1 Participants

The participants in the study included 33 students (15 males and 18 females) of various ages, race, and ethnicity. Thirteen of the participants were eliminated due to

incomplete data on either the pre-test or post-test, leaving 10 males and 10 females. Approximately 3% of the students were American Indian or Alaska Native, 3% were Asian, 24% were Black or African American, 9% were Hispanic or Latino, 55% were Caucasian, and 6% were of other races. Participants were all eighth-grade students from a North Carolina public school ranging in age from 13 to 15 ($M = 13.79$, $SD = 0.65$). Prior to receiving the instruments, tests, and intervention of this study, the students had completed the microbiology curriculum mandated by the North Carolina standard course of study. Two wizards assisted with the study, one male and one female. Prior to the study, wizards underwent extensive training and participated in pilot studies.

4.2 Participant Procedure

Students entered the study room having completed a ten question pre-test approximately one week prior to the intervention. Upon arriving, students were greeted by a researcher and instructed to review a set of CRYSTAL ISLAND instructional handouts, including information on the CRYSTAL ISLAND back-story, task description, characters, and controls. Upon completing their review of the handouts, the researcher provided further direction to the students on the use of the keyboard and mouse controls. The researcher then informed the students that they would be collaborating with another human-controlled character, the camp nurse, in the learning environment to solve the science mystery. Students were encouraged to freely communicate with the camp nurse using voice chat throughout their learning sessions. The students and wizards were physically located in different rooms. Finally, the researcher answered any questions from the students, informed them that the sessions were being videotaped, instructed them to put on their headsets and position their microphones, and asked them to direct all future communication to the camp nurse. The researcher remained in the room for the duration of their session. The CRYSTAL ISLAND session concluded once the student and wizard agreed on a final diagnosis. Immediately after reaching agreement, students exited the CRYSTAL ISLAND learning environment and completed the post-test (which consisted of the same items as the pre-test). The post-test was completed by the students within 20 minutes. In total, the students' sessions lasted no more than 60 minutes.

4.3 Wizard Protocol

To improve the consistency of the wizards' tutorial planning, narrative planning, and natural language dialogue activities, a protocol was iteratively developed and refined through a series of pilot studies. The resulting protocol included a high-level procedure for the wizard to follow (e.g., introduce yourself as the camp nurse, describe the patient situation to the student, review the scientific method with the student), a set of interaction guidelines (e.g., collaboratively work with the student to solve the mystery, organize the student's activities around the scientific method, act as a senior peer to the student, encourage the student to explain her conclusions and ensure they are logical and consistent with the available data, engage the student in constant face-to-face inquiry dialogue), and a set of narrative guidelines (e.g., overall story structure,

ordering constraints between narrative events, appropriate situations to introduce narrative events, pacing advice to help ensure sessions complete on time).

Prior to the study with the eighth grade students, each wizard received training on the CRYSTAL ISLAND microbiology curriculum and the materials to be provided to students during the study. The wizard training included information on key concepts from the CRYSTAL ISLAND curriculum and the protocol to follow. After carefully reviewing the materials over the course of a week and having all of their questions answered, the wizards participated in at least three (and up to four) training sessions with college students. After each training session, a researcher performed an “after action review” with the wizard to discuss her interactions with the student and adherence to the wizard protocol (both from a tutorial perspective and narrative perspective).

5 Results and Analysis

To investigate narrative profiles as they relate to learning outcomes, the students’ CRYSTAL ISLAND sessions were analyzed using a five-phase narrative arc. The narrative phases used in the analysis were defined by the classic *exposition*, *complication*, *escalation*, *climax*, and *resolution* plot structure inspired by Freytag’s pyramid [19]. To compute the narrative profiles for students, the time they spent in each phase of the narrative arc was automatically calculated using event timestamps from behavior traces recorded during their learning session.

Table 1. Percentage of time spent in each phase of the narrative arc

Narrative Phase	Cluster A		Cluster B	
	Mean	SD	Mean	SD
<i>Exposition</i>	11	2	10	2
<i>Complication</i>	8	1	10	3
<i>Escalation</i>	23	6	42	7
<i>Climax</i>	39	7	19	6
<i>Resolution</i>	20	5	20	9

Employing an unsupervised learning method, students were partitioned into groups using their narrative profiles as the observation vectors. Two groups, *Cluster A* and *Cluster B*, were identified utilizing k-means clustering, containing 9 and 11 student narrative profiles, respectively. Table 1 lists the percentage of time spent in each phase of the narrative arc for both clusters. The difference in time spent between clusters was statistically significant during both the *escalation* ($t = 6.733$, $p < 0.0001$) and *climax* ($t = 7.176$, $p < 0.0001$) phases. In short, these two clusters group together the students whose narrative profiles are most similar to one another with respect to time spent in each phase of the narrative arc and they are significantly different.

5.1 Results

It was found that students’ CRYSTAL ISLAND interactions yielded positive learning outcomes. Students exhibited learning gains ($M = 2.20$, $SD = 1.58$) as measured by the difference of their post-test ($M = 8.05$, $SD = 1.57$) and pre-test scores ($M = 5.85$,

$SD = 1.27$). A matched pairs t-test between post-test and pre-test scores indicates that the learning gains were significant, $t(19) = 6.24, p < 0.0001$. There were learning gains within both Cluster A ($M = 2.91, SD = 1.45$) and Cluster B ($M = 1.33, SD = 1.32$). A matched pairs t-test shows that the learning gains for both clusters were significant (Cluster A: $t(8) = 6.67, p < 0.0001$, Cluster B: $t(10) = 3.02, p = 0.01$). There was no significant difference between the pre-test scores of the clusters, $t(19) = 1.19, p = 0.12$. In addition, adjusting for pre-test scores using ANCOVA, the learning gains for both clusters were significantly different, $F(2, 18) = 4.25, p = 0.03$. Thus students in Cluster A achieved higher learning gains than students in Cluster B.

5.2 Analysis

To analyze the student narrative profiles in more detail, each phase of the students' narrative profile was further decomposed into the percentage of time spent on a given interaction mode (Figure 3). The interaction modes used in the analysis were *data collection* (e.g., examining patients, testing food items), *science reading* (e.g., studying disease books, reviewing scientific method poster), and *inquiry* (e.g., updating and discussing the diagnosis worksheet) activities. By inspecting the percentage of total session time spent in each of these interaction modes during the narrative phases we gain insight into the students' experiences as they relate to learning outcomes.

First, for *data collection* activities, differences between the clusters were found to be significant during the *complication* ($t = 1.78, p = 0.05$) and *climax* ($t = 3.56, p = 0.002$) phases of the narrative. Second, for *science reading* activities, significant differences were found between the clusters during the *escalation* ($t = 4.73, p = 0.0002$) and *climax* ($t = 2.88, p = 0.005$) phases of the narrative. Finally, for *inquiry* activities, differences between the clusters were found to be significant during all phases of the narrative except for the *resolution*. The differences showed

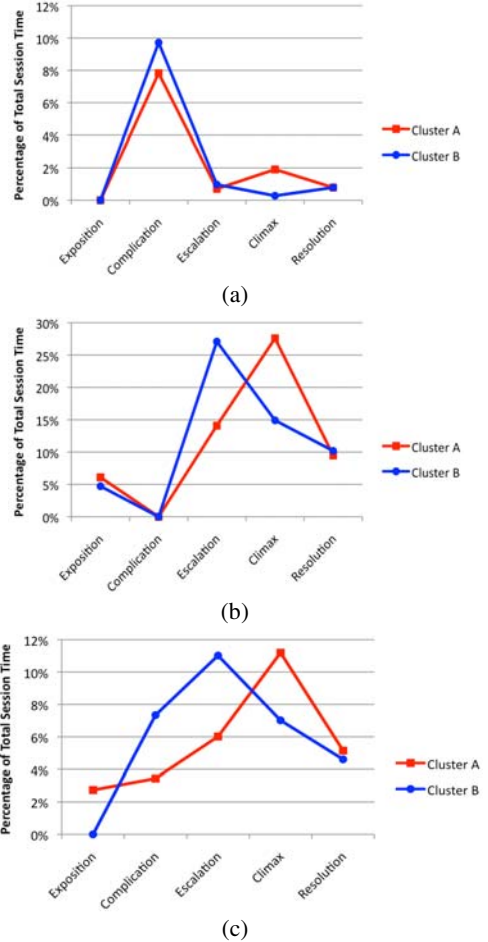


Fig. 3. Mean percentage of total session time spent on (a) data collection activities, (b) science reading activities, and (c) inquiry activities during each phase of the narrative arc

statistical significance for *climax* ($t = 2.39, p = 0.03$), while *exposition* ($t = 4.04, t = 0.001$), *complication* ($t = 5.05, p < 0.0001$) and *escalation* ($t = 3.36, p = 0.003$) showed strong statistical significance.

6 Discussion and Design Implications

The study found that students interacting with the WOZ-enabled CRYSTAL ISLAND narrative-centered learning environment achieved significant learning gains. Through the analysis it was found that students had a range of narrative experiences, which were captured by the differences in their narrative profiles. After clustering the narrative profiles it was found that students within each cluster achieved significant learning gains, and, notably, one cluster was found to have significantly outperformed the other with respect to learning gains.

Further analysis showed that students within the two groups utilized their time in each phase of the narrative arc for different activities. Overall, students in the higher performing group spent more time on *data collection*, *science reading*, and *inquiry* activities during the *climax* phase of the narrative than students in the lower performing group. Correspondingly, students in the lower performing group devoted more of their time to these activities during the *complication* and *escalation* phases of the narrative. A qualitative exploration of the student and wizard interactions during the *climax* phase of the narrative revealed that the higher performing group tended to be more actively engaged with the wizard during this phase of the narrative. They were much more likely to provide detailed explanations of their hypotheses and support them with relevant facts from books and results from their lab testing activities.

Although the analysis provides insight into how students' narrative experiences relate to learning outcomes, it does not pinpoint the learner activities that are most responsible for learning gains. For example, the learning gains might be caused by the self-explanation effect since the higher performing students seem to spend more time reasoning about and explaining their hypotheses to the wizard. Similarly, lower performing students might be experiencing higher cognitive load during the complication and escalation phases of the narrative since the problem space is more open-ended at these points in the story. Additional investigation needs to be conducted to understand what learner activities are contributing the most toward learning gains.

The study suggests design implications for optimizing story-based learning. First, narrative event representations should include metadata for encoding temporal attributes of student activities (e.g., durations, ratios) to support reasoning about narrative structure and interaction modes as they bear on learning outcomes. Second, narrative planners should be designed to reason about the interaction modes associated with desirable learning gains for each phase of the narrative. This capability would allow narrative planners to emphasize the interaction modes that are most promising given the current situation. Third, narrative-centered tutorial planners should be able to craft the structure of stories and scaffold student interactions to most effectively balance their time in each phase of the narrative. This capability would allow narrative planners to appropriately guide students to the next phase of the narrative arc at the most opportune moment.

7 Conclusion

Narrative-centered learning environments offer significant potential for creating effective learning experiences. Identifying factors that contribute to effective story-based learning is critically important in optimizing these experiences. To this end, a Wizard-of-Oz study was conducted with students interacting with a narrative-centered learning environment. It was found that students exhibited significant learning gains and that partitioning students by narrative profiles resulted in clusters with significantly different learning gains. Furthermore, a detailed analysis of the activities performed by students in each phase of the narrative arc revealed that the clusters differ significantly with respect to interaction modes.

The narrative profile technique introduced in this paper represents a first step towards developing a clearer understanding of student learning in narrative-centered learning environments. Future work will use a narrative lens to investigate techniques for incorporating more detailed student behavior trace data, annotations of dialogue, tutorial strategies, virtual world behaviors, and affective feedback to further refine design principles for optimizing story-based learning.

Acknowledgments. The authors wish to thank members of the IntelliMedia Group for their assistance, Omer Sturlovich and Pavel Turzo for use of their 3D models, and Valve Corporation for access to the Source™ engine and SDK. Special thanks to Joe Grafsgaard, Kate Lester, Jen Robison and Jon Rowe for assisting with the study and data analysis. This research was supported by the National Science Foundation under Grants REC-0632450 and 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.

References

1. Mott, B., Callaway, C., Zettlemoyer, L., Lee, S., Lester, J.: Towards Narrative-Centered Learning Environments. In: AAAI Fall Symposium on Narrative Intelligence, Cape Cod, MA, pp. 78–82 (1999)
2. McQuiggan, S., Rowe, J., Lee, S., Lester, J.: Story-Based Learning: The Impact of Narrative on Learning Experiences and Outcomes. In: 9th International Conference on Intelligent Tutoring System, Montreal, Canada, pp. 239–249 (2008)
3. Mott, B., Lester, J.: Narrative-Centered Tutorial Planning for Inquiry-Based Learning Environments. In: 8th International Conference on Intelligent Tutoring System, Jhongli, Taiwan, pp. 675–684 (2006)
4. Rowe, J., Mott, B., McQuiggan, S., Robison, J., Lee, S., Lester, J.: Crystal Island: A Narrative-Centered Learning Environment for Eighth Grade Microbiology. In: Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, Brighton, U.K., pp. 11–20 (2009)
5. Aylett, R., Louchart, S., Dias, J., Paiva, A., Vala, M.: FearNot! – An Experiment in Emergent Narrative. In: 5th International Conference on Intelligent Virtual Agents, Kos, Greece, pp. 305–316 (2005)

6. Marsella, S., Johnson, W.L., Catherine, L.: Interactive Pedagogical Drama for Health Interventions. In: 11th International Conference on Artificial Intelligence in Education, Sydney, Australia (2003)
7. Wang, N., Johnson, L.: The Politeness Effect in an Intelligent Foreign Language Tutoring System. In: 9th International Conference on Intelligent Tutoring System, Montreal, Canada, pp. 270–280 (2008)
8. Johnson, L.: Serious Use of a Serious Game for Language Learning. In: 13th International Conference on Artificial Intelligence in Education, Marina del Ray, CA, pp. 67–74 (2007)
9. McAlinden, R., Gordon, A., Lane, C., Pynadath, D.: UrbanSim: A Game-Based Simulation for Counterinsurgency and Stability-Focused Operations. In: Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, Brighton, U.K, pp. 41–50 (2009)
10. Si, M., Marsella, S., Pynadath, D.: Thespian: Modeling Socially Normative Behavior in a Decision-Theoretic Framework. In: 6th International Conference on Intelligent Virtual Agents, Marina del Ray, CA, pp. 369–382 (2006)
11. Core, M., Lane, H., van Lent, M., Gomboc, D., Solomon, S., Rosenberg, M.: Building Explainable Artificial Intelligence Systems. In: 18th Conference on Innovative Applications of Artificial Intelligence, Boston, MA (2006)
12. Wells, C.: *The Meaning Makers: Children Learning Language and Using Language to Learn*, Heinemann, Portsmouth, NH (1986)
13. Vala, M., Raimundo, G., Sequeira, P., Cuba, P., Prada, R., Martinho, C., Paiva, A.: ION Framework – A Simulation Environment for Worlds with Virtual Agents. In: 9th International Conference on Intelligent Virtual Agents, Amsterdam, Netherlands, pp. 418–424 (2009)
14. Machado, I., Brna, P., Paiva, A.: Learning by Playing: Supporting and Guiding Story-Creation Activities. In: Moore, J., Redfield, C., Johnson, W. (eds.) 10th International Conference on Artificial Intelligence in Education, Amsterdam, Netherlands, pp. 334–342 (2001)
15. Johnson, L., Wu, S.: Assessing Aptitude for Learning with a Serious Game for Foreign Language and Culture. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 520–529. Springer, Heidelberg (2008)
16. Traum, D., Marsella, S., Gratch, J., Lee, J., Hartholt, A.: Multi-party, Multi-issue, Multi-strategy Negotiation for Multi-modal Virtual Agents. In: 8th International Conference on Intelligent Virtual Agents, Tokyo, Japan, pp. 117–130 (2008)
17. Gratch, J., Marsella, S.: A Domain-independent Framework for Modeling Emotion. *Journal of Cognitive Systems Research* 5(4), 269–306 (2004)
18. Gratch, J., Wang, N., Gerten, J., Fast, E., Duffy, R.: Creating Rapport with Virtual Agents. In: 7th International Conference on Intelligent Virtual Agents, Paris, France, pp. 125–138 (2007)
19. Freytag, G.: *Technique of the Drama: An Exposition of Dramatic Composition an Art*. In: Macewan, E. (trans.) B. Blom, New York (1968)

Integrating Learning and Engagement in Narrative-Centered Learning Environments

Jonathan P. Rowe^{*}, Lucy R. Shores, Bradford W. Mott, and James C. Lester

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

Abstract. A key promise of narrative-centered learning environments is the ability to make learning engaging. However, there is concern that learning and engagement may be at odds in these game-based learning environments and traditional learning systems. This view suggests that, on the one hand, students interacting with a game-based learning environment may be engaged but unlikely to learn, while on the other hand, traditional learning technologies may promote deep learning but provide limited engagement. This paper presents findings from a study with human participants that challenges the view that engagement and learning need be opposed. A study was conducted with 153 middle school students interacting with a narrative-centered learning environment. Rather than finding an oppositional relationship between learning and engagement, the study found a strong positive relationship between learning outcomes and increased engagement. Furthermore, the relationship between learning outcomes and engagement held even when controlling for students' background knowledge and game-playing experience.

Keywords: Narrative-Centered Learning Environments, Game-Based Learning, Engagement, Situational Interest, Presence.

1 Introduction

Narrative-centered learning environments show significant potential for providing engaging learning experiences that are tailored to individual students. By leveraging the motivational characteristics of narrative and games, along with the adaptive pedagogy of intelligent tutoring systems, narrative-centered learning environments offer a promising platform for students to acquire enhanced problem solving, strategic and analytical thinking, decision making, and other twenty-first century skills [1,2]. As an active and growing area of research, narrative-centered learning environments are under investigation in a range of domains, including language learning [3], anti-bullying education [4], and middle school science [5].

Despite the ITS community's growing interest in narrative-centered learning environments, there is concern that the narrative and gameplay elements of these systems may not contribute to improved learning outcomes. This belief stems in part from a

^{*} Corresponding author.

view that gains in engagement achieved by a narrative-centered learning environment are primarily diversionary [6,7]. The view suggests that while students may become engaged in the rich virtual environments or compelling characters provided by many narrative-centered learning environments, the reasons for engagement are tangential to learning [8]. In this view, there is a tradeoff between learning and engagement, suggesting that on the one hand, students interacting with a game-based learning environment may be engaged but unlikely to learn, and on the other hand, traditional learning technologies may promote deep learning but provide limited engagement [7,9,10].

This paper challenges the above view by presenting findings from an empirical study investigating the relationship between learning and engagement in a narrative-centered learning environment. This work assesses engagement by considering a number of factors hypothesized to be associated with engagement, including presence, situational interest, avoidance of “gaming the system,” and problem-solving efficiency. Findings are presented from a study with 153 eighth-grade students interacting with CRYSTAL ISLAND, a narrative-centered learning environment for middle school microbiology. Results show that students who experienced higher levels of engagement during interactions with the CRYSTAL ISLAND environment achieved improved learning outcomes. Notably, this result is independent of students’ prior microbiology knowledge and gaming experience.

2 Background

Narrative-centered learning environments embed educational content and activities in story-centric, problem-solving scenarios and interactive virtual worlds. Multi-user virtual environments such as Quest Atlantis [11] and River City [5] use rich narrative settings to contextualize inquiry-based science learning scenarios with strong social and collaborative elements. Other work has utilized interactive narrative generation and agent behavior planning to foster adaptive narrative experiences that are pedagogically effective and tailored to individual students [4,12,13]. A key motivation for this line of work is the development of systems that simultaneously promote deep learning and high engagement.

For years, devising techniques for detecting and measuring student engagement has been an important area of investigation within the ITS community [14,15,16]. A number of techniques have been proposed to assess related factors such as student motivation [14,17] and affective states such as flow [18]. Other work has sought to devise automated models for detecting symptoms of disengagement, namely, off-task behavior [16,19]. One of the most salient examples of off-task behavior is “gaming the system,” where students exploit elements of a learning environment interface to progress through a lesson without having mastered the associated content [19].

Engagement in narrative-centered learning environments can take several forms, including engagement in the learning scenario and engagement in tangential or aesthetic elements of the virtual environment [8]. Narrative-centered learning environments often provide vast interactive environments, realistic physics, and engaging characters, which may risk introducing *seductive details* into learning experiences

[20]. Seductive details have the potential to distract, disrupt, or divert students' attention from pedagogical objectives and to reduce students' time-on-task. To adequately investigate the complex nature of engagement in narrative-centered learning environments, assessments of engagement should consider a variety of factors. For example, students' problem-solving efficiency within the virtual environment is likely an indication of engagement, as well as resistance to seductive details. Off-task behavior such as "gaming the system" can be viewed as evidence of disengagement from a learning environment. In addition to these factors, we hypothesize that students' situational interest in a narrative-centered learning experience, as well as their sense of presence in the narrative environment, are likely contributors to engagement.

Situational interest is characterized by varying lengths of concentrated attention coupled with affective reaction activated during a particular time period by certain environmental stimuli [21,22]. Studies have shown that situational interest directed towards an instructional task can influence cognitive performance [23] and facilitate deeper learning [24]. Also, learning tasks and environments that yield significant situational interest have been shown to benefit students who have previously been disengaged in similar learning activities [25]. However, situational interest is not exclusive to learning tasks; game design and adaptive scaffolding should encourage interest in on-task actions, rather than interest in purely aesthetic or gameplay features of narrative-centered learning environments [8].

Presence contributes to the goal of transparency in technology-mediated interactions [26]. Although there has been substantial debate on formal definitions, there is a general consensus that *presence* describes a user's sense of "being there" when interacting with a mediated environment [27,28]. Presence has been alternatively defined as "the subjective experience of being in one place or environment, even when one is physically situated in another" [29]. It is related to students' sense of transportation into a story, which is an important contributor to the engaging quality of narratives. Presence is distinguished from related concepts such as immersion and involvement. *Immersion* generally refers to the extent and nature of technology-provided sensory stimuli; it is often associated with the pervasiveness and fidelity of visual, auditory, olfactory, and tactile inputs [28]. *Involvement* refers to the degree of attention and meaning devoted to some set of stimuli [29].

3 CRYSTAL ISLAND

Now in its third major iteration, CRYSTAL ISLAND (Figure 1) is a narrative-centered learning environment built on Valve Software's Source™ engine, the 3D game platform for Half-Life 2. The curriculum underlying CRYSTAL ISLAND's mystery narrative is derived from the North Carolina state standard course of study for eighth-grade microbiology. The environment is designed as a supplement to classroom instruction. Students play the role of the protagonist, Alyx, who is attempting to discover the identity and source of an infectious disease plaguing a newly established research station. Several of the team's members have fallen gravely ill, and it is the student's task to discover the nature and cause of the outbreak.



Fig. 1. CRYSTAL ISLAND narrative-centered learning environment

CRYSTAL ISLAND's narrative takes place in a small research camp situated on a recently discovered tropical island. As students explore the camp, they investigate the island's spreading illness by forming questions, generating hypotheses, collecting data, and testing hypotheses. Throughout their investigations, students interact with virtual characters offering clues and relevant microbiology facts via multimodal "dialogues" delivered through student menu choices and characters' spoken language. The dialogues' content is supplemented by virtual books, posters, and other resources encountered in several of the camp's locations. As students gather useful information, they have access to a personal digital assistant to take and review notes, consult a microbiology field manual, communicate with characters, and report progress in solving the mystery. To solve the mystery, students complete a *diagnosis worksheet* to manage their working hypotheses and record findings about patients' symptoms and medical history, as well as any findings from tests conducted in the camp's laboratory. Once a student enters a hypothesized diagnosis, cause of illness, and treatment plan into her diagnosis worksheet, the findings are submitted to the camp nurse for review and possible revision.

To illustrate the behavior of CRYSTAL ISLAND, consider the following scenario. Suppose a student has been interacting with non-player characters in the storyworld and learning about infectious diseases. In the course of having members of the research team become ill, she has learned that a pathogen is an illness that can be transmitted from one organism to another. As she concludes her introduction to infectious diseases, she learns from the camp nurse that the mystery illness seems to be coming from food items the sick members recently ate. Some of the island's characters are able to help identify food items and symptoms that are relevant to the

scenario, while others provide helpful microbiology information. The student is careful to take notes recording information about bacteria and viruses in her personal digital assistant, and corroborates these notes with information contained in her microbiology field manual. After forming several hypotheses about which food items may be sickening the team members, the student discovers through a series of tests that a container of unpasteurized milk in the dining hall is contaminated with bacteria. By combining this information with her knowledge about the characters' symptoms and recent dining habits, the student infers that the disease is *E. coli*, for which ample rest is the best immediate treatment plan. She records her findings in a diagnosis worksheet, and submits them to the camp nurse for review and implementation.

4 Empirical Study

An experiment involving human participants was conducted with the entire eighth grade population of a North Carolina middle school. The primary goal of the experiment was to investigate the impact of different scaffolding techniques on learning and engagement in the CRYSTAL ISLAND narrative-centered learning environment. However, no condition effects were observed for either learning or engagement. This paper's findings come from a secondary analysis of the data, which considers the experiment's conditions as a whole.

4.1 Participants

A total of 153 eighth grade students ranging in age from 12 to 15 ($M = 13.3$, $SD = 0.48$) interacted with the CRYSTAL ISLAND environment during the study. Three of the participants were eliminated due to incomplete data. Among the remaining students, 80 were male and 70 were female. Approximately 3% of the participants were American Indian or Alaska Native, 2% were Asian, 32% were African American, 13% were Hispanic or Latino, and 50% were White. Although CRYSTAL ISLAND is ultimately intended to be used concurrently with classroom coverage of an associated microbiology unit, scheduling issues necessitated that the study be conducted prior to students being exposed to the microbiology curriculum unit of the North Carolina state standard course of study in their regular classes.

4.2 Materials and Apparatus

Students completed an online demographic survey and CRYSTAL ISLAND curriculum test prior to the intervention. The curriculum test consisted of 16 multiple-choice questions created by an interdisciplinary team of researchers. The test consisted of eight factual and eight application questions assessing students' knowledge of pathogens, select diseases, and the scientific method.

Post-experiment materials were completed immediately following the CRYSTAL ISLAND intervention. Included in these materials were the same curriculum test used in the pre-experiment, a variation of the Perceived Interest Questionnaire [30], and the Presence Questionnaire [29]. The interest scale was adapted from measures used by Schraw to examine within-subject relationships with learning outcomes [30]. The measure consists of ten Likert items measuring students' situational interest related to

CRYSTAL ISLAND. To illustrate the scale, example items include the following: “I got absorbed playing CRYSTAL ISLAND without trying to,” and “CRYSTAL ISLAND really grabbed my attention.” The Presence Questionnaire (PQ) is a validated measure containing several subscales, including involvement/control, naturalism of experience and quality of interface [29]. The natural subscale is intended to assess the student’s perception of the virtual environment’s consistency with reality, in terms of locomotion and nature of the interaction. The interface quality subscale indicates how seamlessly the control and display devices are integrated into the interactive experience. Example items include the following: “How compelling was your sense of moving around inside the virtual environment,” “How much did your experiences in the virtual environment seem consistent with your real-world experiences,” and “How much did the visual display quality interfere or distract you from performing assigned tasks or required activities?”

In addition to pre- and post-experiment subjective measures, the CRYSTAL ISLAND software calculated a numerical score to assess students’ progress and efficiency in completing the science mystery. Students could view their scores in the upper left corner of their screens throughout their interactions with the software. The score consisted of a weighted sum of gameplay sub-scores, and incorporated time taken to accomplish important goals, students’ ability to demonstrate microbiology content knowledge, and evidence of careful hypothesis formulation. Students were penalized for any attempt to “game the system” by repeatedly submitting incorrect diagnoses to the camp nurse or guessing on content knowledge quizzes. Details of the score’s calculation are shown in Table 1. As an objective measure assessing students’ understanding of the curricular content and performance at completing the CRYSTAL ISLAND mystery, students’ final score is treated as a measure to investigate engagement alongside subjective measures of presence and situational interest.

4.3 Participant Procedure

Participants entered the experiment room having completed the majority of pre-test materials one week prior to the intervention. Students were initially provided general details about the CRYSTAL ISLAND mystery and game controls during an introductory presentation by a researcher. After the presentation, students completed the remaining pre-test materials and received several CRYSTAL ISLAND supplementary documents. These materials consisted of a CRYSTAL ISLAND backstory and task description, a character handout, a map of the island, and an explanation of the game’s controls.

Participants were given 60 minutes to work on solving the mystery. Solving the mystery consisted of several objectives including: learning about pathogens, viruses, and bacteria; compiling the symptoms and recent history of the sick researchers; recording details about diseases believed to be potentially afflicting the team members; testing a variety of possible sources for the disease; and reporting the solution—including cause, source, and treatment—to the camp nurse. Immediately after solving CRYSTAL ISLAND’s science mystery, or 60 minutes of interaction, participants completed the post-experiment questionnaires. Completion of post-experiment materials took no longer than 30 minutes for participants. In total, sessions lasted up to 120 minutes.

Table 1. Point values for calculation of final game score

Action	Points (pts)
Overall Mystery Solution	
Correct Solution	500 pts
Solution Efficiency	(7500 / elapsed time) pts
Incorrect Solution Attempt	-100 pts
In-game Quiz Questions	
First Attempt Correct	25 pts
Second Attempt Correct	10 pts
Second Attempt Incorrect	-10 pts
Object Contaminant Testing	
Test Milk for Pathogens	200 pts
Incorrect Object	-10 pts
Incorrect Contaminant	-25 pts
Character Interactions	
Talk to Kim	(25 / elapsed time) pts
Talk to Teresa	(50 / elapsed time) pts
Talk to Ford	(125 / elapsed time) pts
Talk to Robert	(125 / elapsed time) pts
Talk to Quentin	(125 / elapsed time) pts
Pathogen Labeling Activities	
Correct Answer	10 pts
Incorrect Answer	-10 pts
Total Maximum Points	≈ 1665 pts

5 Results

An investigation of learning found that on average, students answered 2.35 ($SD = 2.75$) more questions correctly on the post-test than they did on the pre-test. Matched pairs t-tests (comparing post-test to pre-test scores) indicated that students' learning gains were significant, $t(149) = 10.49$, $p < .001$.

5.1 Learning and Engagement

Examining factors believed to reflect engagement and students' understanding of the curriculum, Pearson correlations indicated significant relationships between microbiology background knowledge and presence, $r = .17$, $p < .05$, and final score, $r = .28$, $p < .01$. Similar relationships were found between microbiology post-test scores and presence, $r = .295$, $p < .01$, final score, $r = .445$, $p < .01$, and situational interest, $r = .239$, $p < .01$. To more closely investigate the relationships between learning and engagement, additional analyses controlling for background knowledge were conducted.

A partial correlation controlling for pre-test score found significant relationships between microbiology post-test scores and two of our engagement measures, presence, $r = .25$, $p < .01$, and final game score, $r = .38$, $p < .01$. The same type of analysis also found a borderline significant relationship between situational interest and post-test score, $r = .15$, $p < 0.1$. Offering further evidence for a connection between learning and engagement in CRYSTAL ISLAND, a linear regression indicated that

microbiology background knowledge, presence, and final score were all significant predictors of performance on the microbiology post-test, and the model as a whole was significant, $R^2 = .33$, $F(3, 143) = 23.46$, $p < .001$.

As a supplement to these findings, further analyses were conducted to determine whether similar relationships held for the involved/control subscale of the Presence Questionnaire, which provides a more specific measure of involvement in the environment. A partial correlation controlling for microbiology background knowledge revealed significant relationships between the involved/control subscale and final score, $r = .376$, $p < .01$, situational interest, $r = .181$, $p < .05$, and microbiology post-test performance, $r = .334$, $p < .01$.

5.2 Engagement and Individual Differences

Additional analyses were conducted to determine whether particular subpopulations experienced different levels of engagement while interacting with the CRYSTAL ISLAND environment. Pearson correlations indicated significant relationships between game-playing frequency and presence, $r = .269$, $p < .01$, as well as between self-perceived game-playing skill and presence, $r = .178$, $p < .05$. Game-playing frequency was found to have a significant relationship with the PQ's involved/control subscale, $r = .327$, $p < .01$, as did game-playing skill, $r = .211$, $p < .05$. A significant relationship between game-playing frequency and the PQ's natural subscale was observed, $r = .17$, $p < .05$. No significant relationships were found between game-playing frequency and the PQ's interface quality subscale, nor between game-playing skill and the naturalism of experience or interface quality subscales. No significant correlation was found between either of the game-playing demographics and situational interest, or between either of the game-playing demographics and final game score.

A regression analysis was conducted to examine the simultaneous contributions of game-playing frequency, microbiology background knowledge, presence, and final score on microbiology post-test scores. The overall model was significant, $R^2 = .327$, $F(4, 136) = 16.535$, $p < .01$, but only microbiology background knowledge, presence, and final score were significant predictors of post-test performance, not game-playing frequency. A similar regression analysis was conducted to examine the contributions of self-assessed game-playing skill, microbiology background knowledge, presence, and final score on microbiology post-test scores. The overall model was significant, $R^2 = .33$, $F(4, 136) = 16.750$, $p < .01$, but again only microbiology background knowledge, presence, and final score were significant predictors, not self-assessed game-playing skill.

Examining gender, an independent samples t-test analyzing the relationship between gender and presence found that males tended to feel more present in the environment than females, $t(139) = 3.01$, $p < .01$. Similar results were found for the involved/control subscale of the Presence Questionnaire: an independent samples t-test analyzing the relationship between gender and the involved/control measure found that males tended to feel significantly more involved/control when interacting with CRYSTAL ISLAND than females, $t(140) = 2.96$, $p < .01$. Males also tended to rate the interface quality more highly, $t(140) = 1.97$, $p < .01$, but no gender effect was found on the PQ's natural subscale. Table 2 displays raw scores, by gender, for each of the content knowledge, situational interest, and presence measures.

Table 2. Raw scores by gender on content knowledge, situational interest, and presence questionnaires

Group	Microbiology Pre-Test	Microbiology Post-Test	Situational Interest	Overall Presence	Presence Subscales		
					Involved / Control	Natural	Interface Quality
Males	6.37 (2.23)	8.60 (3.03)	31.8 (8.73)	89.5 (16.4)	53.1 (10.4)	13.5 (3.44)	13.6 (3.45)
Females	6.31 (1.77)	8.62 (2.94)	31.4 (8.37)	82.3 (15.6)	48.0 (9.99)	12.5 (3.47)	12.6 (2.37)
Total	6.34 (2.02)	8.61 (2.98)	31.6 (8.54)	86.2 (16.4)	50.8 (10.5)	13.0 (3.48)	13.2 (3.04)

Significant differences were observed between genders for gaming demographics. Males reported significantly higher ratings for self-perceived game-playing skill, $F(1, 143) = 57.49$, $p < .001$, and reported playing games more frequently, $F(1, 143) = 60.15$, $p < .001$, than females. Although males tended to feel more present in CRYSTAL ISLAND, an analysis of covariance controlling for game-playing frequency found no significant effect of gender on presence, $F(1, 138) = 2.01$, $p = .158$. Significant differences were not found between genders for situational interest or final score.

A linear regression considering only the female population yielded a significant model for predicting microbiology post-test performance, $R^2 = .25$, $F(2, 62) = 10.12$, $p < .01$, but only microbiology background knowledge and final score were significant predictors, not presence.

6 Discussion

The findings indicate that student engagement with the CRYSTAL ISLAND environment was associated with improved learning outcomes. Results showed a significant relationship between students' pre-test scores and presence, as well as between pre-test scores and final game scores. This suggests that students who demonstrated greater prior content knowledge tended to become more engaged with the narrative environment. However, all three measures for engagement—presence, situational interest, and final game score—were found to be significantly associated with post-test score, independent of pre-test score. These findings suggest that students who were more engaged with the CRYSTAL ISLAND narrative environment tended to experience greater learning gains, regardless of prior knowledge. The findings contrast with perspectives that place engagement and learning at odds with one another in narrative-centered learning environments. Further, analyses found no relationships between game-playing experience and learning. This finding suggests that both gamers and non-gamers who were engaged in the narrative-centered learning experience achieved improved learning outcomes. Students can be productively engaged in a narrative-centered learning environment, and this relationship is independent of prior knowledge or game-playing experience.

The findings suggest that engagement and learning need not be at odds in narrative-centered learning environments, and may in fact reinforce one another. We hypothesize that well-designed story and gameplay elements may contribute to this synergistic relationship. However, poorly designed story and gameplay elements may

detract from both engagement and learning by introducing seductive details and promoting off-task behavior. Additional investigation is needed to determine which elements of narrative-centered learning environments are most closely associated with learning and engagement. These efforts will contribute to the development of models to automatically detect student engagement and learning during narrative-centered learning interactions.

Interesting findings were also observed concerning the effects of gender and game-playing experience on presence. Males tended to be more present during CRYSTAL ISLAND interactions than females. An initial interpretation might be that the game was better designed for males than females. However, a significant correlation was also observed between presence and game-playing experience. Furthermore, males tended to have significantly greater game-playing experience than females. An ANCOVA suggested that game-playing experience, not gender, may be the more predominant factor associated with presence. These findings raise important questions about the effective design of narrative-centered learning environments for males and females, as well as gamers and non-gamers. However, additional investigation is necessary to better understand these relationships.

Extending studies of narrative-centered learning interactions beyond individual sessions is an essential next step for understanding the relationship between engagement and learning in narrative-centered learning environments. Studies spanning multiple sessions, along with in-class integration, are important to assess how engagement can be sustained over time with narrative-centered learning environments, how long-term engagement is related to deep learning and transfer, and whether engagement can impact student attitudes and self-efficacy. To accommodate these larger scale studies, devising additional subjective and objective measures for engagement beyond those used in this work will also be important.

7 Conclusions

Narrative-centered learning environments offer a promising vehicle for delivering experiences that are both effective and engaging. To investigate the hypothesis that learning and engagement need not be in opposition in narrative-centered learning environments, an empirical study was conducted with middle school students interacting with the CRYSTAL ISLAND learning environment. It was found that increased engagement was associated with improved learning outcomes, independent of students' prior content knowledge or game-playing experience. As narrative-centered learning environments mature, it will become increasingly important to understand how students can most effectively interact with them, and what role narrative and game features can play in scaffolding learning and realizing sustained engagement.

Acknowledgements

The authors wish to thank members of the IntelliMedia Group of North Carolina State University for their assistance, Omer Sturlovich and Pavel Turzo for use of their 3D model libraries, and Valve Software for access to the Source™ engine and SDK. This

research was supported by the National Science Foundation under Grants REC-0632450, IIS-0757535, DRL-0822200, IIS-0812291, and CNS-0540523. 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.

References

1. Gee, J.P.: *What Video Games Have to Teach Us about Learning and Literacy*. Palgrave Macmillan, New York (2003)
2. Shaffer, D.W.: *How Computer Games Help Children Learn*. Palgrave Macmillan, New York (2006)
3. Johnson, W.L.: Serious use of a Serious Game for Language Learning. In: 13th International Conference on Artificial Intelligence in Education, pp. 67–74 (2007)
4. Aylett, R., Louchart, S., Dias, J., Paiva, A., Vala, M.: FearNot! An Experiment in Emergent Narrative. In: 5th International Conference on Intelligent Virtual Agents, pp. 305–316 (2005)
5. Ketelhut, D.: The Impact of Student Self-Efficacy on Scientific Inquiry Skills: An Exploratory Investigation in River City, a Multi-User Virtual Environment. *Journal of Science Education and Technology* 16, 99–111 (2007)
6. Hallinen, N., Walker, E., Wylie, R., Ogan, A., Jones, C.: I was playing when I learned: A Narrative Game for French Aspectual Distinctions. In: *Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education*, pp. 117–120 (2008)
7. Rai, D., Heffernan, N., Gobert, J., Beck, J.: Mily's World: Math game involving authentic activities in visual cover story. In: *Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education*, pp. 125–128 (2009)
8. Rowe, J., McQuiggan, S., Robison, J., Lester, J.: Off-Task Behavior in Narrative-Centered Learning Environments. In: 14th International Conference on Artificial Intelligence and Education, pp. 99–106 (2009)
9. Prensky, M.: *Digital Game-based Learning*. McGraw-Hill, New York (2001)
10. McNamara, D., Jackson, G., Graesser, A.: Intelligent tutoring and games (ITaG). In: *Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education*, pp. 1–10 (2009)
11. Barab, S., Dodge, T., Tuzun, H., et al.: The Quest Atlantis Project: A Socially-Responsive Play Space for Learning. In: Shelton, B.E., Wiley, D. (eds.) *The Educational Design and Use of Simulation Computer Games*, pp. 159–186. Sense Publishers, Rotterdam (2007)
12. Si, M., Marsella, S., Pynadath, D.: THESPIAN: An Architecture for Interactive Pedagogical Drama. In: 12th International Conference on Artificial Intelligence in Education, pp. 595–602 (2005)
13. Traum, D., Marsella, S., Gratch, J., Lee, J., Hartholt, A.: Multi-Party, Multi-Issue, Multi-Strategy Negotiation for Multi-Modal Virtual Agents. In: 8th International Conference on Intelligent Virtual Agents, pp. 117–130 (2008)
14. Vicente, A., Pain, H.: Informing the Detection of the Student's Motivational State: An Empirical Study. In: 6th International Conference on Intelligent Tutoring Systems, pp. 933–943 (2002)
15. Beck, J.: Engagement Tracing: Using Response Times to Model Student Disengagement. In: 12th International Conference on Artificial Intelligence in Education, pp. 88–95 (2005)

16. Walonoski, J., Heffernan, N.: Detection and Analysis of Off-Task Gaming Behavior in Intelligent Tutoring Systems. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 382–391. Springer, Heidelberg (2006)
17. Johns, J., Woolf, B.: A Dynamic Mixture Model to Detect Student Motivation and Proficiency. In: 21st National Conference on Artificial Intelligence, pp. 163–168 (2006)
18. D’Mello, S., Picard, R., Graesser, A.: Towards an Affect-Sensitive AutoTutor. Special issue on Intelligent Educational Systems – IEEE Intelligent Systems 22, 53–61 (2007)
19. Baker, R., Corbett, A., Koedinger, K., Wagner, A.: Off-Task Behavior in the Cognitive Tutor Classroom: When Students “Game the System”. In: ACM Conference on Human Factors in Computing Systems, pp. 383–390 (2004)
20. Harp, S., Mayer, R.: How Seductive Details Do Their Damage: A Theory of Cognitive Interest in Science Learning. *Journal of Educational Psychology* 90(3), 414–434 (1998)
21. Hidi, S.: Interest and its Contribution as a Mental Resource for Learning. *Review of Educational Research* 60, 549–571 (1990)
22. Ainley, M., Hidi, S., Berndorff, D.: Interest, Learning, and the Psychological Processes that Mediate their Relationship. *Journal of Educational Psychology* 94, 545–561 (2002)
23. Schiefele, U.: Topic Interest, Text Representation, and Quality of Experience. *Contemporary Educational Psychology* 21, 3–18 (1996)
24. Wade, S., Buxton, W., Kelly, M.: Using Think-Alouds to Examine Reader-Text Interest. *Reading Research Quarterly* 34, 194–216 (1999)
25. Hidi, S., Harackiewicz, J.: Motivating the Academically Unmotivated: A Critical Issue for the 21st Century. *Review of Educational Research* 70, 151–179 (2000)
26. Norman, D.: *The Invisible Computer*. MIT Press, Cambridge (1998)
27. Kim, Y., Baylor, A.: A Social-Cognitive Framework for Pedagogical Agents as Learning Companions. *Educational Technology Research & Development* 54, 569–590 (2006)
28. Schubert, T., Friedmann, F., Regenbrecht, H.: Embodied Presence in Virtual Environments. In: Paton, R. (ed.) *Visual Representations and Interpretations*, pp. 269–278. Springer, London (1999)
29. Witmer, B., Singer, M.: Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments* 7, 225–240 (1998)
30. Schraw, G.: Situational Interest in Literary Text. *Contemporary Educational Psychology* 22, 436–456 (1997)

Collaborative Lecturing by Human and Computer Tutors

Sidney D’Mello, Patrick Hays, Claire Williams, Whitney Cade,
Jennifer Brown, and Andrew Olney

Institute for Intelligent Systems, University of Memphis, USA
{sdmello, dphays, mcwilliams, wlcade, jlbrown7, aolney}@memphis.edu

Abstract. We implemented and evaluated a collaborative lecture module in an ITS that models the pedagogical and motivational tactics of expert human tutors. Inspired by the lecture delivery styles of the expert tutors, the collaborative lectures of the ITS were conversational and interactive, instead of a polished one-way information delivery from tutor to student. We hypothesized that the enhanced interactivity of the expert tutor lectures were linked to efforts to promote student engagement. This hypothesis was tested in an experiment that compared the collaborative lecture module (dialogue) to less interactive alternatives such as monologues and vicarious dialogues. The results indicated that students in the collaborative lecture condition reported more arousal (a key component of engagement) than the controls and that arousal was positively correlated with learning gains. We discuss the implications of our findings for ITSs that aspire to model expert human tutors.

Keywords: collaborative lecture, expert tutor, Guru, arousal, engagement.

1 Introduction

There is probably nothing more boring and less effective than a lecture on a topic that the recipient has little or no intrinsic motivation to learn. Most would agree that pedagogical activities in the form of “long-winded didactic explanations” that are characteristic of lectures [1] have little to no value, at least when compared to more interactive alternatives such as scaffolding explanations and active problem solving [2, 3]. Although lectures go by many names such as transmission/information delivery [4], direct instruction [5], and didactic teaching [1], they never make the list of ideal tutoring models. Simply put, lectures are inefficient at promoting deep learning because polished deliveries of information by a teacher or a tutor makes the typical student a passive information receiver rather than an active problem solver [2].

Given this bleak sketch of the merits of lecturing in educational contexts, we were somewhat surprised to discover that lectures were abundant in our analysis of 50 naturalistic tutoring sessions between students and expert human tutors [6]. In particular, when we segmented the tutoring sessions into eight *dialogue modes* (i.e., pedagogically distinct phases in a session that last for several minutes and encompass multiple speech acts), lecturing was the second most frequent mode. Lectures

comprised 22.1% of the modes and 30.2% of the turns. Lectures were only surpassed by the scaffolding mode, which comprised 27.8% of the modes and 46.4% of the turns [7].

One explanation for the somewhat counterintuitive finding of the relatively high incidence of lectures might lie in the students that were tutored. These students were seeking expert tutoring because they were having considerable difficulty in their classes. It might be the case that the expert tutors extensively lectured in order to provide the necessary common ground before collaborative problem solving can be effective or even functional. There is some evidence to support this hypothesis. First, interactive problem solving is not very effective if the students do not have the requisite knowledge base [8]. For example, it is difficult to imagine a student solving a cytokinesis problem (cell splitting) without knowing what a cell is. Second, and more importantly, problem scaffolding is most likely to follow lectures in the expert tutoring corpus [7]. Hence, it is reasonable to assume that lectures are used to establish the knowledge foundation (i.e., common ground) upon which problems can be modeled, scaffolded, and faded [3].

The fact that lectures are frequent in expert tutoring has important implications for ITSs that aspire to model expert tutors. We are currently in the process of developing a tutoring system (Guru) for high school biology based on the tactics, actions, and dialogue of expert human tutors. It is in this respect that expert tutor lectures are very relevant to our research.

The process of developing a computational model of expert tutoring for Guru, highlighted some important characteristics of expert tutor lectures. Contradictory to the popular conception of lectures primarily being a one-way information transmission stream from the tutor to the student, we were intrigued to discover that the expert tutor lectures were quite interactive [9]. Although direct instruction and explanations played central roles, the lectures were filled with opportunities for students to play a more active role by doing some of the talking.

For example, tutors attempt to keep the student engaged via comprehension gauging questions (e.g., "Do you understand?"). There is some evidence that these questions are not very useful because students cannot accurately monitor their own understanding [10-12]. However, tutors might interleave these questions into the direct instruction cycle to enhance students' engagement and also to cue students to the fact that they need to be actively comprehending the lecture. A more active form of collaboration occurs when tutors directly engage the student via hints, prompts, forced choices, and simplified problems. These activities make students active participants in the tutorial sessions despite the fact that the primary goal of lectures is to deliver information.

In summary, our analysis of lectures during expert tutoring sessions was not consistent with boring, extended, long-winded, explanations. Instead, we found that expert tutor lectures were highly collaborative, presumably because the expert tutors acknowledge that active participation, even during lectures, is key to learning and engagement [13].

We have recently implemented and evaluated a lecture module that closely mirrors the expert tutor lectures. Although we do not expect impressive learning gains from the lecture module, we predict that the collaborative lecturing strategy will boost engagement, at least when compared to non-interactive alternatives (monologues and

vicarious dialogues). We tested this hypothesis in an experiment where collaborative lecturing (called dialogues) was compared to monologues and vicarious dialogues. Our prediction is that students will be more engaged in the dialogue condition than the other two less collaborative conditions.

2 Modeling and Implementing the Collaborative Lecture

2.1 Modeling the Collaborative Lecture

An extensive analysis of the lecture strategies of our sample of 50 expert tutors is discussed in [9], hence, we will focus on the major points here. In particular, there are two major clusters of dialogue moves as illustrated in Figure 1. The first cluster (information-transmission) is primarily concerned with the tutor delivering information to student (solid lines in Figure 1). The tutor may assert some information (direct instruction and explanation, die), to which the student provides backchannel feedback via an acknowledgment (ack), and the tutor asserts more information (die). Alternatively, the tutor transmits some information (die), asks a comprehension gauging question (cgq) (e.g., “Do you understand?”). The student replies with an acknowledgement (e.g., “Yes sir”) or a metacomment (e.g., “No. I don’t quite get it”), and more information is transmitted. These basic patterns associated with information-transmission account for 70.2% of the dialogue moves during lectures.

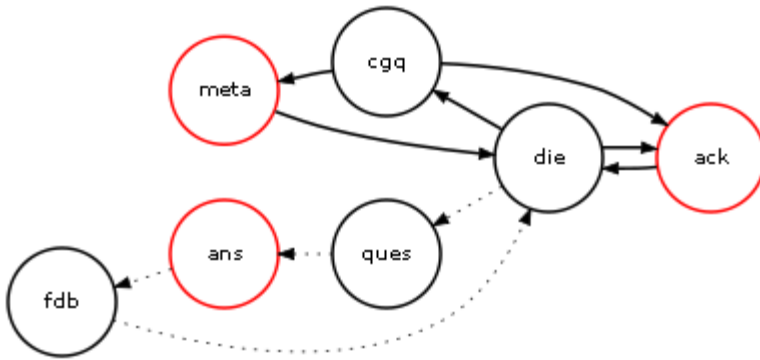


Fig. 1. Information-transmission and information-elicitation clusters

The second cluster, or the *information-elicitation* cluster (dotted links in Figure 1), consists of moves associated with attempts by the tutor to elicit information from the student. These moves are variations of the Initiate Respond Evaluate (IRE) sequence [14]. The sequence begins by the tutor asking the student a question (ques) with prompts, pumps, forced choices, or simplified problems. The student responds with an answer (ans). The tutor evaluates the student’s response and provides feedback (fdb) followed by more direct instruction (die). This cluster accounts for 18.6% of the moves during lectures.

In addition to these primary clusters that account for 88.6% of the dialogue moves, there is also an off-topic conversation cluster (9%), and a student-initiated question cluster (2.2%). The latter two clusters were not included in the current study because they are difficult to implement in the monologue condition (described below).

2.2 Implementing the Collaborative Lecture

We developed a lecture module in Guru for eight biology topics (cellular respiration, amino acids and RNA, etc). As previously stated, Guru's lecturing strategies were designed to closely mirror the expert tutor lectures. This was accomplished in two ways. First, the content of the lectures was obtained from transcripts of actual expert tutoring sessions. This made the lecture delivery style more conversational, informal, and presumably more engaging.

Second, the tutor closely modeled the collaborative lecturing tactics that were observed from our analysis of the human tutors (see Figure 1). In particular, Guru primarily transmitted information (68% of the time) but occasionally provided cues for acknowledgements (e.g., "Right?", "ok?"), asked comprehension gauging questions, and prompted the student for answers (e.g., "X is a type of what?"). On average, the lectures contained 32% opportunities for student involvement. The tutor:student dialogue move ratio of the tutor strongly correlated ($r = .97$) with the tutor:student ratio from the actual tutoring sessions. Hence, we are quite confident that Guru does indeed model the collaborative lecturing styles of the expert human tutors.

The lectures were delivered via a simple conversational interface that consisted of an animated conversational agent that delivered the content of the lectures via synthesized speech, a media panel that displayed images relevant to the lectures, and a dialogue box for students to type their responses.

We implemented two non-interactive variants of the collaborative lecture module. The collaborative module (called *dialogue*) closely mirrors the lecturing strategies of the expert tutors, as described above. Alternatively, in the *monologue* version, the tutor did all the talking and the student was a passive recipient. This module was designed to simulate a conventional non-collaborative lecture that is not expected to be very engaging.

The third version consisted of *vicarious dialogues*, where the dialogue patterns were structurally similar to the dialogue module, but with one important exception. Here, it was a virtual student, instead of the learner, that answered the tutor's comprehension gauging questions and prompts. The virtual student always provided the correct answer and the human learner simply watched the interaction. This was the only difference between the vicarious and the dialogue condition. All other aspects of the interface and interaction were equivalent.

Sample dialogues from the human tutors and Guru are presented in Table 1. In the actual lecture, the tutor introduces the topic (T1), uses a discourse marker (T2), asserts some information (T3), and then gives the student an opportunity to chime in (T4). The student provides an acknowledgment (S1), the tutor responds with a conversational OK (T5), asserts some more information (T6), and then prompts the student (T7). The student responds (S2), to which the tutor provides some feedback (T8), followed by an assertion, and so on (T9 and S3).

When Guru delivers a monologue for this sample lecture, it preserves most of the conversational style, asserts the same content, but does not give the learner an opportunity to type a response (see Table 1). In contrast, the learner in the dialogue condition has three opportunities to type in a response, which is consistent with the 1:3 student to tutor dialogue move ratio discussed above.

Although not included in Table 1, the vicarious-dialogue condition was identical to the dialogue condition. However, a virtual student, instead of the human learner, would type in (i.e., via simulated keystrokes) responses to items S1, S2, and S3. The simulated keystrokes were carefully calibrated in order to mirror the temporal dynamics of actual typing (i.e., onset delay, variable interstroke delay, and delay before hitting enter key to submit response).

On average, the expert human tutors articulated 790 words in each lecture, while the ITS articulated an average of 677, 718, and 718 words in the monologue, dialogue, and vicarious-dialogue conditions, respectively.

Table 1. Sample excerpts from lectures

N	Actual Lecture	Monologue	Dialogue
T1	Let’s talk about mitosis.	Let’s talk about mitosis.	Let’s talk about mitosis.
T2	Ok.	Ok.	Ok.
T3	Now, let’s say here’s a skin cell, he’s just sitting around, and he needs to divide.	Now, let’s say here’s a skin cell, it’s just sitting around, and it needs to divide.	Now, let’s say here’s a skin cell, it’s just sitting around, and it needs to divide.
T4	Someone’s got to tell him, right?	Someone’s got to tell him to divide.	Someone’s got to tell him to divide, right?
S1	Mm hmm.	<pause>	<student response>
T5	Ok	Ok	Ok
T6	I mean, let’s say a skin, skin cell is sitting around.	I mean, let’s say a skin cell is sitting around.	I mean, let’s say a skin cell is sitting around.
T7	Do you think somebody needs to tell him to split, or do you think he can just say, oh, I think I’ll split?	Do you think somebody needs to tell him to split, or do you think he can just say, oh, I think I’ll split?	Do you think somebody needs to tell him to split, or do you think he can just say, oh, I think I’ll split?
S2	Tell him it’s time?		<student response>
T8	Yeah!		
T9	Because, see, now folks need to get instructions, right?	Someone must tell him to split because he needs to get instructions.	Someone must tell him to split because he needs to get instructions, ok?
S3	Mm hmm.		<student response>

3 Method

Participants were 90 college students from a mid-south university in the US who participated for extra course credit.

Participants engagement levels were tracked at multiple points in the tutorial session with the affect grid [15]. The affect grid is a validated single item affect measurement instrument consisting of a 9 × 9 (valence × arousal) grid; these are the primary dimensions that underlie all affective experiences [16]. The arousal dimension ranges from

sleepiness to high-arousal, while the valence dimension ranges from unpleasant feelings to pleasant feelings. Participants indicate their affective state by marking an X at the appropriate location on the grid.

The knowledge tests (used to measure learning gains) were 24-item multiple-choice tests with three questions for each lecture. *Prompt* questions tested participants on content for which the tutor explicitly prompted the student in the dialogue and vicarious conditions. Although there were no explicit prompts in the monologue condition, we verified that the content of the prompts was explicitly covered in the monologue. *Assertion* questions tested participants on content that the tutor explicitly asserted to the student via direct instruction. Finally, there were *deep reasoning* questions that required causal reasoning, inference, etc. rather than recall of shallow facts. Participants completed alternate test versions for pretest and posttest that were counterbalanced across participants.

Participants were tested individually over a two hour session. Participants completed an informed consent followed by the pretest. They then read instructions on how to use the affect grid. On the basis of random assignment participants then completed a tutorial session with the monologue, dialogue, or vicarious tutor. There were 30 participants in each condition. The tutoring session consisted of eight lectures that were randomly ordered for each participant. Participants used the affect grid to indicate their affective state after each lecture. They completed the posttest after the tutorial session and were fully debriefed.

4 Results and Discussion

Engagement levels presumably decrease over time, hence, an analysis comparing engagement without controlling for time on task would be confounded. As could be expected, the monologue condition was shorter ($M = 37.2$ minutes) than the dialogue ($M = 54.6$) and vicarious conditions ($M = 55.4$). Since dialogue and vicarious were of equivalent length our first analyses compared engagement across these two conditions; monologues were considered in a follow-up analysis that equated time on task in a post-hoc fashion.

4.1 Engagement Levels

Self reported engagement trajectories were computed by averaging participants' valence and arousal scores (from the affect grid) for each lecture. These are presented in Figure 2 as a dialogue trajectory (D1, D2, ...D8) and a vicarious trajectory (V1, V2,...V8). When averaged across lectures, learners reported higher levels of arousal in the dialogue condition ($M = 4.10$, $SD = 1.80$) than the vicarious condition ($M = 3.00$, $SD = 1.30$), $t(58) = 2.73$, $p = .008$, $d = .70$. There was no significant difference ($p = .648$) in valence levels across conditions ($M = 4.48$, $SD = 1.66$ for dialogue and $M = 4.68$, $SD = 1.79$ for vicarious).

Comparisons of arousal scores for each lecture (i.e., D1 vs. V1, D2 vs. v2, etc) indicated that participants in the dialogue condition were significantly ($p < .05$) more aroused than their vicarious counterparts for the first six lectures. The difference was marginally significant ($p = .102$) in favor of the dialogue condition for the seventh

lecture. There was no significant difference ($d = .270$) for the eighth lecture, although there was a small to medium sized effect ($d = .3$) in favor of the dialogue condition.

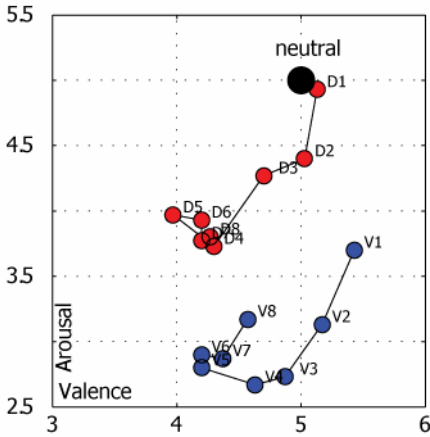


Fig. 2. Engagement trajectories. (Numbering, D1, D2, etc indicates order).

significant difference ($p = .098$) with a medium sized ($d = .50$) effect when arousal scores for the monologue condition were compared to the vicarious condition ($M = 3.00, SD = 1.30$).

The ANOVA comparing valence scores for the three conditions was not significant ($p = .750$), so the interactivity afforded by the collaborative lectures impacts arousal but not valence.

4.2 Learning Gains

Proportional learning gains were computed for each of the question types (prompts, assertions, and deep reasoning questions as described in the Methods section) as $(\text{posttest} - \text{pretest}) / (1 - \text{pretest})$. Since it is generally acknowledged that tutoring differentially benefits low versus high domain knowledge students, our analyses proceeded by dividing participant into these two groups on the basis of their pretest scores (see Table 2). There were no differences in learning gains across conditions for the high prior knowledge group, so the subsequent discussion focuses on the low domain knowledge group.

The results indicated that participants in the dialogue condition had marginally significantly ($p = .104$) higher scores for prompt questions when compared to their monologue counterparts ($d = .56$). The monologue versus vicarious comparison was not significant, however, there was a medium effect ($d = .49$) in favor of the vicarious condition. Hence, the pattern for prompt questions appears to be [Dialogue = Vicarious] > Monologue. This pattern is intuitively plausible because prompts direct the learner’s attention to specific words in the dialogue and vicarious conditions, but not the monologue condition (there were no explicit tutor prompts in this condition).

We performed a follow-up analysis that controlled for time on task. Specifically, mean arousal and valence scores were computed for each participant by only including their responses for the first 37 minutes, which is the mean length of the monologues. An ANOVA indicated that there was a significant difference in arousal scores across conditions, $F(2, 87) = 5.59, p < .01$. As predicted, arousal scores for the dialogue condition ($M = 4.3, SD = 1.81$) were significantly ($p < .5$ on a one-tailed test) greater than arousal scores for the monologue ($M = 3.65, SD = 1.34$) condition, with an effect size of .41 sigma. There was a marginally

Table 2. Mean proportional learning gains

Question	Low Prior Knowledge			High Prior Knowledge		
	Monologue	Dialogue	Vicarious	Monologue	Dialogue	Vicarious
Prompt	.27	.47	.43	.34	.14	.42
Assert	.43	.29	.17	.20	.32	.21
Deep	.28	.23	.22	.13	.19	.27

It should be noted that medium sized, marginally significant effects are meaningful for the current learning gains analyses because there was a significant loss of statistical power when the participants were split into low and high knowledge groups; these effects are likely to be significant with a larger sample.

A somewhat different pattern was observed for questions that tested participants' retention of the tutor's assertions. Here, the monologue condition was on par with the dialogue condition, but outperformed the vicarious condition ($p = .051$, $d = .81$). These results suggest that participants in the vicarious condition overlooked important assertions by the tutor, presumably because their focus was on the virtual student's responses to the tutor's prompts.

There was no difference in learning gains for deep reasoning questions. In summary, these results suggest that when it comes to low prior knowledge students, the monologue and vicarious conditions yield *inconsistent* results because they are ineffective for prompts and assertions, respectively. In contrast, low-domain knowledge students assigned to the dialogue performed *consistently* across the different question types (i.e. it was never significantly worse than other conditions).

4.3 Correlations between Engagement and Learning Gains

Our results so far are indicative of (a) the following hierarchical ordering of arousal levels across conditions: Dialogue > Monologue > Vicarious, (b) equivalent valence levels, and (c) differential patterns in learning gains across conditions and prior knowledge. It appears that it is arousal and not valence that is most relevant to learning gains. Arousal is correlated with deep learning gains in the dialogue and vicarious conditions and overall learning gains (i.e. gains not segregated by question category) in all three conditions (see Table 3).

Table 3. Correlations between engagement and learning

Condition	Arousal				Valence			
	Prompt	Assert	Deep	Overall	Prompt	Assert	Deep	Overall
Monologue	.072	.315*	.188	.363**	.002	-.280	.115	-.132
Dialogue	.022	.042	.537**	.347*	-.075	.203	.112	.119
Vicarious	.441**	.154	.347*	.492**	.073	.145	.144	.146

Notes. ** $p < .05$; * $p < .10$

There was one more interesting pattern pertaining to the relationship between arousal and learning. Recall that participants in the monologue condition outperformed vicarious participants for assertion questions, while a reverse pattern was observed for prompt questions. The correlational analyses indicate that these patterns were related to self reported arousal, thereby providing further evidence that it is arousal and not valence that is relevant to learning gains.

5 General Discussion

As most people in the field of education will attest, the task of keeping students engaged in educational activities is extremely challenging. Establishing and maintaining student engagement is especially critical in situations with high degrees of learner control, such as in distance education, computer-based tutoring, and informal learning environments, because learners are a mouse click away from ending the session. The engagement problem is undoubtedly more severe in situations where the computer tutor does most of the talking as when lectures are delivered to remedial students.

Although we were initially surprised by the high incidence of lectures in our sample of 50 expert tutoring sessions, we hypothesized that expert tutors implement a collaborative lecturing strategy to avoid the pitfalls associated with boring, one-way, didactic instruction. This hypothesis was confirmed in our evaluation of a computer tutor that simulated the lecturing style of expert human tutors. Our results indicated that arousal, a key component of engagement, was higher in the condition that implemented collaborative lecturing when compared to less interactive alternatives. Furthermore, arousal is critical because it is positively correlated with learning gains.

The correlation between arousal and learning gains is consistent with theories that highlight the importance of affect to deep learning [17, 18]. Physiological arousal is a universal and fundamental dimension of affective experience, a component of all emotional episodes, and a signal for alertness and action [16]. Hence, it comes as no surprise that arousal was highest in the most interactive condition and that arousal was linked to learning gains.

Our implementation of the collaborative lecture strategies of expert human tutors is one important step towards the larger goal of understanding the tactics that underlie their effectiveness. However, several important questions have not yet been answered. How do the expert tutors blend lecturing and scaffolding in order to optimize learning gains? What motivational strategies do they use to enhance self-efficacy and heighten engagement? How do they detect and respond to students affective states in order to prevent students from wallowing in negative emotions and promote more fruitful trajectories of thought? It is our hope that answers to these questions will deepen our understanding of expert tutors and launch next-generation ITSs to new levels of effectiveness.

Acknowledgements. This research was supported by the by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A080594. The opinions expressed are those of the authors and do not represent views of the IES.

References

1. Chi, M.: Constructing Self-Explanations and Scaffolded Explanations in Tutoring. *Applied Cognitive Psychology* 10, 33–49 (1996)
2. Bransford, D., Brown, A., Cocking, R. (eds.): *How People Learn: Brain, Mind, Experience, and School Committee on Developments in the Science of Learning*. National Academy Press, Washington (2000)
3. Rogoff, B., Gardner, W.: Adult Guidance of Cognitive Development. In: Rogoff, B., Lave, J. (eds.) *Everyday cognition: Its development in social context*, pp. 95–116. Harvard University Press, Cambridge (1984)
4. Wheatley, G.: Constructivist Perspectives on Science and Mathematical Learning. *Science Education* 75, 9–21 (1991)
5. Lu, X., Di Eugenio, B., Kershaw, T., Ohlsson, S., Corrigan-Halpern, A.: Expert vs. Non-expert Tutoring: Dialogue Moves, Interaction Patterns and Multi-Utterance Turns. In: *Proceedings of Eighth International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 456–467. Springer, Heidelberg (2007)
6. Person, N., Lehman, B., Ozbun, R.: Pedagogical and Motivational Dialogue Moves Used by Expert Tutors. In: *Proceedings of 17th Annual Meeting of the Society for Text and Discourse*, Glasgow, Scotland (2007)
7. Cade, W., Copeland, J., Person, N., D’Mello, S.: Dialogue Modes in Expert Tutoring. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008. LNCS*, vol. 5091, pp. 470–479. Springer, Heidelberg (2008)
8. Ausubel, D.: *Educational Psychology: A Cognitive View*. Holt, Rinehart and Winston, Inc., Austin (1978)
9. D’Mello, S., Olney, A., Person, N.: Mining Collaborative Patterns in Tutorial Dialogues. *Journal of Educational Data Mining* (in review)
10. Glass, M., Kim, J., Evens, M., Michael, J., Rovick, A.: Novice vs. Expert tutors: A Comparison of Style. In: *Midwest Artificial Intelligence and Cognitive Science Conference*, Bloomington, IN (1999)
11. Graesser, A., Person, N., Magliano, J.: Collaborative Dialogue Patterns in Naturalistic One-To-One Tutoring. *Applied Cognitive Psychology* 9(6), 495–522 (1995)
12. Person, N., Graesser, A., Magliano, J., Kreuz, R.: Inferring What the Student Knows in One-to-One Tutoring - the Role of Student Questions and Answers. *Learning and Individual Differences* 6(2), 205–229 (1994)
13. VanLehn, K., Graesser, A.C., Jackson, G.T., Jordan, P., Olney, A., Rose, C.P.: When Are Tutorial Dialogues More Effective Than Reading? *Cognitive Science* 31(1), 3–62 (2007)
14. Mehan, H.: *Learning lessons: Social Organization In The Classroom*. Harvard University Press, Cambridge (1979)
15. Russell, J.A., Weiss, A., Mendelsohn, G.A.: Affect Grid - A Single-Item Scale of Pleasure and Arousal. *Journal of Personality and Social Psychology* 57(3), 493–502 (1989)
16. Russell, J.: Core Affect and the Psychological Construction of Emotion. *Psychological Review* 110, 145–172 (2003)
17. Meyer, D., Turner, J.: Re-conceptualizing Emotion and Motivation to Learn in Classroom Contexts. *Educational Psychology Review* 18(4), 377–390 (2006)
18. Csikszentmihalyi, M.: *Flow: The Psychology of Optimal Experience*. Harper and Row, New York (1990)

Computational Workflows for Assessing Student Learning

Jun Ma, Erin Shaw, and Jihie Kim

Information Sciences Institute, University of Southern California
4676 Admiralty Way, Marina del Rey CA 90292, United States
{junma, jihie, shaw}@isi.edu

Abstract. The use of technology for instruction, and the enormous amount of information available for consumption, places a considerable burden on instructors who must learn to integrate appropriate student practices and learning assessment. The Pedagogical Workflows project is developing a novel workflow environment that supports efficient assessment of student learning through interactive generation and execution of various assessment workflows. We focus especially on how student discussion use can be combined with more traditional assessment data. In this paper, we present our initial assessment workflows, the initial feedback from instructors, and the user portal that is being developed for running the workflows. Inherent in the development of the workflows is an examination of what teachers think is important to learn about their students, a question that is central to every intelligent tutoring system. We anticipate that assessment workflows will become an important tool for instructors, researchers, and ITS development.

Keywords: Discourse analysis, workflow technology, student learning assessment.

1 Introduction

Educational technology for online learning is now centrally supported by many colleges and universities. The perceived mandate to use technology for instruction, in addition to the enormous amount of information available for consumption on the Web, places a considerable burden on instructors who must learn to integrate appropriate student practices and learning assessment via the new media. Discussion boards, for example, have become an essential tool for student-student and student-instructor communication beyond the walls of the classroom; however, properly integrating participation results and traditional assessments is very much a challenge.

Workflow technology has been successfully applied to scientific applications [1, 2]. Existing workflow generation and execution approaches can be useful for making educational assessment tools more accessible to instructors, and for making large scale assessment, requiring large amounts of course data, feasible. Workflow results can be used to answer questions and provide formative feedback to instructors to facilitate “just in time” instructional adaptation to students learning and needs. Our workflow project modularizes the selection and application of both traditional and non-traditional assessment techniques in online instruction. Instructors can use

workflows to perform a variety of correlations, including those that require privacy protected data such as grade information. For example, an instructor may wish to analyze student discussion participation data, the grades of a student or entire class, and additional data from earlier semesters, to answer questions such as ‘What type of discussion board use correlates to better performance?’, ‘How long do students wait before classmates respond?’, ‘How do online activities this semester differ from those of previous semesters?’ These questions guide instructional improvement and are traditionally difficult to answer when focused on online environments.

The goal of our project is to create a novel workflow environment that supports efficient assessment of student learning through interactive generation and execution of various assessment workflows. With respect to online activities, we focus on qualitative methods for evaluating learning by discourse [3, 4]. These methods use information retrieval and natural language processing (NLP) techniques to analyze the impact of discussion board participation on conceptual understanding and communication, both forms of cognitive assessment that inform learning. In addition, we are taking into account longitudinal student changes by electronically tracking students’ learning performance across courses as they matriculate to their degree [5].

In this paper, we present our initial assessment workflows that are developed based on the Wings/Pegasus workflow system [6, 2] and initial feedback we received from two Computer Science instructors. We also show the PedWorkflow portal which was developed for the instructors to support easy access.

1.1 Assessment Questions and Supporting Workflows

Table 1 illustrates the assessment question categories that we have compiled so far and Table 2 gives examples of questions from these categories. Some of these questions come from interactions with engineering instructors who use our enhanced online discussion board. Other questions are from research in learning assessment [7, 8]. Most questions require the execution of some combination of multiple workflow steps, where often concurrent steps are desirable. Some of the jobs that perform Trend

Table 1. Assessment categories and descriptions

Category	Workflow Description
Analysis of online activities	Composition of discussion data processing/ Classification steps
Correlation between online activities & performance	Composition of discussion data processing steps, student profiles, and correlation analysis
Correlation between online activities & self-assessment	Composition of self-assessment survey, student activity profiles, and correlation analysis
Student profiling	Composition of student information and discussion data processing/ classification steps
Discussion profiling	Composition of discussion data processing/classification steps and relation analysis
Trend analysis	Splitting of discussion data and iterative analysis
Group comparison	Composition of discussion data processing/ classification steps, student profiles and relation analysis

Analysis, which use data sets from different time frames, or Correlation Analyses, which analyze data from multiple semesters, may need parallel execution to improve efficiency. Some of the workflows or sub-workflows may be re-used for similar questions. Provenance of the workflow creation and execution will help the instructor keep track of student activity details and understand how the results are produced.

Table 2. Assessment questions and their priority ratings by two instructors

Priority Rating		Question
L	M	Which topics have been discussed in the last three semesters?
H	H	Which topics do students ask the most questions about?
H	M	Were all of the questions about topic x answered?
L	H	Which questions were unanswered?
L	M	Do students who participate more often receive better grades?
H	M	Do gender and politeness affect participation?
H		Do more motivated students perform better?
H		Do more confident students participate more?
H	M	Who are the mentors for topic x?
H	H	Which students are confused about topic x?
H	H	Is a student a mentor or help seeker?
H	L	What are his/her strengths?
H	L	Were there similar questions or answers in previous semesters?
H	M	How long did students have to wait for an answer?
H	M	How has student participation changed over time?
H	L	How are online activities in this semester different from previous semesters?

Two instructors, who are working with us to develop and evaluate assessment workflows, were asked to rate the priority of the assessment questions in Table 2 as part of a formative assessment. Their ratings are shown on the left (L=low priority, M=medium, H=high). Both instructors teach multiple undergraduate courses and both have been teaching for at least 10 years. Each had very different concerns. The first instructor (leftmost rating) actively participates in discussions and thus gleans information about topics discussion and questions answered directly, and so rated these types of assessment questions as low priority. Where as the second instructor rated them as high or medium. The second instructor was less focused on correlating personal traits and more focused on identifying students who were behind or who might deserve extra credit, as well as flagging topics that students ask many questions about. The second instructor uses his course teaching assistant to monitor discussions but said he would prefer an objective measure.

Teachers were also asked to contribute assessment questions they might use for their own classes. Both instructors mentioned the importance of having objective measures that were linked to class goals. One instructor desired concrete evidence that the non-computer science students for whom his class was required did not have the pre-requisite skills to succeed in his class. In one of the classes taught, the assignments were team projects and the instructor wished to assess individual students relative to their teammates. Identifying patterns of student performance relative to student activity was also discussed. The results indicate a need for facilitating measurement for many types of assessment.

2 Workflow Portal for Learning Assessment

2.1 Background: Wings/ Pegasus Workflow System

Wings takes the user's workflow requirements and generates a high-level workflow for execution [2]. Pegasus generates executable workflows by assigning execution resources to the computations in the workflow [1]. Pegasus also reduces the workflow execution time by eliminating unnecessary computations whose results already exist and can be reused, and reorganizing the structure of the workflow to minimize job queuing time and data movements. It then submits the workflows to the grid for execution and monitors their status.

There have been intelligent interfaces developed for supporting end-user composition of workflows [9, 10]. The approach exploits knowledge-rich descriptions of the individual components and their constraints in order to validate the composition, and uses artificial intelligence planning techniques in order to systematically verify formal properties of valid workflows. The system analyzes partial workflows created or modified by the user, determines whether they are consistent with the background knowledge that the system has, notifies the user of issues to be resolved in the current workflow, and suggests to the user what actions could be taken to correct those issues. To represent and reason about datasets, workflow components, and workflows, OWL W3C's standard (OWL 2010) [11] and Jena (Jena 2010) [12] were used.

2.2 The PedWorkflow Portal

The Wings/Pegasus workflow system is a powerful tool for semantic workflow generation and execution [2]. Its portal services ease the use of the workflow system by end users. Based on them, we created the PedWorkflow Portal for instructors. The instructors do not need to write any workflow template nor any programming codes to use the workflow system.

The PedWorkflow Portal contains two main parts: the first one is the workflow generation and execution that enables the instructors to select and run a specific assessment workflow with data of his/her interest. And the second one is its presentation of workflow runs that enables the instructors to access and understand the results using the web interface.

When an instructor logs onto the PedWorkflow Portal, he can view the assessment questions that are supported by the portal. For each question, there is a corresponding workflow template in the workflow template library. Once the specific question (i.e. workflow template) has been selected, he can select/upload the inputs for the workflow. The input can be files, database entries or parameters. Once the inputs of the workflow template have been identified, the PedWorkflow Portal binds the data with the workflow template and uses the Wings engine to instantiate the workflow template with the inputs. The portal then executes the workflow instance. During the execution, each step run is recorded and the instructor can monitor the generation and running of the workflow from the portal interface. When the workflow execution is finished, the instructor can view all the output data and intermediate data using the portal in the result page. The procedure is illustrated in Figure 1: the left window shows selection of assessment question (workflow template) and dataset binding.

Once the instructor click run, then the top right window pop up showing the workflow generation and execution detail. In viewing the details, there are links to all data corresponding to this run of the workflow, including the final output graphs, which are shown in the bottom right window.

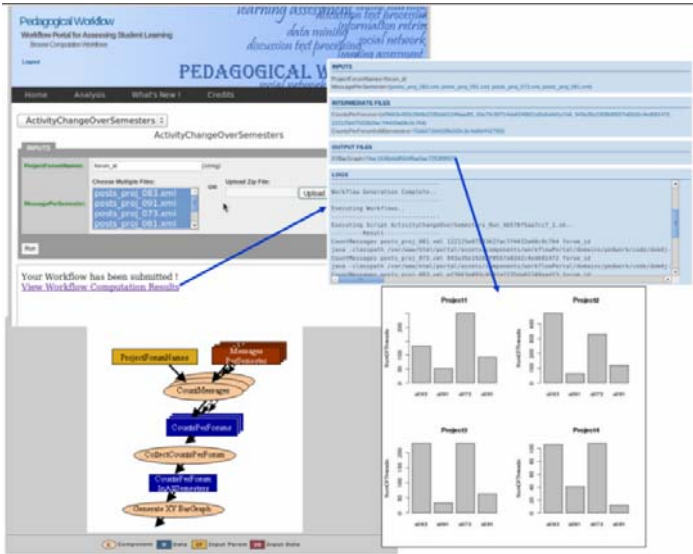


Fig. 1. The PedWorkflow Portal

3 Learning Assessment Workflows

This section describes several categories of assessment workflows that we have developed so far: 1) collective analysis of online student activities that covers all the students in the class; 2) correlation analysis with multiple measures: student online activity vs. performance; and 3) comparative analysis of data from multiple classes. We illustrate how our workflow framework supports instructional assessment.

3.1 Analysis of Online Activities

This type of analysis collects data from all the participating students, and combines the dataset with respect to a certain measure, to provide insight on student collective behaviors. The following is an example query:

How long did students have to wait for an answer?

Each discussion thread consists of question and answer exchanges among students, instructors and teach assistants. Thus the average time a student needs to wait for a question is of great importance to evaluate the effectiveness of the online discussion board.

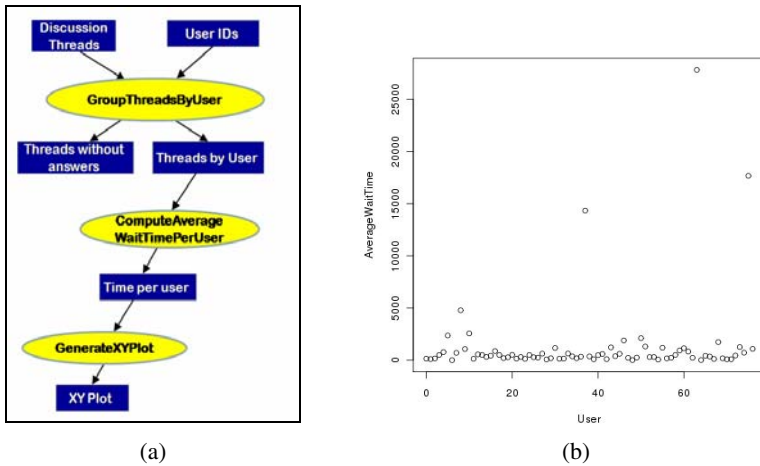


Fig. 2. The workflow diagram for StudentWaitTimeAnalysis and an output example

The computation of analyzing wait time can be split into several steps. Each computation step is a workflow component in component library. In this example, first we need to have all discussion thread data and all student profile data as inputs (DiscussionThreads and StudentIDs in the workflow diagram Fig 2(a)). Then we use a grouping component GroupThreadsByUser to group the threads by users. Moreover, we locate each user's participation in different discussion threads, and compute the difference between the initial post (i.e. the initial question by a student) and the following response to the question. The output file ThreadsByUser for this grouping component is an xml file recording the all the sessions' waiting time of each user. We also export all the threads that are not replied with an answer as the output file ThreadsWithoutAnswers.

In the second step, we pick a general average computing component called ComputeAverageWaitTimePerUser component and calculate the average wait time for each student. The output of this component is a raw data table TimePerUser which indicates the average wait time for each user. And this output can be further fed into general plot components for result visualization, like the GenerateXYPlot component in this example, which is actually an R function in component library that plot, the input 2D data table into an image figure file XYPlot.

Once we have chosen a component from the component library to build up our workflow, we import it into workflow and link the corresponding input/output files to connect different components, as shown in Fig 2(a). The result is called workflow template, which is represented by OWL. We cannot directly run workflow templates since they are not fully instantiated yet. An executable workflow needs bindings of input/output files and parameters. In case we need multiple runs of the same component, the system needs to know how many runs are needed for each component based on the input file bindings. The PedWorkflow Portal system has an interface to generate the instance of workflow templates dynamically. The interface enables the user to upload/select the input files and parameters, and then generate the corresponding

workflow instance. Finally, the workflow is executed by either local PedWorkflow Portal or the Wings/Pegasus grid system. The result is returned to Portal once the running of the workflow instance is finished.

Figure 2(b) shows the output of the StudentWaitTimeAnalysis workflow with the input discussion threads of CSCI402 Fall2007 at USC. We can see from the graph that the discussion response is highly active: most threads can be replied within 24 hours (1440 minutes). We can also see from the result that there are many students participating in the online discussion: 75 online users plotted in the graph, among 120 students enrolled in the class.

3.2 Correlation between Online Activities and Performance

Instructors may be interested in many types of correlation analyses. Here is an example question:

Do students who participate more often receive better grades?

In this case, we want to find the relationship between the student grades and the student activity in online discussion. Instructors want to know whether students who participate more often receive higher course grades. This workflow needs student grades data as input. The workflow running instance should keep the grades data private during the running of the workflow. We are currently using dedicated file system in a protected local machine but plan to explore more secure approaches.

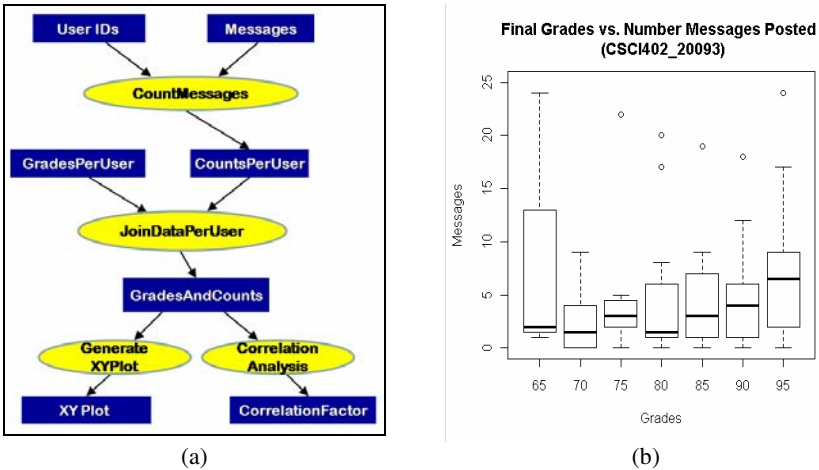


Fig. 3. The ActivityVsGradeAnalysis workflow diagram and an output example

We also decompose our task the same way we did in the previous example to build up the workflow template: The grade and activity analysis can be represented by four computation steps. The first component (CountMessages in Fig 3(a)) computes the activity weight for each student. Here we use the number of messages posted as the weight of a student's activity. This produces the CountsPerUser file as the output. Second, the

student activity data (*CountsPerUser*) is *joined* with student grades (*GradesPerUser*) using the *JoinDataPerUser* component. The grade information is provided to the workflow running instance directly by the instructor. The data is stored in a dedicated local file system as described above. Once we have both grades and activities we join the data into a data table (*GradesAndCounts*), which indicates each student's grade and corresponding activity weight. The system then uses the data table to compute the correlation using the *CorrelationAnalysis* component. Finally, a graph is generated through the *GenerateXYPlot* component, which was also used in the previous example.

Fig 3(b) is a project grade vs. project activity box-and-whisker plot with the same data used in Fig 2. For each grade level, five number summaries are presented: minimum, first quartile, median, third quartile and maximum. The output shows that there is a trend towards higher grades for higher discussion participation, as shown by medians, but the trend is weak and the grade-activity relation is not strong.

3.3 Group Comparison: Compare Student Activities over Semesters

How are online activities in this semester different from previous semesters?

Finally, we might want to compare online participation across different semesters. For example, we may wish to compare the number of discussion threads that correspond to each course project for four past semesters, for a specific course. In this case we need *component collections*; we need to combine the result of many duplicates of a component with different data inputs.

As illustrated in Fig 4(a). We have the same *CountMessages* component as before to calculate the number of threads along all the course projects in the semester (this

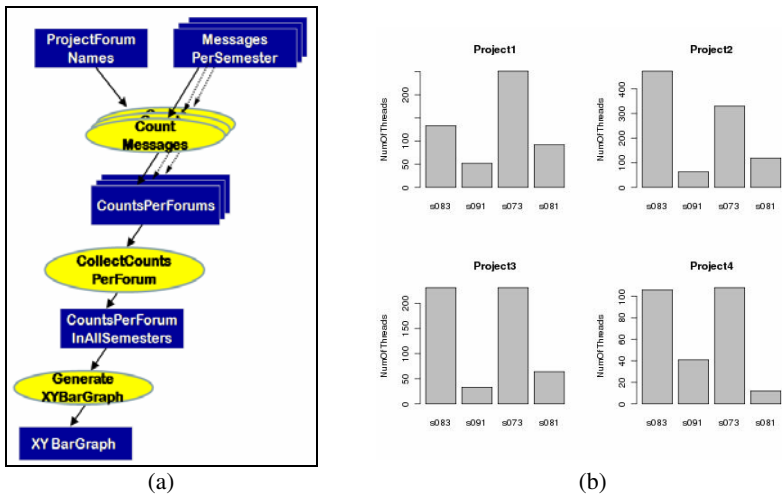


Fig. 4. ActivityChangeOverSemesters workflow diagram and its output (A comparison of course project 1~4 between 4 semesters:07Fall, 08Spring, 08Fall, 09Spring)

will be achieved by providing ProjectForumNames to the component rather than providing UserIDs to it). Since the CountMessages component is used for each semester input, the workflow system automatically make copies of component runs based on the number of files in the input collections. The output file collections (CountsPerForums) will be fed into the CollectCountsPerForum component, which joins the count result of all the semesters and generates a data table for further visualization using GenerateXYBarGraph component.

An example output is shown in Fig 4(b), we instantiate the workflow with four semesters' of CSCI402 discussion data, which contain discussions about the courses' four project assignments per semester, and we compare the activity for each project's discussion across the semesters. The graph is valuable for instructors who wish to assess student learning from a historical perspective.

One major advantage of creating workflows as tools to analyze online education discussion is that the workflow representation is abstract, semantic and data-independent. Moreover, the components and workflows are reusable. For example, we can easily extend this workflow to analyze the number of threads about a particular discussion topic, over multiple semesters, by adding a topic detection component in front of the CountMessages component, without changing any code in the current system.

4 Related Work

Researchers are working on non-traditional, qualitative assessment of instructional discourse include [14, 15]. Our workflows can incorporate some of these as workflow components. Combined with traditional cognitive assessment methods such as assignment and exam grades, our workflow-based approach can be powerful tool in assessing impact of online learning. There have been interests in a *longitudinal* (repeated measures) analysis of student changes [5, 13]. We can use workflows that electronically track students' learning performance across courses as they progress to their degree completion.

5 Summary and Discussion

We have presented a novel learning assessment approach that is empowered by computational workflow techniques. We are providing a workflow portal for instructors, to assist them in integrating discussion participation results and traditional assessments. Initial feedback from instructors indicates that the system will help track student activities efficiently and that its assessment results will potentially change their teaching strategies.

We are currently increasing the number of assessment questions that the system can handle. Since the existing workflow system cannot effectively handle dynamic data access and repeated runs of the same workflow, we are developing new extensions to the workflow architecture that will handle diverse assessment workflows. We will also improve the user interface for readability and usability.

Acknowledgement

This work is supported by the NSF, CISE Information and Intelligent Systems award (#0917328). The authors thank Instructors Micheal Crowley, Geza Bottlik and Jim Arvo for their time and support of the project. We also thank Varun Ratnakar and Gaurang Mehta for their help with software installation.

References

1. Deelman, E., Singh, G., Su, M., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G.B., Good, J., Laity, A., Jacob, J.C., Katz, D.S.: Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems. *Scientific Programming Journal* 13(3) (2005)
2. Gil, Y., Ratnakar, V., Kim, J., Gonzales-Calero, P., Groth, P., Moody, J., Deelman, E.: WINGS: Intelligent Workflow-Based Design of Computational Experiments. *IEEE Intelligent Systems* (2010)
3. Kim, J., Shaw, E.: Pedagogical Discourse: Connecting Students to Past Discussions and Peer Mentors within an Online Discussion Board. In: *The 21st Innovative Applications of Artificial Intelligence Conference* (2009)
4. Ravi, S., Kim, J.: Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers. In: *Proceedings of the AI in Education Conference* (2007)
5. Reed-Rhoads: C16 – Tools for Assessing Learning in Engineering. *Presentation on Inventions and Impact 2: Building Excellence in Undergraduate Science, Technology, Engineering, and Mathematics (STEM) Education* (2008)
6. Kim, J., Deelman, E., Gil, Y., Mehta, G., Ratnakar, V.: Provenance Trails in the Wings/Pegasus Workflow System. *Concurrency and Computation: Practice and Experience, Special Issue on the First Provenance Challenge* 20(5) (April 2008)
7. Suthers, D., Hundhausen, C.: The effects of representations on students' elaborations in collaborative inquiry. In: *Proceedings of Computer Support for Collaborative Learning*, pp. 280–472. Erlbaum, Hillsdale (2002)
8. Stahl, G.: The Complexity of a Collaborative Interaction. In: *Proc. of the International Conference of the Learning Sciences, ICLS 2002* (2002)
9. Kim, J., Gil, Y., Spraragen, M.: Principles for Interactive Acquisition and Validation of Workflows. *Journal of Experimental and Theoretical Artificial Intelligence* (2009)
10. Kim, J., Spraragen, M., Gil, Y.: An Intelligent Assistant for Interactive Workflow Composition. In: *Proceedings of the International Conference on Intelligent User Interfaces* (2004)
11. OWL 2010 (2010), <http://www.w3.org/TR/owl-features/>
12. Jena 2010 (2010), <http://jena.sourceforge.net/>
13. Della-Piana, C., Pimmel, R., Watford, B.: Project Evaluation. In: *Workshop for Faculty from Minority Serving Institutions, February 8 -10* (2006)
14. McLaren, B.M., Scheuer, O., De Laat, M., Hever, R., De Groot, R., Rose, C.P.: Using Machine Learning Techniques to Analyze and Support Mediation of Student E-Discussions. In: *Proceedings of the 13th International Conference on Artificial Intelligence in Education* (2007)
15. Graesser, A.C., Olney, A., Ventura, M., Jackson, G.T.: AutoTutor's Coverage of Expectations during Tutorial Dialogue. In: *Proceedings of the FLAIRS Conference 2005*, pp. 518–523 (2005)

Predictors of Transfer of Experimental Design Skills in Elementary and Middle School Children*

Stephanie Siler¹, David Klahr¹, Cressida Magaro¹, Kevin Willows¹,
and Dana Mowery²

¹ Carnegie Mellon University, Department of Psychology,
5000 Forbes Avenue, 15213, Pittsburgh, PA, United States
{siler, klahr, cmagaro, KevinWillows}@cmu.edu

² Pittsburgh Science and Technology Academy,
107 Thackeray St., 15213, Pittsburgh, PA, United States
dmowery1@pghboe.net

Abstract. A vital goal of instruction is to enable learners to transfer acquired knowledge to appropriate future situations. For elementary school children in middle-high-SES schools, “explicit” instruction on the Control of Variables Strategy (CVS) has proven to be very effective at promoting transfer, even after time delays, when administered by human instructors [1], [2] and when administered by our computer tutor (“TED” for Training in Experimental Design). However, when the same instruction was delivered to students in low-SES schools, near—but especially far—transfer rates were lower. We discuss our findings of the predictors of transfer in this population, and an initial investigation assessing the causal status of one candidate factor for far transfer, understanding the logic of CVS. Finally, we discuss the potential implications of these findings for ways to adapt instruction to individual students.

Keywords: transfer; experimental design skills; computer-based tutor.

1 Introduction

The primary goal of instruction is to enable learners of widely varying abilities to transfer newly acquired knowledge to future situations. Over the past decade, our lab has studied the effects of several instructional and contextual factors, as well as student characteristics, on their ability to learn the core procedural and conceptual knowledge elements associated with simple experimental design and to transfer that knowledge to appropriate future contexts [1], [2], [3]. The “Control of Variables Strategy” (CVS) is an important domain-general topic in elementary and middle school science. It involves controlling all variables in an experiment except for the

* The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305H060034 to CMU and the National Science Foundation through the Pittsburgh Science of Learning Center, Grant SBE0354420. The opinions expressed are those of the authors and do not represent views of IES or the U.S. Department of Education.

focal variable, i.e. the variable whose effect on an outcome is being investigated. In this paper, we summarize our previous work in order to lay the groundwork for the (ongoing) creation of an adaptive, intelligent tutor to teach CVS. Our primary effort is to identify student-specific factors that reliably predict different levels of transfer, and then to devise ways to diagnose and rules for the tutor to respond to those individual factors in teaching children about CVS.

Our prior studies have revealed a consistent relationship between the average SES of the students in different participating classrooms and their ability to learn CVS. (Throughout this paper, our proxy for SES is the proportion of students in a school eligible for free or reduced-price meals.) In middle- and high-SES classrooms, human-delivered explicit CVS instruction that draws students' attention to the reasons why an unconfounded experiment is, in fact, unconfounded has been very effective in promoting not only *near transfer* (i.e., successfully designing experiments in the domain used during instruction) but also *far-transfer* (i.e., successfully designing and evaluating experiments in domains other than the instructional domain) [1], [2], [3]. However, in low-SES classrooms, where students typically have had less exposure to science inquiry, and have poorer reading and math skills — this type of instruction has been much less successful [4]. One of the aims of our current research is to determine what *cognitive* factors – probably correlated with the distal SES measure – are actually influencing the differential effectiveness of our instruction for high and low SES students.

Over the past several years, we have been incrementally converting this explicit instruction into an adaptive, intelligent, tutor that will assess student knowledge and provide highly tailored instruction on the procedural and conceptual aspects of CVS. We call our system the “TED” (Training in Experimental Design) tutor. In its current non-adaptive state, the TED tutor – running in either stand-alone or networked contexts – consists of the following series of components that closely replicate the procedures used in our “human teacher” experimental training studies [1], [2], [3] as well as several human-teacher classroom implementations [4], [5].

1. Story pretest: 6 items requiring students to design (3 items) or evaluate (3 items) experiments presented as “story problems,” and to provide a rationale for their responses. These story problems include three different contexts—cookie baking, drink sales, and rocket ship design. For each context, students are first asked to design an experiment to test a particular variable by selecting values for each of three variables in two conditions. Then they explain why they set up the experiment as they did. Following the design question, students evaluate a given experiment in that context as a “good” or “bad” way to find out whether the focal variable makes a difference and explain their response. If they indicate it is a bad way, they were asked to change it into a good experiment, which required them to change at least one value setting.
2. Ramps intro: A Flash-based section (~ 2 min) presenting simple animated color line drawings – accompanied by audio voice-over and dynamic visual pointers that introduce the four variables relevant in the ramps apparatus (steepness of slope, length of run, surface type, and ball type).
3. Ramps pretest: Similar visual and audio presentation of 4 test items. Each item requires students to (a) design an experiment to determine the causal role of one of the four ramp variables and (b) provide a rationale for their design. Students select

one of two values to use for each of four factors on each of two comparison ramps by moving parts of the diagram or selecting values in a table. The diagram and table are linked (e.g., a text selection of a “steep” ramp automatically raises the ramp steepness in the diagram, or increasing the ramp steepness in the diagram automatically generates “steep” in the corresponding table cell). At present, student justifications are entered as free text, but we are currently replacing text boxes with drop-down menus for students to select rationales for their designs. This will greatly simplify the challenge of using free-form explanations to infer knowledge states in the adaptive version of TED.

4. Introductory video: A brief (~ 2 min) video that consists of a professional actress presenting an instructional “lecture,” consisting of an introduction to experimental design, its purpose, scope, and the central idea of “comparing and contrasting” things to find out whether or not they produce different outcomes, accompanied by simple animated line drawings supported by Flash-based graphics. (Note: In later versions of TED, this video will precede the ramps intro.)
5. Explicit Instruction “EI”: Based on “explicit” CVS instruction developed in our previous studies [1], [2], [3], in this portion of instruction – delivered in the same format as the ramps pretest – students are presented with three different pairs of contrasting ramps set-ups (i.e., “experiments”). For each, they are probed for whether and why the design is or is not “a good way” to find out about the focal variable; and whether the design would allow them to “know for sure” whether the focal variable made a difference in outcomes; and why or why not. Students type responses to the “why” questions into text boxes. After responding to these deep questions, they are given feedback and an explanation for why the design could—or could not—lead to valid inferences about the focal variable. Any experimental confounds are corrected by the tutor, and students answer the same two deep probes described above. Students are then given feedback and an explanation for why the unconfounded experiment would lead to valid inferences about the focal variable.
6. Ramps posttest: Identical to the ramps pretest; this is our measure of *near-transfer* performance, completed by students immediately after the Explicit Instruction (“EI”).
7. Story posttest (identical to the story pretest); this is our measure of *immediate far-transfer* performance, completed by students the following day.
8. Delayed story posttest (identical to story pre/posttest, except for different focal variables); this assesses students’ delayed far-transfer performance three weeks later.

In the version of the TED tutor as of this writing, students progress from one component to the next in the sequence given above. We are presently integrating Bayesian knowledge tracing, driven by students’ menu-based responses on the ramps pretest, to decide when to take students through alternative assessment and instructional paths. At minimum, the adaptive version of TED will assess three CVS procedural knowledge components or “rules”: R1: identify the focal variable given in the problem statement; R2: contrast levels of the focal variable across conditions; R3: control all other variables. Based on their responses, students may be given instruction in one or more of these rules prior to entering the EI phase.

In a recent evaluation, the TED tutor was compared to human instructors who followed the same script and procedure (described above) but used physical rather than virtual materials. Because there was no difference between TED and human instructors in near and far transfer outcomes in either population of students who

did not display CVS mastery on the story pretest, human and TED-tutored students were combined. As shown in Table 1, and consistent with previous findings [4], the mastery rates of two classrooms of low-SES 5th-grade children (L2, $n = 16$; L3, $n = 14$) were lower than those of their middle-SES counterparts¹, ($n = 50$) particularly on the far-transfer assessments, where the transfer mastery rates were more than four times greater in the middle-SES classroom.

Table 1. Summary of transfer mastery rates from a recent TED evaluation

	Mastery ramps pretest	Near transfer mastery (ramps post) ^a	Immediate far transfer ^b	Delayed far transfer ^b
Mid-SES	20%	87.5%	61.9 ³ %	62.5%
Low-SES	10%	60.0%	13.4%	15.4%

^a At least 3 of 4 CVS set-ups. ^b At least 5 out of 6 CVS set-ups.

2 Predictors of CVS Transfer

In what follows, we first identify the predictors of these near and far transfer outcomes for the low-SES student population. Then we look at the relationship between student-specific measures and these predictors. Based on these findings, we propose ways to make instruction in the TED tutor adaptive to individual student users, in part by informing the Bayesian model of key knowledge components that are strong predictors of learning and transfer. We used data from the previously described study (classes L2 and L3) as well as from an earlier evaluation performed as part of the TED project, in which a science teacher administered EI to 6th-grade students in a low-SES classroom (L1, $n = 23$).

We first looked at which initial knowledge and standardized measures were most highly correlated with posttest performance for the two low-SES 5th-grade classrooms. These measures were ramps pretest, story pretest, standardized (CTB/TerraNova²) reading comprehension, science, nonverbal “IQ,” and verbal (or deductive) reasoning national percentile scores. The verbal deductive reasoning measure on the CTB/TerraNova test assesses the skill of identifying a conclusion that is based only on information given. To recognize the correct response, students must integrate the information to produce that response. This task also requires that students do *not* select distractors that may be consistent with common knowledge. The following is an example of a verbal deductive reasoning practice item (Level 2, for U.S. grades 4-5), with the correct response italicized:

A fire must have heat, air, and fuel or it will not burn.

Wood can be used as fuel for a fire.

The scouts made a campfire.

(a) The scouts used wood to make their campfire.

(b) The scouts toasted marshmallows over their fire.

¹ The two low-SES classes were from schools in which 95% and 59% of students were eligible for free or reduced lunch; the middle-SES class was from a school where 20% of students were eligible.

² For more information on this test, go to: http://www.ctb.com/mktg/terranova/tn_technical.jsp

- (c) *The campfire had heat, air, and fuel.*
 (d) The campfire burned for a long time.

Predictors of near transfer. Of these measures, only reading comprehension was significantly related to near-transfer performance ($r = +.47, p = .03$) in a forward regression. This relationship did not differ by classroom (L2 or L3) or condition (Human- or TED-tutored). The same result was found in the earlier evaluation, in which a science teacher administered EI to 6th-graders in a low-SES classroom (L1). Using the same variables in a forward regression, only reading comprehension was significantly related to ramps posttest scores ($r = +.87, p < .001$). In both cases, because instruction was presented orally by the teacher or with audio voice-over in TED, we believe that a more general comprehension skill may underlie the relationship between reading comprehension and near transfer than reading ability per se. These results are summarized in Table 2.

Table 2. Assessment and standardized test correlates of transfer performance

Assessment	(L1) (classroom instruction)	(L2 & L3) (Human & TED)
Near transfer (ramps post)	Reading comprehension	Reading comprehension
Immediate far transfer	(n/a)	Deductive reasoning
Delayed far transfer	Deductive reasoning	Story post & Deductive reasoning

Predictors of far transfer. Of both pretest and all standardized measures (reading comprehension, science, nonverbal “IQ,” and verbal/deductive reasoning), only deductive reasoning was significantly related to the measure of immediate far transfer ($r = +.58, p = .006$). Similarly, when immediate story posttest was also included in a forward regression, only deductive reasoning and immediate story posttest were significantly related to delayed story posttest score ($r = +.49, p = .04$, for both variables). Likewise, in L1, including all these independent variables (with the exception of the immediate story posttest, not administered for L1), only deductive reasoning was significantly related to the delayed story-evaluation posttest ($r = +.82, p = .001$).

Deductive reasoning may play a role during learning of CVS that may account for far transfer. Prior research [6] has found that conceptually-oriented explanations are predictive of procedural transfer and higher quality explanations were positively related to performance. Additionally, Kuhn and Dean [7] speculated that helping students to understand why to use CVS is critical metastrategic knowledge necessary for transfer. Therefore, we performed a finer-grained analysis and coded for students’ “highest quality” responses—those that demonstrated a complete understanding of the determinate nature of an unconfounded set-up (or the indeterminate nature of a confounded experiment), or “CVS logic.” For example, when given the probe: “Imagine the balls rolled different distances. Could you tell for sure that the surfaces caused the difference?,” one TED-tutored student responded: “Yes. Because everything is the same and if there is a difference it’s because of the surface.” This response explicitly demonstrates an understanding of the determinate causal link between the focal variable and outcome. In contrast, the following response to the same probe, though correct, is of lower quality because it does not explicitly express the causal link between the variable and outcome differences: “Yes. Because everything is the same except [surface].”

Deductive reasoning—as assessed in the TerraNova—could be related to the quality of explanation because both involve integrating and drawing conclusions from given information. Whether or not students explicitly expressed this causal logic during the experimental evaluation portion of instruction was more highly related to their deductive reasoning scores than any other pretest, standardized measure, or correct responses to questions posed during the “EI” component. Furthermore, deductive reasoning was more highly related to an expression of causal logic than other coded measures. Thus, expression of CVS logic and deductive reasoning appear to be highly inter-related.

Regarding near transfer performance, when reading comprehension and ramps pretest scores were included in the regression, there was no correlation between student expression of the logic of CVS and ramps posttest. Thus, this deeper understanding did *not* predict near transfer performance. However, with both deductive reasoning and expression of the causal logic in the regression model, only whether students expressed CVS logic during the experimental evaluation phase was significantly related to immediate far transfer performance. These correlations are shown below:

Deductive reasoning → CVS logic → Immediate & Delayed story posttest

Thus, CVS logic understanding—assessed during instruction—may explain the relationship between deductive reasoning and far transfer. Similarly, in a regression with immediate story posttest and deductive reasoning as covariates, delayed far transfer was only predicted by whether or not students explicitly expressed CVS logic, and not by immediate story posttest performance or deductive reasoning. Nor did it interact with condition, and thus was predictive of far transfer performance for both human- and TED-tutored students as anticipated. Thus, again this measure of deep conceptual understanding was predictive of far transfer performance, and may play a causal role in it. Note that if knowledge integration is primarily related to far transfer, this may explain why self-explanation prompts did not improve near transfer CVS performance in a recent study [8].

3 Pilot Study

For an initial test of whether CVS logic understanding improves far transfer performance, we compared students given “basic” TED-delivered CVS instruction to students who were additionally prompted to think about the link between the experimental setup and the conclusions that could be drawn about causality (i.e., the logic of CVS)³. If this deep understanding is related to far transfer, then students given the added prompt should be more likely to express CVS logic and out-perform control students on the story posttests.

3.1 Participants, Design, and Procedure

Participants were 8th-grade students in one science class at a local magnet school participated in this pilot study. The majority of students in this school (69%) were eligible for free or reduced-price lunch; thus, we consider this a low-SES population.

³ This was done as part of a larger study comparing TED instruction to a control lesson on CVS.

Of the 22 students who completed the pretest, after removing from analyses students who: (a) did not have parental permission for data use, (b) showed incoming CVS mastery, (c) did not have available reading levels, and (d) were absent for some of the instruction, only 11 students remained. Given the small sample, the results presented below, although quite interesting, are only suggestive at this point.

Table 3. One evaluation cycle of explicit instruction “EI” phase by condition ^a

Baseline	Added-questions (AQ)
Exp 1: Unconfounded experiment (focal variable = surface)	
Q1: “Is this experiment a good way to find out whether the balls go different distances just because of the ramp surface?” (Y/N)	
Q2: “Why or why not?” (typed response)	
Q3: “Imagine the balls rolled different distances. Could you tell for sure that the surfaces caused the difference?” (Y/N)	
Q4: “Why or why not?”	
(A3 Feedback: You’re right/Actually, we <u>could</u> tell for sure from this comparison whether changing the surface (or making the surfaces different) causes a change in how far the balls roll.)	
E5: “ <u>The reason we could tell for sure is that the only thing different between these two ramps is the surface.</u> One is [value 1] and the other is [value 2]. The ramps are built exactly the same way, except for the surface.”	Q5: “What else besides the [focal variable] could have made the balls roll different distances?” (<i>student selects one or more variables</i>) (Feedback on A5: “Right/Actually, <u>ONLY the different surfaces could have caused one ball to roll farther than the other, because only the surfaces are different between the two ramps.</u> ”)
E6 (CVS logic explanation): “The <u>ONLY</u> thing that is different is the thing Amal is trying to find out about. Everything else is the same. They have the same slope, the same ball, and the same starting position. <u>If one of the balls rolled farther, Amal would know that it could only be the surface that caused this result, since it’s the only thing different between the two ramps.</u> Amal could say whether the surface affects how far the balls roll. So, Amal made a <u>GOOD</u> experiment!”	

^aThis cycle was repeated twice for each of two initially confounded experiments—once for confounded state and once for fixed unconfounded state, for a total of 5 added questions.

Students were randomly assigned to the “baseline” or “added-questions” (AQ) condition in a two-condition, between-subjects design. Each phase of the procedure took place during the final period of the school day, their regular science class. The procedural sequence is the same as described in the introduction. On the first day, all students completed the computerized story pretest (described earlier). The EI phase was split between the second and third days. On the second day, students evaluated the first (unconfounded) experiment and received explicit instruction on experimental design. On the third day, students completed the EI phase by evaluating and receiving explicit instruction on the second and third experimental designs, both confounded. Thus, students in both conditions evaluated the same number of experiments.

The instructional script of the EI phase for the first experimental evaluation is shown in Table 3. Students in both conditions received the same questioning until after Q4. At that point, students in the baseline condition were given a procedural

explanation for why the unconfounded design was good. Students in the AQ condition were prompted to identify potential causal factors in the set-up and given feedback on their response. Then all students were told why the experiment would allow them to determine whether the focal variable was causal (i.e., the logic of CVS).

After the EI phase on the third day, students viewed a brief summary of the lesson and first completed the ramps posttest, then the story posttest and answered four standardized questions on paper. Finally, three weeks later, students completed a paper version of the delayed story posttest, followed by the four standardized CVS test items. The delayed story posttest was identical to the story pretest and immediate posttest, but targeted different focal variables. (Due to space constraints, we will not discuss the results of the standardized items in detail here, other than to note that there were no significant differences between conditions on this measure).

4 Results

Understanding of CVS logic. Students' responses during the EI phase of the intervention were coded for understanding of CVS logic, that is, whether they included expressions of the causal indeterminacy of confounded experiments or the determinacy of unconfounded experiments. Counter to expectation, students in the AQ condition were no more likely to give at least one CVS logic statement in the EI phase than students in the baseline condition (4 of 5, and 3 of 6, respectively), Fisher's exact $p = .55$. Nor were they more likely to justify their designs on the ramps posttest in terms of causal logic: no student in either condition did so.

Near and far transfer. As expected, the majority of students (80% and 89% in baseline and AQ conditions, respectively) achieved near-transfer mastery. These rates did not differ and were similar to those of middle-SES 5th-graders (Table 1).

Table 4. Story test means (and standard deviations) and mastery rates by time

Condition	Immediate	Delayed	Immediate mastery rate ^a	Delayed mastery rate ^a
Added-questions	4.17 (1.33)	5.17 (1.60)	33%	67%
Baseline	3.00 (2.45)	2.75 (2.75)	40%	25%

^a At least 5 out of 6 CVS set-ups.

Students who were asked the additional questions in the TED tutor did not significantly out-perform students in the baseline condition on the immediate story posttest, $F(1, 8) = 0.16$, $p = .70$ (Table 4). Because the prior period ran late, students had less time than they may otherwise have taken on the immediate story posttest. Students in the AQ condition may have been even more rushed: they spent less time on the immediate story posttest than students in the baseline condition ($M = 5.97$ min, $SD = 2.26$; $M = 9.03$ min, $SD = 3.21$, respectively), $F(1, 10) = 3.80$, $p = .08$. With time on immediate posttest included in ANCOVA, there was a nearly significant condition by reading level interaction, $F(1, 6) = 5.80$, $p = .05$. Lower-reading students (basic and below basic) tended to perform better in the AQ condition whereas the higher-reading students (proficient and advanced) tended to perform better in the baseline condition. This suggests

adapting TED instruction to students' reading level by assigning lower-reading students to the AQ version and higher-reading students the baseline version of EI.

With respect to delayed far transfer performance, though students in the AQ condition tended to score higher than those in the baseline condition (Table 4), this difference missed significance, $F(1, 7) = 3.44, p = .11$. Failure to reach significance may be due to small sample size. However, the AQ students showed significantly higher story test gains from the immediate to delayed posttest, $F(1, 11) = 5.17, p = .04$, where only students who answered the added questions showed significant immediate to delayed posttest gains. In sum, though students in the AQ condition were no more likely to express the logic of CVS during the EI, they tended to perform better on the far transfer assessments. Because this result does not support the hypothesis that CVS logic understanding causes far transfer, we sought to determine which factors were predictive of far transfer.

Predictors of far transfer. The factors we investigated in pair-wise correlations were story pretest, ramps pretest score, expression of CVS logic, the number of correct responses to Q1 in Table 3 (“Is this a good way...”) and Q3 (“Can you tell for sure...”), and reading level. No other standardized measures (e.g., deductive reasoning) were available.

In the baseline condition, reading level and expression of CVS logic were significantly related to immediate far transfer performance ($r = +.88, p = .048$; $r = +.95, p = .02$, respectively). When both expression of CVS logic and reading were included in a backward (or forward) step-wise regression, only expression of CVS logic remained in the model. However, no other factors were predictive of expression of CVS logic. Only immediate posttest score was significantly related to performance on the delayed story posttest ($r = +.99, p = .01$). These correlational links for students in the baseline condition, used to derive adaptive rules in the TED tutor, are shown below:

CVS logic \rightarrow Immediate story posttest \rightarrow Delayed story posttest

In the AQ condition, using the same variables as above but also including the number of correct responses to the added questions, neither expression of CVS logic nor reading level was related to far transfer performance. Rather, the number of correct responses to Q1 & Q3 (Table 3) was the best predictor of immediate far transfer performance ($r = +.87, p = .002$). In turn, reading level was the best—and only significant—predictor of number of correct Q1 and Q3 responses ($r = +.95, p = .004$). For the delayed story posttest, the number of correct responses on Q1 and Q3, the number of correct responses on the added questions, and the immediate story posttest were all significantly correlated with delayed story posttest performance ($r = +.93, p < .001$; $r = +.88, p = .002$; $r = +.82, p = .006$; $r = +.92, p = .001$, respectively). In both a forward and backward step-wise regression with these variables, only the number of correct responses on Q1 and Q3 remained in the model. These correlations for students in the AQ condition are shown below:

Reading \rightarrow Correct responses on Q1 & Q3 \rightarrow Immediate & Delayed story posttest

The number of correct responses to Q1 and Q3 was only significantly related to immediate and delayed far transfer performance for students in the AQ condition. It may be that correct responses to these items indicate a deeper understanding for students in the AQ than in the baseline condition (and incorrect responses may indicate greater confusion). However, the significant relationship between reading and number of cor-

rect responses to Q1 and Q3 questions may indicate that students with poorer comprehension skills do not understand the content of the EI as well as they might.

5 Implications for the TED Tutor

In previous studies with low-SES 5th and 6th-grade students, reading comprehension was the best predictor of near transfer performance. This relationship was not found for the 8th-graders, likely because they were near ceiling on the ramps posttest. We believe that one way to address the needs of students with poorer comprehension skills is to reduce cognitive load by reducing the amount of information students must process in a given conversational turn. Previous work [9] found that shorter tutor turns were related to better posttest performance, especially for students with poorer comprehension skills. Consistently, in our earlier work that included human tutoring of children who failed to learn CVS from the “EI” phase, we found that presenting the rules of CVS in a more incremental way—while still emphasizing the rationales for applying them—often helped students to develop a robust understanding of CVS. Thus, students with lower reading comprehension scores may be given more incrementally-delivered training on controlling non-focal variables before entering the “EI” phase of the tutor (“Rule 3 training” in Fig. 1). In this training, students are asked to select values for one non-focal variable at a time and receive immediate procedural and conceptual feedback on their responses.

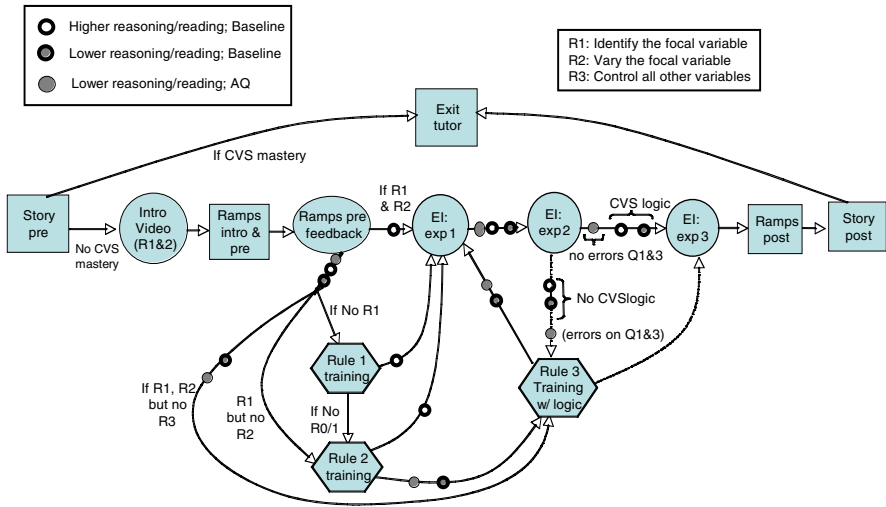


Fig. 1. Adaptive pathways with current and future components of the TED tutor

If the Bayesian knowledge-tracing model detects that a student likely lacks R1 or R2, remedial training can be given on these concepts (Fig. 1). Students with better reasoning or comprehension skills can then be ushered into the (baseline version of) the EI, whereas students with poorer skills can first be provided with the more

incremental Rule 3 training prior to entering the EI phase. Students with poorer reading/reasoning skills, who tended to perform better in the AQ condition, may then go to this version of the EI.

As found in earlier studies, expression of CVS logic was the best predictor of far transfer performance for students in the baseline condition. Thus, students in this version who do not express the logic of CVS by the end of the second experimental evaluation can be directed into the R3 training module to ensure they understand why to control.

In the AQ condition, the number of correct responses to the Q1 and Q3 questions was predictive of far transfer performance. It is possible that students in the AQ condition of the pilot study were more likely to understand the logic of CVS yet failed to express it in their responses during EI, perhaps because they thought this concept was more obvious than students in the baseline condition. Students who answer these questions incorrectly may be diverted to R3 training (Fig. 1).

Evidence of the effectiveness of these modifications would include both improved near transfer performance and a weaker relationship between reading comprehension and near transfer performance than was found in the past. It would also include a greater proportion of students in the baseline version expressing an understanding of the logic of CVS, and in the AQ version, a weaker link between reading and correct responses to the EI evaluation questions and more correct responses. But of course, the ultimate test of the effectiveness of the chosen adaptations is whether a greater percentage of students of varying backgrounds and characteristics develops a robust concept of CVS.

References

1. Chen, Z., Klahr, D.: All Other Things Being Equal: Children's Acquisition of the Control of Variables Strategy. *Child Dev.* 70(5), 1098–1120 (1999)
2. Strand-Cary, M., Klahr, D.: Developing Elementary Science Skills: Instructional Effectiveness and Path Independence. *Cog. Dev.* 23(4), 488–511 (2008)
3. Klahr, D., Nigam, M.: The Equivalence of Learning Paths in Early Science Instruction: Effects of Direct Instruction and Discovery Learning. *Psych. Sci.* 15(10), 661–667 (2004)
4. Klahr, D., Li, J.: Cognitive Research and Elementary Science Instruction: From the Laboratory, to the Classroom, and Back. *J. of Sci. Ed. and Tech.* 4(2), 217–238 (2005)
5. Toth, E., Klahr, D., Chen, Z.: Bridging Research and Practice: A Cognitively-based Classroom Intervention for Teaching Experimentation Skills to Elementary School Children. *Cog. & Instr.* 18(4), 423–459 (2000)
6. Matthews, P., Rittle-Johnson, B.: In Pursuit of Knowledge: Comparing Self-explanations, Concepts, and Procedures as Pedagogical Tools. *J. of Exp. Child Psych.* 104(1), 1–21 (2009)
7. Kuhn, D., Dean Jr., D.: Is Developing Scientific Thinking All About Learning to Control Variables? *Psych. Sci.* 16(11), 866–870 (2005)
8. Sao Pedro, M., Gobert, J., Heffernan, N., Beck, J.: Comparing Pedagogical Approaches for Teaching the Control of Variables Strategy. In: Taatgen, N.A., van Rijn, H. (eds.) *Proceedings of the 31st Annual Conference of the Cognitive Science Society*. Cognitive Science Society, Austin (2009)
9. Siler, S., VanLehn, K.: Learning, Interactional, and Motivational Outcomes in One-to-one Synchronous Computer-mediated Versus Face-to-face Tutoring. *Inter. J. of Artif. Intell.* in Ed. 19(1), 73–102 (2009)

Moodle Discussion Forum Analyzer Tool (DFAT)

Palak Baid, Hui Soo Chae, Faisal Anwar, and Gary Natriello

EdLab Development & Research
525 West 120th Street, 5th Floor Russell Hall
New York, NY 10027, USA
{pb2358, hsc2001, fa2227, gjn6}@columbia.edu

Abstract. In this paper we discuss the development of the Discussion Forum Analyzer Tool (DFAT) for the Moodle Learning Management System (LMS). This application was designed to enhance student engagement in course discussion forums by emphasizing the participation of fellow learners and by making discussions more efficient and pertinent to students. In addition, by mining user-submitted content and applying learning algorithms, DFAT “tags” replies/comments as relevant or non-relevant, and recommends other related discussions and readings.

Keywords: discussion, learning, interactive, motivation, data mining, LMS.

1 Introduction

Discussion forums are an important tool in Learning Management Systems (LMS) like Moodle [1] and Blackboard. They provide instructors with a mechanism to assess learning, evaluate student engagement, and gain insight into students’ thinking [2]. For students, discussion forums provide an environment to exchange knowledge, request assistance, demonstrate understanding, and pose questions [3]. However, active user participation (i.e., replying to and posting new content frequently) from students, instructors and teaching assistants is central to an effective discussion forum.

A review of 5 discussion forums in an online course at Teachers College Columbia University (TC CU) revealed that only 8/30 students participated more than five times in a single discussion, and that 15/30 students read but did not participate in any discussion. Furthermore, 7/30 students did not use the discussion forum.

To encourage increased participation in discussions we developed DFAT. This tool emphasizes social learning [4], reduces distractions to aid focus [5], and highlights the relevance of specific forum content to students [6].

1.1 Activating Social Forces

To make more salient the participation of others in discussions we propose:

- A public display of user activity (i.e., total number of posts, total number of replies) for a single forum and/or multiple forums. This allows users to learn the importance of the forum among other users.

- Weekly reports (generated automatically) that inform users about the activities in the discussion board. The reports will also recommend posts and comments that a specific user might find relevant to their interests as determined by their submissions and activity in the discussion forum.
- Email updates summarizing user activity in the discussion forum. This email will also highlight comments and replies that were made specifically to the email recipient. This is intended to direct users back to the discussion forum for continued dialogue.

1.2 Supporting Focus and Efficiency

To minimize distractions and support focus for efficient learning we propose:

- A heat map, to differentiate active and non-active threads within a discussion. This allows users to focus their attention to popular and active forums.
- A dashboard containing information about most viewed, replied to, and recommended discussion threads on the board. This draws users' attention to the specific information that is suggested as helpful readings.
- Filtering tools that enable users to sort information based on topics, number of replies, and views. This can be done for a single or multiple discussions over a specified time range to improve user experience.
- Quick links to discussions of user interest, generated automatically by analyzing user history of participation and reading activity. This builds on demonstrated user interest.

1.3 Enhancing Relevance

To increase the relevance of forum activities to students, we propose:

- Automatically recommending articles and readings corresponding to each discussion, to facilitate user's access to detailed information about the topic.
- Automatically recommending other discussions that match a particular discussion, to redirect users to similar discussions for additional learning.
- Marking replies and comments as relevant and non-relevant to allow users to focus their attention on more germane replies rather than all comments.
- Informing a user about similar discussions while creating a new discussion post. This allows users to build on pre-existing information in a forum, instead of replicating content in a separate posting.

2 Current Implementation

Thus far we have implemented five of the aforementioned features of DFAT in the Moodle LMS at TC CU. Specifically, we modified the current discussion board to include: 1) a heat map; 2) a list of participants and non-participants for single and multiple discussions; 3) sort options for each discussion board by name, number of replies, and views; and 4) displays of active discussions based on both number of replies and number of views. 5) All selections and displays can be filtered over a time range.

3 Conclusion

DFAT has the potential to make discussion forums more useful, interactive, and personalized for individual users. We believe these changes will: 1) encourage users to further participate and contribute to discussions, 2) improve their experience by focusing on important discussions, and 3) facilitate learning about topics of their interest. Furthermore, it will assist instructors by providing real-time records on student participation in discussions, as well as longitudinal data over the course period. The automated recommendation tools proposed also have the potential to motivate users to engage in more self-directed learning opportunities. A future version of DFAT can also be extended to recommend courses to a user based on his topics of interests in the discussions. However, a study to determine the impact of DFAT on discussion board activity is necessary before continuing future development. We anticipate conducting such an analysis in summer 2010.


References

1. Cole, J., Foster, H.: *Using Moodle: Teaching with the Popular Open Source Course Management System*. O'Reilly Media, Sebastopol (2007)
2. Bye, L., Smith, S., Rallis, H.M.: Reflection Using an Online Discussion Forum: Impact on Student Learning and Satisfaction. *Soc. Wk. Edu.: The Intl. Jrl.* 28, 841–855 (2009)
3. Patel, J., Aghayere, A.: Students' Perspective on the Impact of a Web-based Discussion Forum on Student Learning. In: 36th *Frontiers in Education Conference*, pp. 26–31. IEEE Press, San Diego (2006)
4. Bandura, A.: *Social Learning Theory*. General Learning Press, New York (1977)
5. Caputo, G., Guerra, S.: Attentional selection by distractor suppression. *Vis. Res.* 38, 669–689 (1998)
6. Kember, D., Ho, A., Hong, C.: The importance of establishing relevance in motivating student learning. *Actv. Lrng. in Hhr. Ed.* 9, 249–263 (2008)

Peer-Based Intelligent Tutoring Systems: A Corpus-Oriented Approach

John Champaign and Robin Cohen

David R. Cheriton, School of Computer Science
University of Waterloo, Waterloo, ON, Canada
{jchampai,rcohen}@uwaterloo.ca

Abstract. Our work takes as a starting point McCalla's proposed ecological approach for the design of peer-based intelligent tutoring systems and proposes: (i) to develop an algorithm for selecting appropriate content (learning objects) to present to a student, based on previous learning experiences of like-minded students (ii) to build on this research by also having students leaving explicit annotations on learning objects to convey refinements of their understanding to subsequent students; the challenge is to intelligently match students to those annotations that will be most beneficial for their tutoring (iii) to develop methods for intelligently extracting learning objects from a repository of knowledge, in a manner that may be customized to the needs of specific students (iv) to apply our work to the specific application of assisting health care workers via peer-based intelligent tutoring, primarily for homecare environments 

Keywords: peer-based intelligent tutoring, simulating students, corpus-based ITS development, ecological approach to instructional design, modeling learners.

1 Content Sequencing

Two central challenges in the design of intelligent tutoring systems are compiling the material for the lessons and determining the best methods to use, for the actual teaching of those lessons. We observe in particular that it is desirable to provide a framework for determining the material to be taught that does not rely on experts hand-coding all the lessons and deciding how they should be sequenced. Indeed, that particular approach presents considerable challenges in time and effort. We are interested in techniques for bootstrapping the system in order to initiate peer-based learning and in developing robust methods for validating the models that are presented (including the technique of employing simulated students). Once the content is in place, our efforts will be aimed at refining our model in order to enable students to benefit the most from the learning that their peers are undergoing.

We have currently developed an algorithm for reasoning about the sequencing of content for students in a peer-based intelligent tutoring system inspired by

¹ Thanks to NSERC for funding and to Gord McCalla for helpful advice.

McCalla’s ecological approach [1]. We record with each learning object those students who experienced the object, together with their initial and final states of knowledge, and then use these interactions to reason about the most effective lessons to show future students based on their similarity to previous students. As a result we are proposing a novel approach for peer-to-peer intelligent tutoring from repositories of learning objects.

We used simulated students to validate our content sequencing approach. Our motivation for performing this simulation was to validate that, in the experimental context, our approach leads to a higher average learning by the group of students than competing approaches. We added a modeling of the knowledge that each object is aimed at addressing (for example, an object in a first year computer science course may be aimed at addressing the knowledge of recursion). By abstracting all details from the intelligent tutoring system and the student, we defined a formula to simulate learning (Equation 1).

$$\Delta UK[j,k] = \frac{I[l,k]}{1 + (UK[j,k] - LOK[l,k])^2} \quad (1)$$

where UK is the user j ’s understanding of knowledge k , I is the educational benefit (how much it increases or decreases a student’s knowledge) of learning object l on knowledge k and LOK is the learning object l ’s target level of instruction for knowledge k . When running our algorithm in the simulation, each student would be presented with the learning object that was expected to bring the greatest increase in learning, determined by extracting those learning objects that had resulted in the greatest benefit for previous students considered to be at a similar level of understanding as the current students [2].

Simulated students allowed us to avoid the expense of implementing and experimenting with an ITS and human students to see the impact of our approach in contrast with alternative approaches. In particular we contrasted our method with a baseline of randomly assigning students to learning objects and to a “look ahead” greedy approach where the learning was precalculated and used to make the best possible match. One variant we considered was a “simulated annealing” inspired approach, where greater randomness was used during the initial, exploratory phase of the algorithm, then less randomness was used once more information about learning objects had been obtained. We discovered that our approach showed a clear improvement over competing approaches and approached the ideal.

2 Annotation

To extend the basic evolutionary approach, we are particularly interested in exploring the use of student annotations, which would fit naturally with our

² Each student in the simulation is modeled to have a current level of understanding for each possible knowledge area, a value from [0,1] reflecting an overall grade from 0 to 100. Simulated students are randomly assigned an initial set of knowledges and are not modeled by training on human data.

proposed corpus-based design for the lesson base. Student annotations on learning objects would involve allowing students to leave short comments on lessons they are interacting with (e.g. “Functions and procedures are really similar” for an introductory computer science course). Subsequent students would identify which annotations they found useful, which would then be intelligently shown to similar students. The idea behind this is allowing students to “collaborate” with one another but not in real time (or, at least, to allow the interactions of the student in the past to inform the interaction with the current student, which honours the ecological approach [1]). There will be a decision theoretic reasoning element to this, when “low quality” annotations should be shown as part of a dialogue involving high quality annotations, and some trust modeling in addition to student modeling similar to what we are advocating for Content Sequencing.

To date we have only developed preliminary steps towards an overall algorithm for reasoning about annotations. We have not yet explored how best to validate our approach.

3 Corpus-Based

We are interested in exploring the construction of the lesson base that forms the centrepiece of a peer-based intelligent tutoring system, and are concerned with facilitating the authoring of such a lesson base, through the mining of existing repositories of information. This stands in contrast to McCalla’s ecological work [1], which assumes that learning objects are already created and available to the system. This would be especially useful for applications where large repositories of information already exist, possibly employing varied forms of media, that could be leveraged for the creation of an ITS. Towards this end we have been exploring scenarios applicable to peer-based home healthcare assistance for caregivers or patients. Working in conjunction with health care workers affiliated with our hSITE (Healthcare Support through Information Technology Enhancements) project (an NSERC Strategic Research Network), our aim is to design a system that will be of some benefit in actual healthcare environments.

For future work, we are interested in refining learning objects created for a variety of purposes (e.g. book chapters, instructional videos or research papers) by combining the original learning object with a student model. The aim is to separate the most important information in the object, potentially breaking it into multiple, more targeted learning objects based on what students would find most relevant. A second path for future work is to identify learning objects currently missing from an existing ITS which could be deployed for pedagogical benefit.

Reference

1. McCalla, G.: The Ecological Approach to the Design of E-Learning Environments: Purpose-based Capture and Use of Information About Learners. *Journal of Interactive Media in Education: Special Issue on the Educational Semantic Web* 7, 1–23 (2004)

Intelligent Tutoring Systems, Educational Data Mining, and the Design and Evaluation of Video Games

Michael Eagle and Tiffany Barnes

University of North Carolina at Charlotte, Department of Computer Science,
Charlotte, North Carolina
{mjeagle, tiffany.barnes}@uncc.edu

Abstract. Technological support for personalized learning has the potential to *transform the educational system* in the United States. There is a growing interest in educational games and their potential for motivating learners. Techniques from the educational data mining and intelligent tutoring systems communities can be leveraged to better understand, design, and evaluate educational games for both learning effectiveness and learner engagement. This work explores the use of intelligent feedback in games as well as the potential pitfalls; it concludes with a proposed study designed to explore the differences between intelligent tutoring systems and educational video games.

Keywords: Intelligent Tutoring Systems, Educational Data Mining, Games.

1 Introduction

Educational games are receiving widespread attention for their potential to engage and motivate students in learning, but little is known about how best to take advantage of game technologies to support learning [1]. Studies have defined qualities that can be leveraged to improve education [1, 2] but the study of games does not have a coherent research paradigm [3]. In a recent study of 55 educational games, only 22 of these were found to be designed based on pedagogical theories [4]. On the other hand, extensive research has shown that intelligent tutoring systems can be very effective in improving student performance through personalization and support for learning. For example, the PUMP Algebra intelligent tutor, used in thousands of schools, improves student performance on standardized tests by 15—25% [5]. We argue that methods for the exploration of data in intelligent tutoring systems can provide important insight into the reasons for their effectiveness for learning, and that these same educational data mining methods can be applied to educational games as a coherent and scientific way to understand and evaluate their effectiveness.

There are some important parallels between Intelligent Tutoring Systems and video games such as instant feedback and scaffolding techniques [6, 7]. There is a call for finding how we can best leverage the Intelligent Tutoring System and game communities for future educational learning environments [8]. In this paper, we make an argument for the incorporation of intelligent tutoring system research in the design and evaluation of educational video games. We also propose an experiment that will

shed light on the differences in student learning between educational video games and intelligent tutoring systems.

2 Intelligent Tutoring Systems and Video Games

Games and intelligent tutoring systems often keep track of similar user log-data; both systems have rapid feedback loops. Educational data mining provides way to explore this information. One example of the techniques is generating and evaluating *empirical learning curves*, by plotting problem-solving speed and accuracy over time, to understand how learning occurs in educational video games; *learning curves* are used to evaluate intelligent tutoring systems [9], and can also be used to evaluate educational video games. [6, 10]

According to the theory of flow [11], what makes games entertaining is constant learning and increasing challenges. Educational data mining on student game-log data allows game developers to model student learning and identify places where we can improve flow and learning in educational games. Educational games could also benefit from advancements in student modeling, such as keeping students in flow by dynamically adapting to individual players as proposed in [7].

However, there are important problems could arise from the use of video game technologies. One potential problem with educational video games is the violation of the coherence principle from working memory; laboratory studies have found that extraneous material distracts from learning [12]. Including extra information in the form of interesting characters and locations could distract from the learning objectives of the system. However, in a classroom experiment run by Muller, Lee, and Sharma they found that the addition of approximately 50% of extraneous, but interesting, material did not significantly affect learning performance. [13] Therefore, it could be possible to harness the motivational aspects of video games while not reducing the educational value of the intervention.

More study is needed to compare the results of video game environments and how they differ from traditional intelligent tutoring systems. Experiments such as [8] have done some study into comparing intelligent tutoring systems and educational video games. However, there is little quantitative evidence presented that shows the game and intelligent tutoring system used in these studies to be directly comparable. Without some kind of reliability or validity study on the game and the intelligent tutoring system it is difficult to directly compare things such as learning efficiency, affect, and educational value of game elements.

To study the effects of educational video games on learning and the differences between intelligent tutoring systems we propose an experiment involving three conditions using a previously developed educational video game *Wu's Castle* [14], a comparable intelligent tutoring system, and a comparable control condition. We will insure the interchangeability of the tutoring system and video game by running a parallel form reliability analysis. We can control for the pedagogical content and explore the differences in motivation, retention, and enjoyment between games and tutoring systems; this allows future developers to harness the best parts of both intelligent tutoring systems and educational games for improved education.

3 Conclusion

Intelligent tutoring systems and educational video games could have profound effects on education. Educational game research can benefit from incorporating evaluation and development principles from the more established field of intelligent tutoring systems. Experiments comparing the learning effects between intelligent tutoring systems, game environments, and traditional educational materials will provide insight into the educational value of video games as well as the effects of deviating from the coherence principle on learning results. These experiments need strong controls and a highly comparable intelligent tutoring system, educational video game, and traditional education material.

References

1. Squire, K.: Video Games in Education. *International Journal of Intelligent Simulations and Gaming* 2, 49–62 (2003)
2. Prensky, M.: *Digital game-based Learning*. Computers in Entertainment (2003)
3. Hunnicke, R., Robison, A., Squire, K., Steinkuehler, C.: Games, learning, and literacy. In: *Sandbox*. ACM Press, New York (2006)
4. Kebritchi, M.H.: A Examining the pedagogical foundations of modern educational computer games to inform research and practice. *Computers & Education* 4, 1729–1743 (2008)
5. Koedinger, K.R., Corbett, A.T.: Cognitive tutors: Technology bringing learning science to the classroom. In: Sawyer, K. (ed.) *The Cambridge Handbook of the Learning Sciences*. Cambridge University Press, Cambridge (2006)
6. Baker, R., Habgood, M., Ainsworth, S., Corbett, A.: Modeling the Acquisition of Fluent Skill in Educational Action Games, pp. 17–26 (2009)
7. Thomas, J.M., Young, R.M.: Dynamic Guidance in Digital Games. In: *International Conference on Artificial Intelligence in Education*, pp. 107–114. IOS Press, Amsterdam (2009)
8. Rodrigo, M. M.T., Baker, R.S.J.d., D’Mello, S.K., Gonzalez, M. C.T., Lagud, M.C.V., Lim, S.A.L., Macapanpan, A.F., Pascua, S.A.M.S., Santillano, J.Q., Sugay, J.O., Tep, S., Viehland, N.J.B.: Comparing learners’ affect while using an intelligent tutoring system and a simulation problem solving game. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 40–49. Springer, Heidelberg (2008)
9. Martin, B., Koedinger, K., Mitrovic, A., Mathan, S.: On Using Learning Curves to Evaluate ITS. In: *International Conference on Artificial Intelligence in Education* (2005)
10. Eagle, M.: Level up: a frame work for the design and evaluation of educational games. In: *International Conference on Foundations of Digital Games*, pp. 339–341. ACM, New York (2009)
11. Csikszentmihalyi, M.: *Flow: the Psychology of Optimal Experience*. Harper and Row, New York (1990)
12. Mayer, R.E.: *Multimedia learning*. Cambridge University Press, London (2001)
13. Muller, D.A.L., Kester, J., Sharma, M.D.: Coherence or Interest: Which Is Most Important in Online Multimedia Learning? *Australasian Journal of Educational Technology* 24, 211–221 (2008)
14. Eagle, M., Barnes, T.: Evaluation of a game-based lab assignment. In: *International Conference on Foundations of Digital Games*, pp. 64–70. ACM, New York (2009)

An Intelligent Debater for Teaching Argumentation

Matthew W. Easterday

Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA USA
{matt.easterday@cmu.edu}

Abstract. Despite the growing number of ITSs for teaching argumentation, few tutors actually debate the student. An *intelligent debater* allows the student to practice argumentation and provides the motivation to analyze evidence. Here I describe an *intelligent debater* used in Policy World to argue with students about policy recommendations. The *intelligent debater* forces the student to recommend a policy intervention, to describe how the intervention affects the desired policy outcomes, and to provide evidence. The debater then attacks infeasible recommendations, implausible mechanisms, and weak evidence. Key aspects of the intelligent debater algorithm are presented for those interested in using intelligent debaters in science, law, and history.

Keywords: Argumentation, Debate, Pedagogical agents, Causal reasoning.

Deliberative argument [1], e.g., “*We should limit junk food advertising on children’s television to decrease childhood obesity*” forces students to analyze evidence (the focus of many argument ITSs [2]) and is a goal in its own right. However, the majority of causal reasoning tutors for science and history [3-7] do not ask students to make such arguments, perhaps because of the difficulty of designing tutors for ill-defined tasks. Here, I present an *intelligent debater* that can argue with students.

The challenge is to create a microworld that allows the student to make deliberative arguments that the system can evaluate, rebut, (and eventually tutor). In the educational game Policy World, students make recommendations about policy problems. Students begin the game by searching for information such as causal claims in expert testimony or summaries of scientific reports about observational studies and experiments as might be seen on the science page of the New York Times. As the student analyzes this information, Policy World monitors the subset of causal claims found by the student, the student’s evaluation of the type and strength of those claims, and the student’s overall beliefs about which causal claims are best supported by the evidence. Once the student has finished searching for and analyzing evidence, he must then convince a judge to adopt his policy position by defeating a computer opponent in a debate. This debate tests three sets of deliberative argument skills: making a recommendation, explaining the mechanism by which the recommended intervention affects the desired outcome, and providing evidence for that mechanism.

Making a recommendation. In the beginning of the debate, the judge, played by the senator, asks the student to make a recommendation such as: “What should we do about childhood obesity?” The student then picks his intervention from a list that

includes increasing or decreasing the level of each variable in the domain model, e.g., “decreasing the number of junk food advertisements on children’s television,” or doing nothing. Once the student selects an intervention, the student’s avatar states the intervention, and the *debater*, (played by Mr. Harding) will oppose the recommendation. If the student selects a recommendation that cannot be intervened upon, e.g., “decrease genetic propensity for obesity,” then he receives a “strike” and will have to pick another recommendation before proceeding. After 3 strikes, the student receives tutoring.

Table 1. Debate dialogue showing a recommendation, explanation and evidence

Role	Dialogue
Judge	What do you recommend we do about childhood obesity?
Student	I think we should decrease the amount of junk food commercials seen!
Debater	That will never work!
Judge	How will decreasing junk food advertising seen affect obesity?
Student	Decreasing the number of junk food commercials seen will decrease the amount of junk food eaten, which will decrease obesity.
Debater	You must be joking. <i>Should the Government Regulate Junk Food Advertising?</i> shows that junk food commercials don’t affect the amount of junk food eaten!
Judge	What evidence do you have that the amount of junk food commercials seen increases the amount of junk food eaten?
Student	<i>Fighting Obesity: An Uphill Battle</i> shows that the increasing the amount junk food commercials seen increases the amount of junk food eaten!
Judge	Hmmm, your experimental data is stronger than Harding’s case study. You’ve proven your case. It looks like we should decrease junk food advertising.

Explaining mechanism. In the next stage of the debate, the judge asks the student to explain how his recommendation affects the outcome. The student uses drop down menus to construct a causal explanation using terms for the expert model. For example, the explanation begins with the student’s recommendation, e.g., “Decreasing junk food advertising...”. The student then selects an effect, e.g. “will increase...” or “will decrease...” and a second variable from the list of variables in the expert model, e.g. “the amount of junk food eaten.” The student continues to construct a chain of effects from his recommendation to the desired outcome. Once the student has constructed the main causal path of his explanation, e.g., “Decreasing junk food advertising will decrease the amount of junk food eaten which will decrease obesity,” he can then submit his explanation, or add additional mechanisms and outcomes. To add an additional mechanism, the student selects another starting variable, which may or may not already be included in his existing explanation, and continues to construct a causal path in the same way. After the student has submitted his explanation, the debater may attack the explanation on essentially syntactic grounds, e.g., if the student’s explanation does not include the outcome, or if the outcome is not the terminal variable on the causal path. In this case, the student receives a “strike” and must submit another explanation of his mechanism. If the mechanism is plausible, then the debate enters the evidence phase.

Providing evidence. At this point the debater attacks different parts of the student’s explanation. The debater will select one causal link in the student’s explanation, e.g., that “junk food advertising increases the amount of junk food eaten” then cite a report

contradicting the student's claim. The debater selects a link to attack in the following manner: it will first select a causal claim for which the majority of evidence opposes the student's position, it will next select causal claims for which the evidence is stronger than the amount of evidence the student actually collected, and finally, it will select causal claims randomly in the hope that the student will make a mistake defending the claim. Once the debater has attacked a claim, the judge will ask the student to defend it. The student then selects one or more reports from the list of reports that he collected before the debate. If he presents weak evidence and fails to defend a causal claim, he receives a strike. The student can then attempt to present evidence again, or modify his explanation or recommendation. If he successfully defends several attacks, he wins the debate.

Conclusion. This work contributes to the literature on ITS systems for argumentation by describing how to design an *intelligent debater* that can argue with the student, both to practice argumentation and to motivate the search and analysis of evidence supported by most argumentation tutors. Future work is empirically testing the educational and motivational impact of *intelligent debaters*.

Acknowledgements. This work was supported by a graduate training grant awarded to Carnegie Mellon University by the U.S. Department of Education (# 305B040063) and the Pittsburgh Science of Learning Center (National Science Foundation # SBE-0836012).

References

1. Walton, D.: Fundamentals of critical argumentation. Cambridge University Press, New York (2006)
2. Scheuer, O., McLaren, B.M., Loll, F., Pinkwart, N.: Automated analysis and feedback techniques to support argumentation: A survey. In: Pinkwart, N., McLaren, B. (eds.) Educational technologies for teaching argumentation skills. Bentham Science Publishers, Oak Park (in press)
3. Britt, M.A., Aglinskas, C.: Improving students' ability to identify and use source information. *Cognition & Instruction* 20(4), 485–522 (2002)
4. Forbus, K.D., Carney, K., Sherin, B.L., Ureel, L.C.: Vmodel: A visual qualitative modeling environment for middle-school students. *AI Magazine* 26(3), 63–72 (2005)
5. Graesser, A.C., Wiley, J., Goldman, S.R., O'Reilly, T., Jeon, M., McDaniel, B.: SEEK web tutor: Fostering a critical stance while exploring the causes of volcanic eruption. *Metacognition and Learning* 2, 89–105 (2007)
6. Leelawong, K., Biswas, G.: Designing learning by teaching agents: The Betty's Brain system. *International Journal of Artificial Intelligence in Education* 18(3), 181–208 (2008)
7. Masterman, L.: A knowledge-based coach for reasoning about historical causation. In: J. Breuker et al. (series eds.) Looi, C.K., McCalla, G., Bredeweg, B., Breuker, J. (vol. eds.) *Frontiers in artificial intelligence and applications. Artificial intelligence in education: Supporting learning through intelligent and socially informed technology*, vol. 125, pp. 435–442. IOS Press, Amsterdam (2005)

Multiple Interactive Representations for Fractions Learning

Laurens Feenstra¹, Vincent Aleven², Nikol Rummel³, and Niels Taatgen⁴

¹ University of Groningen, Human Machine Interaction, Groningen, The Netherlands
and Carnegie Mellon University, HCI Institute, Pittsburgh PA, USA

² Carnegie Mellon University, HCI Institute, Pittsburgh PA, USA

³ University of Freiburg, Institute for Psychology, Freiburg im Breisgau, Germany

⁴ University of Groningen, Artificial Intelligence, Groningen, The Netherlands

Abstract. Multiple External Representations (MERs) have been used successfully in instructional activities, including fractions. However, students often have difficulties making the connections between the MERs spontaneously. We argue that interactive fraction representations may help students in discovering relevant features and relating the MERs to one another. Support for guiding student interaction is provided by example-tracing tutors.

Keywords: Interactive fraction representations, virtual manipulatives.

1 Introduction

“Oh fractions! I know there are lots of rules but I can’t remember any of them and I never understood them to start with.” This utterance, by a middle-school student performing fraction operations, exemplifies the difficulties students have with understanding rational numbers [1]. Indeed, fractions are considered the most challenging topic in the elementary school curriculum [2].

Part of the challenge is the different conceptual interpretations of rational numbers: part-whole, percentage, ratio, measurement, division, decimal, etc. [3]. Using multiple external representations (circles, number lines, rectangles) besides the mathematical symbols, leads to deeper understanding of fractions [3].

The positive effect of instructional activities that combine MERs has been widely acknowledged [4][5]. Learners must detect relevant structures within representations and relate the different representations to each other. However students often do not make these connections spontaneously [6]. Accordingly, to facilitate learning with multiple fraction representations it seems to be important to support students in dealing with the requirements of connection making.

In a prior study on fractions learning conducted by our research group, students learned better with MERs, compared to working with a single graphical representation of fractions, when prompted to self-explain how each graphical representation relates to the standard fraction notation [7]. The representations in this study were presented as static graphics. In the current study, we want students to gain an even better understanding of the fraction concepts, by allowing them to actively manipulate the representations. These *interactive fraction representations* encourage exploration of the representational features, by requiring student action on the underlying

concepts (e.g. drag-and-drop equivalent fractions on top of each other, to signal the importance of equal proportion to the whole (see Fig. 1)). By doing so, student attention is directed to the relevant structures, supporting connection making between the different representations.

However, when learners explore interactive environments they are often not able to interact with them in a systematic and goal-oriented way [6][8][9]. To support and guide students in interaction, many solutions have been tried, such as an integration phase with static representations [8], integrating representations [6], step-by-step introduction of interactivity and dynamic linking of representations [9]. We will add additional support for interaction by embedding the interactive representations within a technology proven to improve student learning: *example-tracing tutors* [10], providing hints and feedback at every problem solving step and directing student attention by both visual cues and step-by-step instruction.

Reimer et al. [11] explored the use of interactive fraction representations (referred to as *virtual manipulatives*) in a classroom and found a positive learning effect. They did not however, compare its effect against static representations.

Equivalent fractions Hint

Find another fraction that is equivalent with one half and two fourths.

Well done! You see that one half has the same size as two fourths. We say that one half and two fourths are EQUIVALENT. Enter the fraction you just found below.

$$\frac{1}{2} = \frac{2}{4}$$

Hint: Now, you need to find your own equivalent fraction. Divide the circle into more pieces and drag pieces on top of the orange fourths.

Okay

get next hint

Fig. 1. Example-tracing tutors with interactive representations

2 Research Questions and Method

The goal of this study is to determine how students can best be supported in making connections between the MERs. More specifically: 1. *Are interactive representations more effective in supporting robust fraction learning compared to static representations?* And 2. *Should learners explicitly relate the representations?*

The experiment will have a 2x2 factorial design with four experimental conditions. One factor we differ is the graphical representations: interactive representations compared to static representations. The other factor is explicit connection making activities (multiple different representations in the same problem) versus implicit connection making (single representations only).

We expect that interactive representations will lead to better performance in reproduction and transfer items. Also, we expect tasks where students explicitly relating representations to lead to increased robust fractions learning. We believe that students assigned to the interactive and explicit conditions will show more learning than students in the static, implicit condition.

The study will be carried out in three elementary schools in the Penn-Trafford school district with 312 4th and 5th grade students. Students will be randomly assigned to the four experimental conditions within each of the participating classrooms. The web-based example-tracing tutors will cover a part of the fraction curriculum, including naming and constructing (graphical) fractions, fraction equivalence and ordering fractions. The (interactive) fraction representations include circles, rectangles and number lines. The pre-test and post-test will contain reproduction items and transfer-items (e.g. future learning items) in equal proportion.

The fractions tutors will be delivered over the web, and used as an adjunct to regular 4th or 5th-grade fractions instruction. Students will work on the tutor starting March 22nd 2010 for a total of five consecutive days, with on each day a one-hour session.

References

1. Moss, J.: Pipes, Tubes, and Beakers: New approaches to teaching the rational-number system. In: Brantsford, J., Donovan, S. (eds.) *How people learn: A targeted report for teachers*, pp. 309–349. National Academy Press, Washington (2005)
2. Carpenter, T.P., Ansell, E., Franke, M.L., Fennema, E., Weisbeck, L.: Model of problem solving: A study of kindergarten children's problem processes. *Journal for Research in Mathematics Education* 5(24), 428–441 (1993)
3. Steiner, G.F., Stoecklin, M.: Fraction calculation—a didactic approach to constructing mathematical networks. *Learning and Instruction* 3(7), 211–233 (1997)
4. Ainsworth, S., Loizou, A.T.: The effects of self-explaining when learning with text or diagrams. *Cognitive Science: A Multidisciplinary Journal* 4(27), 669–681 (2003)
5. Schnotz, W., Bannert, M.: Construction and interference in learning from multiple representation. *Learning and Instruction* 2(13), 141–156 (2003)
6. Ainsworth, S., Bibby, P., Wood, D.: Examining the effects of different multiple representational systems in learning primary mathematics. *Journal of the Learning Sciences* 11(1), 25–61 (2002)
7. Rau, M.A., Aleven, V., Rummel, N.: Intelligent Tutoring Systems with Multiple Representations and Self-Explanation Prompts Support Learning of Fractions. In: Dimitrova, V., Mizoguchi, R., du Boulay, B. (eds.) *Proc. of the 14th International Conference on AI in Education*, pp. 441–448. IOS Press, Amsterdam (2009)
8. Bodemer, D., Plötzner, R., Bruchmüller, K., Häcker, S.: Supporting learning with interactive multimedia through active integration of representations. *Instructional Science* 33, 73–95 (2005)
9. van der Meij, J., de Jong, T.: Supporting students' learning with multiple representations in a dynamic simulation-based learning environment. *Learning and Instruction* 16(3), 199–212 (2006)
10. Aleven, V., Koedinger, K.R.: An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science* (26), 147–170 (2002)
11. Third-Graders Learn About Fractions Using Virtual Manipulatives: A Classroom Study. *Journal of Computers in Mathematics and Science Teaching* 24(1), 5–25 (2005)

An Interactive Educational Diagrammatic System for Assessing and Remediating the Graph-as-Picture Misconception

Grecia Garcia Garcia and Richard Cox

Representation and Cognition Lab, School of Informatics, Sussex University, UK
gg44@sussex.ac.uk

Abstract. The graph-as-picture misconception (GAPm) is a commonly reported error, but basic questions about its prevalence and degree have not been addressed. An interactive educational diagrammatic system was designed to assess and to help students overcome GAPm. Three activities were administered to students (N=48) using touch screen technology: a diagram/picture decision task, a multiple choice questionnaire, and an interactive racing car game in which the child moves a car along a track and a speed/distance graph is plotted concurrently alongside. Preliminary results demonstrate the utility of the diagram/picture discrimination task for detecting GAPm and providing rich information about children’s knowledge of different representational forms.

Keywords: graph-as-picture misconception, diagrams, development, interactive environments.

1 Introduction and Research Questions

The Graph-as-picture misconception (GAPm) is a commonly reported error that students make when interpreting line graphs. A student demonstrates a GAPm when he/she interprets an abstract representation as the picture of a situation, e.g. a polynomial function might be interpreted as the picture of a “hill” or a “bridge”. Although the GAPm seems quite well known (e.g. [1]), questions about how to detect it and remediate it, how prevalent it is still need addressing.

The GAPm affects students at different school levels (e.g. [2], [3], [1]) and in different subject areas, e.g. maths, physics. Nevertheless, very little research on primary school students has been conducted and knowledge of young students’ proficiency with diagrammatic representations is limited. It is uncertain whether the GAPm actually occurs in other representational types, e.g. set diagrams. A survey of the UK National Curriculum [4] revealed examples where children are exposed to representations consisting of mixtures of diagrams and pictures. This raises the question of whether conflating abstract/metaphoric representations with pictorial/iconic representations might be causing students to develop the GAPm.

This study aims to extend Janvier's [2] work on GAPm by including a younger population and by using a different (information-processing) approach to GAPm detection instead of psychometric methods.

An educational diagrammatic interactive system was carefully designed as a remedial intervention for students with the GAPm. Using interactive touch screens, students race cars around variously shaped racing tracks. A speed distance graph is plotted dynamically in real-time via dynalinking ([5], [3]) between the racing car and the graph.

Process data was gathered via retrospective debriefing and concurrent verbal protocol recording. The computer-based tasks were also logged (diagram/picture discrimination task, student-car racing interactions).

2 Experimental Design and Apparatus

In an evaluation of the racing car task, two pre- and two post-intervention activities were administered consisting of a diagram/picture discrimination task and a graph comprehension questionnaire.

The *diagram/picture discrimination task* activity is used for assessing graphicity at different cognitive levels and for identifying different types of GAPm. In this task children are asked to discriminate diagrams from pictures across six different representational types (tables, bar charts, line graphs, pie charts, hierarchies/network diagrams, set diagrams). The *graph comprehension questionnaire* (based on [2]) was used to identify the GAPm and to evaluate how efficiently the discrimination task also identifies the misconception.

The *racing car* activity was developed for use by children in (elementary) school years 3, 4, 5 and 6 (ages 8–11 years). It was designed to help students overcome the GAPm. The activity is based on the original paper and pencil "racing car" reported by Janvier [2] and previous research on interactive environments designed to help students understand abstractions (e.g. [5], [3]).

There were two conditions in the study. In one condition, the child "drives" the racing car along a track and a speed/distance graph is plotted concurrently alongside (Condition A). The child was allowed to experiment freely across six different types of tracks; this activity was preceded by a trial session composed of other three tracks. In a vicarious learning condition (Condition B) the child could devote her attentional resources to observing the races and graphs of a peer. Having the two conditions allows comparison of learning outcomes from "observing" versus "doing".

All activities are administered via interactive touch-screen technology. This was used for two purposes. On the one hand, it provided rich data logging of the child's interaction (e.g. latency logging for the discrimination task). On the other hand, it allowed us to take advantage of children's natural movements by providing kinaesthetic experiences - important with young students - in order to reinforce connections between physical actions and graph behaviour.

3 Current and Future Work

A study has already been conducted (N=48) and data analysis is currently being performed. Preliminary results suggests that the diagram/picture discrimination task identifies various degrees and types of GAPm [6]. We also identified several types of representation that students fail to correctly classify. Specifically, it was found that children performed better at classifying common representational types (tables, bar charts, line graphs) compared to less common (e.g. networks, set diagrams).

Although data analysis on the “racing car” activity is still ongoing, an additional improvement to its design is suggested. It seems that some children miss some of the features of the graph, probably because the child’s interaction is focused on the track side (currently, the software allows only to move the car on the track). Therefore, it is suggested to allow the child to manipulate the graph in order to focus her attention on the abstract representation as well. That is, allowing *bi-directionality* on both representations: on the track by moving the car, and on the line graph, by modifying the car’s behaviour. This stage is on current development.

Acknowledgments. Many thanks to my supervisor, Richard Cox, to my sponsor, CONACyT, and to three anonymous reviewers for their comments.

References

1. Leinhardt, G., Zaslavsky, O., Stein, M.K.: Functions, graphs, and graphing: Tasks, learning, and teaching. *Review of Educational Research* 60(1), 1–64 (1990)
2. Janvier, C.: Use of situations in mathematics education. *Educational Studies in Mathematics* 12(1), 113–122 (1981)
3. Vogel, M., Girwidz, R., Engel, J.: Supplantation of Mental Operations on Graphs. *Computers & Education* 49(4), 1287–1298 (2007)
4. Garcia Garcia, G., Cox, R.: Diagrams in the UK National School Curriculum. In: Gem, S., John, H., John, L. (eds.) *Diagrams 2008*. LNCS (LNAI), vol. 5223, pp. 360–363. Springer, Heidelberg (2008)
5. Rogers, Y., Scaife, M., Aldrich, F., Price, S.: Improving Children’s Understanding of Formalisms through Interacting with Multimedia. *Cognitive Science Research Paper* 559 (2003), Retrieved from <http://mcs.open.ac.uk/yr258/papers/csrp559.pdf>
6. Garcia Garcia, G., Cox, R.: A cognitive processing approach to characterizing the ‘graph-as-picture misconceptions’ of primary school students. In: *Diagrams* (2010) (submitted, 2010)

Long Term Student Learner Modeling and Curriculum Mapping

Richard Gluga

University of Sydney, Sydney NSW 2006, Australia

Abstract. Over the years of a university degree, students face many challenges, including: selecting elective subjects, gaining a sense of their own progress, understanding the reasons that they are required to do particular learning tasks, and deciding the time to devote to different assignments. These are important for student engagement and success. We aim to create a new way to help students address these challenges, based on a curriculum mapping system. This will aid students making more informed elective decisions based on personalized progress reports and knowledge gap analysis. Our system will capture whole degree programs, mapping the detailed degree requirements (accreditation, learning objectives, attributes) to the individual subjects, the assessments and each student's actual performance on assessment tasks. We will evaluate this in three stages: qualitative studies of students' interaction with the system based on a fictional student's personalized progress report and knowledge gaps; qualitative studies based on actual profiles; and a field trial which tracks interaction and makes use of questionnaires.

Keywords: Lifelong Learner Modeling, Open Learner Modeling, Curriculum Mapping, Graduate Attributes, Accreditation Competencies, Degree Planning.

1 Introduction and Related Work

University students typically need to deal with a bewildering array of information about their degree requirements, subject choices and their academic progress. There has already been some recognition of the need to support academic staff and curriculum designers in dealing with this complexity as it affects their roles. Mulder [6], [5], discusses the trend and difficulties for design of competency or standards-based development of university curricula in Europe, an example of which is the Tuning project¹.

However, there has been little work that deals with the needs of the *individual* student, who is making their way through a three to five year degree program and is faced with many subject elective decisions along the way. Even within a single subject, students may not be able to see how the individual parts relate to the overall goals of the curriculum. Our work aims to help students cope with this

¹ Tuning - Europe, www.tuning.unideusto.org/tuningeu/

complexity, by creating a long term learner model and associated interfaces that enable a learner to see their own progress across the years of their curriculum.

To establish our precise goals, we first define the term, *attribute*, to be any skill, competency, ability or trait that a student must achieve in order to complete a full degree. Attributes can be described in generic statements such as ‘Level 5 Communication Skills’ or ‘Project Management’, or very specific learning objectives such as ‘Explain the use of big O, omega, and theta notation to describe the amount of work done by an algorithm’. Attributes required for a degree come from diverse sources, including professional bodies like the *Association for Computing Machinery* or accreditation bodies like *Engineering Australia Stage 1 Competency Standard*. These are key drivers for the design of curricula. As we aim to make our work deployable for real use, we need to map such attributes to the learning activities and assessment elements in each subject and then to the individual student’s progress and performance. Essentially, we aim to create Open Learner Models that enable students to answer the following questions: What attributes must I achieve to complete my degree? Which attributes have I achieved so far? Which subjects can I enroll in to achieve required attribute X? Which attribute does Question 2 in Assignment 1 of Subject S support? How am I doing so far in terms of gaining required degree attributes?

An ontological approach to mapping attributes to subjects has been tackled in various forms by Psych et. al. [8], Van Assche [1], and Paquette et. al. in the LORNET TELOS project [7]. One of the acknowledged challenges of this approach is the large setup and maintenance overheads for sophisticated ontologies [7], [4]. These systems do not scale well to large environments and non-expert users. An important foundation for our work is based on lessons learned from Curriculum Central [3] which supports a limited form of curriculum mapping that helps curriculum designers to model links between subjects and a flat set of learning attributes for several Engineering degrees. It does not however support complex degree structures or multiple attribute sets.

The growing recognition that there is a need for such models is reflected in the emerging extensions to Moodle² and Sakai³ which can partially model the links between parts of the subject and overall learning outcomes for that subject. They cannot however show these relations in the context of complex degree programs. Bull & Gardner [2] have taken this further by creating independent OLMs that link subjects to attributes and accreditation requirements across a whole degree. This was restricted to multiple choice questions linked to UK SPEC Standards for Professional Engineering Competence. In our approach we will link arbitrary attributes to any subject assessment type or learning activity, and create learner models to aid students in navigating through degree electives.

2 Approach, Progress and Plans

We have created CUSP, a curriculum management system for 80+ degrees and 800+ subjects spanning 3 Faculties at the University of Sydney. It models the

² Moodle Achieving Competence Today, act.med.virginia.edu/blocks/pla/

³ Sakai Goal Aware Tools, mylsb.keesler.org/goalawaretools.html

mapping between both internal and accreditation attributes to each degree to each subject and each assessment. This provides a foundation for creating our Open Learner Models: the next stages must integrate individual learner's grades on each assessment and explore the creation of effective interfaces for the Open Learner Models.

To evaluate the effectiveness of our Open Learner Models, in meeting our goals, we will conduct a series of three user studies. The first will focus on the usability of the interfaces and understandability of the Open Learner Models. We will define a set of fictional students and create their models. We will recruit students to perform a series of tasks which require them to answer the questions defined above. Each student will repeat the tasks, taking the role of the different fictional students. On the basis of results of this study we will refine the interface and system. The second stage will involve using real student models with real subject enrollments and assessment marks. This will focus on the effectiveness of the system in representing actual data in a meaningful and intuitive way. In our final stage we will perform a field study with students using our system with their own personal data. We will track interactions to assess use of the system. We will use questionnaires to evaluate the user interface effectiveness and capture the student reaction to our tool for degree planning, study and reflection.

References

- [1] Assche, F.V.: Linking learning resources to curricula by using competencies. In: First International Workshop on LO Discovery & Exchange (2007)
- [2] Bull, S., Gardner, P.: Highlighting learning across a degree within an independent open learner model. In: AIED, vol. 200, pp. 275–282 (2009)
- [3] Calvo, R., Carroll, N., Ellis, R.: Curriculum central: A portal system for the academic enterprise. *IJCEELL* 17(1), 43–56 (2007)
- [4] Kalz, M., van Bruggen, J., Rusman, E., Giesbers, B., Koper, R.: Positioning of learners in learning networks with content, metadata and ontologies. *Interactive Learning Environments* (2), 191–200 (2007)
- [5] Mulder, M., Gulikers, J., Wesselink, R., Biemans, H.: The new competence concept in higher education: error or enrichment? *Journal of European Industrial Training* 33, 755–770 (2009)
- [6] Mulder, M., Weigel, T., Collins, K., Bibb, B.: The concept of competence in the development of vocational education and training in selected EU member states: a critical analysis. *Journal of Vocational Education and Training* 59(1), 67–78 (2007)
- [7] Paquette, G., Rosca, I., Mihaila, S., Masmoudi, A.: TELOS, a Service-Oriented framework to support learning and knowledge management. In: *E-Learning Networked Environments and Architectures: A Knowledge Processing Perspective*, p. 434 (2007)
- [8] Psych, V., Bourdeau, J., Nkambou, R., Mizoguchi, R.: Making learning design standards work with an ontology of educational theories. In: Looi, C., McCalla, G.I., Bredeweg, B., Breuker, J. (eds.) *AIED. Frontiers in Artificial Intelligence and Applications*, vol. 125, pp. 539–546. IOS Press, Amsterdam (2005)

Student Dispositions and Help-Seeking in Collaborative Learning

Iris K. Howley and Carolyn Penstein Rosé

Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA, 15213
{ihowley, cp3a}@andrew.cmu.edu

Abstract. Previous work suggests that student dispositional attributes have an effect on help-seeking behaviors. We report on studies attempting to apply this work to a computer-supported collaborative environment that suggest that dispositional attributes of motivation orientation and self-efficacy may be more easily examined in the context of task choices that follow failure events as part of an experimental design. Two studies that have led to this conclusion are discussed, as well as plans for a recovery-from failure study in preparation.

Keywords: Collaborative Learning, Student Dispositions, Motivation.

1 Introduction

Recent work in the student modeling domain has increased in its scope to encompass not only operationalizations of the student's knowledge, but also of the student's attitudes, emotions, and other dispositions. Capturing these dispositions helps us understand not only what topics the student should practice, but what sorts of intelligent help and scaffolding will best benefit the student in his state of mind.

Research has already begun to explore this topic on an individual level. Specifically, Harris et al. (2009) found that type of preferred help differed between performance-oriented students (i.e., ones who tend to do or not do a task in order to build a reputation with others) and mastery-oriented students (i.e., ones who tend to do a task to gain knowledge/skills) [1]. That is, mastery-oriented students were more likely to select help that aided them in their understanding, while performance-oriented students were more likely to select help that gave them the answer to the problem. While this work targets individual learners, it would be worthwhile to determine if the same effects on choice in help-seeking are seen in a collaborative learning settings as well. Collaborative learning introduces additional social and dispositional complications. On the one hand, this makes the issue more difficult to study. Nevertheless, on the other hand, it is more realistic to consider these issues in social contexts since much of learning and work take place in social contexts, and it is also known that collaboration provides many benefits for learning.

This paper describes two example studies performed by the authors to examine how students are influenced by an assortment of different dispositional attributes, in particular: motivation orientation and self-efficacy. If we can determine how these attributes affect help exchange and detect these dispositions automatically, intelligent

support could be added to these computer-supported collaborative learning environments that would improve the overall quality of the interactions and learning.

2 Exploratory Study 1a: Motivation Orientation

We designed an experiment to explore the relationship between sixth-graders' motivation orientation (from Self-Determination Theory, see [2]) and their interactions in a collaborative learning environment. Students worked in 16 pairs in the classroom on tangram problems to learn fractions concepts in the Virtual Math Teams (VMT) environment [3]. Each student sat at her own computer, and partners did not sit next to each other. The methods and results are further discussed in [4].

This study originally operationalized motivation orientations with categories from self-determination theory including amotivated, introjected, identified, and internal regulation, which we eventually collapsed into two categories: extrinsic motivation (comprised of amotivated and introjected) and intrinsic motivation (comprised of identified and internal). Overall, what we see are mostly marginally significant differences and trends that a student's own motivation orientation may color their perception of the exchange of help in the collaboration, sometimes obscuring the reality of the help actually exchanged [4]. Our results examining student perceptions of the collaboration showed that a student's own motivation orientation is correlated with particular perceptions of their partner's behavior, although the student's perception of the partner is not statistically related to the partner's orientation. As this was a correlational study, we cannot make any causal inferences.

3 Exploratory Study 1b: Self-efficacy and Help Buttons

Using the same group of students as Exploratory Study 1a, we used a shared interface for fractions problem-solving to investigate the connection between student dispositions and help-button choice. Students were given a pretest, questionnaire for self-efficacy [5] and for perceptions of the collaboration, and a posttest. As in Study 1a, all group communication was computer mediated. While students solved problems in pairs they had the option of 4 different help buttons, for which they were given 5 minutes of training. The "Help me Solve" button introduced a context-sensitive hint into the chat history, "Help from Partner" sent the student's partner a message requesting help, "Help me Explain" offered self-reflection prompts to help explain a step to his partner, and "Help me Understand" also gave self-reflection prompts.

This exploratory design was intended to connect the motivation orientation data from the previous study with the additional construct of self-efficacy [5], and discover how these constructs relate to choices students make about help exchange. From previous work, it seems plausible that students with varying dispositions would prefer to engage in different help exchange behavior. Results showed a correlation between gender and button usage (with females using the help buttons more), but we were not able to find any statistically significant patterns in the data related to motivation orientation, self-efficacy, and help button usage, possibly due to low statistical power.

3 Conclusions

The two studies described in this paper are accompanied by two more studies done with older engineering students, resulting in similar unclear results about achievement goal orientation and helping-behaviors [6]. While it is possible that motivation orientation, achievement goals, and self-efficacy have no influence on helping behaviors in these domains, or age groups, or in collaborations, it seems more plausible that the relationship could be better examined in a more carefully constructed environment. We need to make the features of these dispositional attributes more salient through the addition of a task choice behavioral measure. Past studies in the achievement goal orientation literature have successfully used task choice as a behavioral measure of achievement goal orientation in conjunction with self-report data [7]. The authors are in the process of developing an experimental design that looks more carefully at how participating in a collaborative versus individual task affects task choice and student dispositions. Students assigned to a collaborative task might show more performance-avoidance behaviors than students in individual tasks, avoiding failure in front of others. A construct pertaining to social roles such as social presentation/perception will help capture the social aspects of collaboration. With a better understanding of how collaboration affects student motivations and vice versa, we will move closer to developing intelligent tutors that consider not only student knowledge, but also student feelings in a social setting.

Acknowledgments. This work was supported in part by Graduate Training Grant awarded to Carnegie Mellon by the Department of Education (# R305B040063).

References

1. Harris, A., Bonnett, V., Luckin, R., Yuill, N., Avramides, K.: Scaffolding Effective Help-seeking Behaviour in Mastery and Performance Oriented Learners. In: Proceedings of Artificial Intelligence in Education (AIED), Brighton, UK (2009)
2. Deci, E.L., Vallerand, R.J., Pelletier, L.G., Ryan, R.M.: Motivation and Education: The Self-determination Perspective. *Educational Psychologist* 26(3-4), 325–346 (1991)
3. Stahl, G.: Analyzing and Designing the Group Cognitive Experience. In: IJCIS (2006)
4. Howley, I., Chaudhuri, S., Kumar, R., Rosé, C.: Motivation and Collaborative Behavior: An Exploratory Analysis. In: Poster presented at Computer Supported Collaborative Learning (CSCL 2009), Rhodes, Greece (2009)
5. Bandura, A.: Self-efficacy: Toward a Unifying Theory of Behavioral Change. *Psychological Review* 84, 191–215 (1977)
6. Howley, I., Chaudhuri, S., Kumar, R., Rosé, C.: Motivation and Collaboration On-Line. Poster presented at Artificial Intelligence in Education (AIED), Brighton, UK (2009)
7. Ames, C., Archer, J.: Achievement Goals in the Classroom: Students' Learning Strategies and Motivation Processes. *Journal of Educational Psychology* 80(3), 260–267 (1988)

Visualizing Educational Data from Logic Tutors

Matthew Johnson and Tiffany Barnes

University of North Carolina at Charlotte, Charlotte NC, 28213, USA
mjokimoto@gmail.com, Tiffany.Barnes@unc Charlotte.com

Abstract. We propose a data visualization tool that offers insights into the way students solve procedural domain problems. The tool uses nodes and edges to represent states and actions which students have generated using an intelligent tutoring system or computer aided instruction tool, ultimately showing the way a student has solved a problem. We use the example of logic tutor data and suggest two methods of evaluation for ensuring the tool is effective at aiding educators to better understand student learning.

Keywords: Educational Data Visualization.

1 Introduction

We propose to build and evaluate EDM Vis, a software tool for visualizing student problem-solving in learning environments such as computer-aided instruction (CAI) or intelligent tutoring systems (ITS). EDM Vis can be used by educational researchers, instructional designers, and educators to better understand student learning, and when integrated with a CAI or ITS, can augment and edit hints as well as feedback provided to students at specific steps in their problem solving processes.

Educators are often responsible for the success of their students and are thus reluctant to relinquish control of their classrooms, but this is just what most ITS and CAI tools expect [1]. Creating educator-oriented software can allow teachers to continue to have control over their classroom and increases the likelihood of adoption of new learning media. Lessons created and administered via a software medium could allow teachers to have more classroom control as well as offer new insights regarding their students' deficiencies, strengths, progress and ways of thinking.

The EDM Vis is a branch off of previous works [2,3] in which they used historical student data to generate hints for a logic-proof tutoring system. With the goal of combining the two pieces allowing educators to navigate, explore and alter the feedback contained in the ITS and CAI tools based on the insights they were able to gain via the use of EDM Vis.

2 The Visualization Tool

EDM Vis can help educators understand student learning because it displays how students have progressed through procedural problems in either an ITS

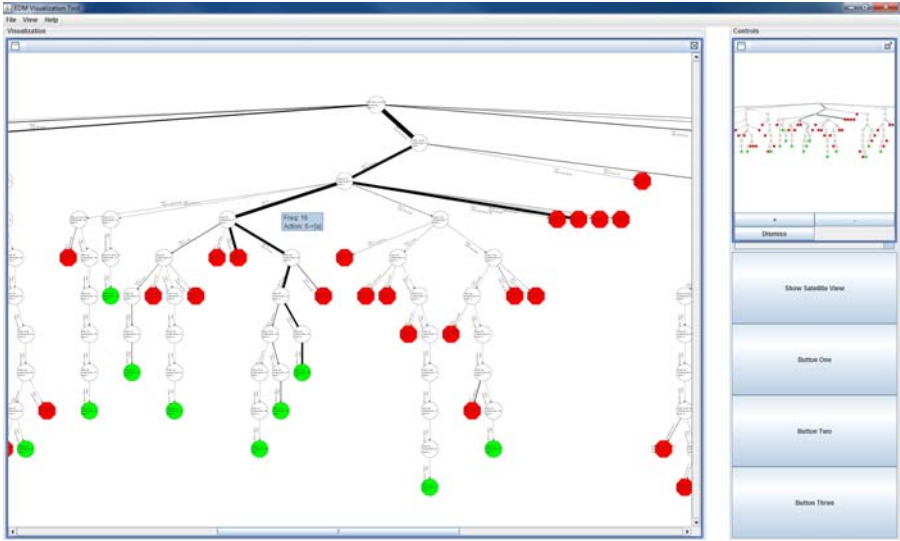


Fig. 1. An example screen shot of the proposed EDM Vis tool

or CAI tool. Problems are discretized into states, with the initial parameters described by the problem as the starting state. Consecutive states are obtained based on the actions students have performed in an ITS or CAI tool. Our current implementation makes use of logic tutor data from Deep Thought [4], and in this case a student's action is the application of a logic rule and the state consists of the resulting parameters.

The visualization shows each state as a node and each action as an edge between the nodes, similar to a tree structure, see figure 1. Error states are also shown but such states do not have children nodes since errors are defined as states which do not lead to the goal. Nodes and edges are generated from strings meaning many domains can be visualized, focused mainly on step-by-step processes with defined goal-states.

A variety of different navigation abilities exist for exploring the data and allow the user to focus on the most important aspects of the information. These abilities are the seven tasks detailed in [5], and include: Overview, zoom, filter, detail-on-demand, relate, history and extract. Through these tasks educators can quickly identify error nodes, high frequency nodes or edges, show a list of students which made use of a specific edge(action) or visited a particular state and more. Though this list is not exhaustive, these functions give an overview of the types of interaction and data exploration we are considering. EDM Vis enables educators to explore their students' data and identify areas that are most troubling for students, and will eventually allow them to annotate learning paths with scaffolding and hints that can be incorporated into the CAI or ITS.

3 Evaluation

We plan to evaluate the EDM Vis Tool in three ways. First, we hypothesize that our tool can be effectively used and understood by educators for exploration of learning data from a CAI or ITS. To test this hypothesis, we will conduct a usability study that asks educators to perform simple navigation and exploration of the data provided, and ask users to explain what they are seeing. We will also ask users for feedback on the interactive elements of the tool, and what suggestions they have for more tools to support their exploration of the data.

Our second hypothesis is that educators will be able to effectively use EDM Vis to better understand student learning. To test this hypothesis, we will collect data from Deep Thought in several logic classes, and present the data collected in the EDM Vis tool to educators, asking them to explore the visualization and report what trends they discover about student learning, focusing on whether their expectations are met and whether they were able to discover new and surprising results. In the case of the logic tutor, an example of such a discovery would be whether or not students ever solved a problem using one of the more complex logic rules.

Our third hypothesis is that educators will be able to use EDM Vis to quickly answer specific questions about student learning. To test this hypothesis, we will ask educators several specific questions about the learning data and determine whether they can answer it by using the tool. Again using our logic example, if we asked the user to determine the number of students who had difficulty with a particular logic rule, we would monitor how quickly the answer was found, with what level of difficulty, and how accurately.

References

1. Bondaryk, L.: Publishing new media in higher education: Overcoming the adoption hurdle. *Journal of Interactive Media in Education* 3(98)
2. Stamper, J., Barnes, T., Lehmann, L., Croy, M.: The hint factory: Automatic generation of contextualized help for existing computer aided instruction. In: *Proceedings of the 9th International Conference on Intelligent Tutoring Systems Young Researchers Track*, pp. 71–78 (2008)
3. Barnes, T., Stamper, J.: Toward automatic hint generation for logic proof tutoring using historical student data. In: *Wolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 373–382. Springer, Heidelberg (2008)*
4. Croy, M.J.: Graphic interface design and deductive proof construction. *J. Comput. Math. Sci. Teach.* 18(4), 371–385 (1999)
5. Shneiderman, B.: The eyes have it: A task by data type taxonomy for information visualizations. In: *IEEE Symposium on Visual Languages*, vol. 336 (1996)

An Authoring Language as a Key to Usability in a Problem-Solving ITS Framework

Jean-François Lebeau, Luc Paquette, Mikaël Fortin, and André Mayers

Université de Sherbrooke, Québec, Canada

{jean-francois.lebeau2, andre.mayers}@usherbrooke.ca

Abstract. Step-based ITS have been proven successful in well-defined domains, but their success is mitigated by their cost. Different approaches have been investigated to reduce these efforts; one of them is a framework that eases the development of tutors for a given class of task domains. In this paper, we explain how a domain is modeled with the ASTUS framework and we discuss why an authoring language is a promising technique to improve its usability.

Keywords: Knowledge representation, Authoring system, Problem-solving ITS.

1 Introduction

Step-based ITS have been proven successful for well-defined domains [5], particularly in well-defined tasks [8], but their success is mitigated by their cost. Typically, this cost is explained by efforts needed to model the task domain. Different approaches have been investigated to reduce these efforts. One is a framework that eases the development of tutors for a given class of domains. Model-tracing tutors (MTT) such as the Cognitive Tutors [3] (CMU's TDK, CTAT's Jess¹-based tutors (JBCT) [1], Carnegie Learning's SDK, etc.) and Andes [11] follow this approach. Another one is to forgo generative models which allow the system to solve the tasks, to rely on evaluative models. For example, in ASPIRE a model consists of constraints defined over a set of pedagogically relevant solutions [7]. A third one is to move the bulk of the efforts from programmers to domain experts. Indeed, an authoring system can infer a more-or-less task-dedicated evaluative model from step-by-step solutions performed in the learning environment (LE). CTAT's example-tracing tutors [2] and ASSISTment are examples of such systems [10]. With ASTUS, we aim to offer to the ITS community a framework for problem-solving tasks in well-defined domains. In such context, relying on a generative model of the task domain was deemed the most interesting choice, as it appeared as the one leading to a comprehensive and flexible solution, which includes not only the capacity to offer next-step hints, but to generate them by instantiating domain-independent templates with data extracted from knowledge components. In summary, ASTUS acts as a MTT by using examinable models that facilitates defining domain-independent pedagogical strategies [6, 9].

¹ <http://www.jessrules.com>

In this paper, we explain how a task domain is modeled within the ASTUS framework and we discuss why an authoring language (prototyped with a Groovy²-based DSL) is a promising technique to improve the usability of such a framework.

2 How and Why a Task Is Modeled Using an Authoring Language

Under ASTUS's knowledge representation approach, a task domain's declarative knowledge is divided into semantic (factual) and episodic (autobiographical) components whereas procedural knowledge is modeled at three different grain sizes. First, *complex procedures* (CPs) are dynamic plans generating a set of goals, (intentions satisfied by procedures), according to an algorithm (sequence, condition, iteration, etc.). Second, *primitive procedures* (PPs) represent mastered abilities that correspond to steps in the LE. Third, *queries and inference rules* (Q&IR) encode elementary or mastered mental abilities (with CPs, they correspond to inferences). A goal is specified for each task and some CPs can be flagged as incorrect to represent known errors. Aside goals, semantic components include: *concepts, relations, functions, contexts* (reifying LEs' subdivisions) and their corresponding instances: *objects, facts, mappings and environments* respectively. Queries specify how semantic components are fetched from an environment; their result appears as arguments of instantiated goals and procedures. IRs are grafted to semantic components to make them operational (via Jess). When a PP is committed, i.e. when the interactions in the LE match a specific sequence of atomic UI actions [4] (text input, button press, etc.), a script is executed to update the environment accordingly. For a given task, the instantiated goals and procedures form an episodic graph which acts as a high-level log, whereas the low-level log contains the actual steps executed by the learner.

The idea behind this representation is to encode tutored skills (CPs) with glass-box components and the underlying ones (Q&IR and PPs) with black-box ones. Thus a model consists of formatted definitions (e.g. CPs, queries and semantic components) and executable code (e.g. PPs and IRs). We already faced a similar situation when we developed a MVC-based method [4] to handle the LE interactions. We used a hybrid approach combining an authoring language used to define templates that recognize the UI actions associated to a PP and Java code that extract semantic components from the learner's raw input or selections and that produce UI elements that represent the current environment. Satisfied with the usability of this method, we decided to use a similar one for the knowledge components. In this case, we assumed that a first critical step for usability would be a model of a task domain completely encoded in a single, coherent, easy-to-navigate file, much like a typical source file should be. To achieve this, we used Groovy, which of course makes it easier to parse our authoring language, but more importantly offer support for closures that make possible to interleave both definitions and code. Also, debugging tools equivalent to those offered by JBCTs [1] are available, and as for IDE-based tools (text edition, syntax checks, etc.), we rely on *Eclipse*. Furthermore, GUI-based tools such as "wizards" and LE visual editors (e.g. as in CTAT) may be included in future versions of ASTUS.

² <http://groovy.codehaus.org/>

3 Conclusion

As the ITSs move from the labs to the classrooms, the next logical step may be to move the authoring efforts from specialized graduate students to domain experts (including teachers), but we are interested in an intermediate step that consist in a comprehensive, flexible and usable framework for people with programming and knowledge-based systems skills. We are aware that our solution, based on generative models, may be justified only in well-defined domains and that some ill-defined tasks, such as design-based ones, may be challenging at best. However, there is no such tool available for the ITS community that is explicitly designed to facilitate the experimentation of different pedagogical approaches. Finally, the next step for us will be to evaluate the benefits of ASTUS as an authoring framework. To do so we will be looking to collaborate with teams that need to create tutors as part of their pedagogical researches and that can objectively assess how ASTUS has helped them.

References

1. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.: The Cognitive Tutor Authoring Tools (CTAT): Preliminary evaluation of efficiency gains. In: *Proceedings of the 8th International Conference on Intelligent Tutoring System*, pp. 61–70 (2006)
2. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: Example-Tracing Tutors: A New Paradigm for Intelligent Tutoring Systems. *IJAIED, Special Issue on Authoring Systems for Intelligent Tutoring Systems*, 105–154 (2009)
3. Anderson, J.R., Corbett, A.T., Koedinger, K.R., Pelletier, R.: Cognitive tutors: Lessons learned. *Journal of Learning Science* 4(2), 167–207 (1995)
4. Fortin, M., Lebeau, J.F., Abdessemed, A., Courtemanche, F., Mayers, A.: A Standard Method of Developing User Interfaces for a Generic ITS. In: *Proc. of the 9th International Conference on Intelligent Tutoring System*, pp. 312–322 (2008)
5. Koedinger, K.R., Anderson, J.R.: Intelligent Tutoring Goes To School in the Big City. *International Journal of Artificial Intelligence in Education* 8, 30–43 (1997)
6. Lebeau, J.-F., Fortin, M., Paquette, L., Mayers, A.: From Cognitive to Pedagogical Knowledge Models in Problem-Solving ITS Frameworks. In: *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, pp. 731–733 (2009)
7. Mitrovic, A., Suraweera, P., Martin, B., Zakharov, K., Milik, N., Holland, J.: Authoring constraint-based tutors in ASPIRE. In: Ikeda, M., Ashley, K., Chan, T.-W. (eds.) *Proc. of the 8th International Conference on Intelligent Tutoring Systems*, pp. 41–50 (2006)
8. Mitrovic, A., Weerasinghe, A.: Revisiting Ill-Definedness and Consequences for ITSs. In: *Proc. of the 14th Artificial Intelligence in Education*, pp. 375–382 (2009)
9. Paquette, L., Lebeau, J.F., Mayers, A.: Integrating Sophisticated Domain-Independent Pedagogical Behaviors to an ITS Framework (2010)
10. Razaq, L., Patvarczki, J., Almeida, S.F., Vartak, M., Feng, M., Heffernan, N.T., Koedinger, K.: The ASSISTment builder: Supporting the Life-cycle of ITS Content Creation. *IEEE Transactions on Learning Technologies, Special Issue on Real-World Applications of Intelligent Tutoring Systems* 2(2), 157–166 (2009)
11. VanLehn, K.: The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education* 16(3), 227–265 (2006)
12. VanLehn, K., et al.: The Andes physics tutoring system: Lessons Learned. *International Journal of Artificial Intelligence and Education* 15(3), 1–47 (2005)

Towards the Creation of a Data-Driven Programming Tutor

Behrooz Mostafavi and Tiffany Barnes

College of Computing and Informatics, UNC Charlotte

Abstract. Educational data mining methods are being used to automatically generate hints to students in intelligent tutoring systems. Using these methods, we hope to create a system that can give individualized instruction. By analyzing time snapshot data from exams in an introductory programming course, we will write a program to construct state graphs for each student's performance, eventually resulting in a Markov decision process that represents different approaches to writing the target program, and providing feedback to students. Once this system is sufficiently tested and refined, it will then be applied to subsequent semesters students in the programming course.

Keywords: Programming Tutor, Markov decision process.

1 Introduction and Background

Intelligent tutoring systems are tools that are built to help students learn and adapt to an individual student. However, a single hour of instruction may require anywhere from 100 to 300 hours of development, and are usually focused on a single topic.

Educational data mining methods have recently been used to automate the construction of domain models to automatically generate hints in a domain-independent way, paving the way to building data-driven ways to provide the benefits of intelligent tutoring systems [Barnes and Stamper, 2009]. There have also been other instances of tools created for the purpose of cognitive tutoring [Aleven and McLaren, et al, 2006], and natural language tutoring [Lane and VanLehn, 2005], which form a basis of knowledge for this work.

2 Current Work

By using the outline of an intelligent tutoring system, and applying data mining methods to speed the process of development, we hope to create a system that can judge a student's performance by the way he or she responds to questions and performs on homework and exam material, determine where the student needs help, select new material for the student, and give individualized instruction in a form and pace that the student can benefit from.

We have started this research by recording exam data from an introductory programming course teaching C++. Each student was required to write a program individually during a 75 minute class, and a snapshot of each student's progress was made

every five minutes. There are data for over 200 students, for two separate programming exams. Each exam had an area of focus; that is, there were specific programming concepts that required demonstration within the exam to receive full credit. The first exam was a guessing game the students wrote, in order to demonstrate knowledge of random number programming and the modulus operator, as well as conditional loops. The second exam focused on writing and calling in-program functions, as well as storing data in arrays. In addition, each exam was built on the knowledge that preceded it. Students therefore also needed to have a grasp of prior concepts taught in order to perform well on the exam.

The student data is currently being organized in the following categories as independent variables: course instructor, lab instructor, exam grade, final course grade, and demographics including the student's class standing and concentration of study, and gender. Within these categories, each student's time snapshots are analyzed, with focus being on how much conceptual progress each student made within each time interval, and where the student may have needed help, either with the conceptual material, or with programming concepts in general. Any trends that appear in analysis will be compared with the previously mentioned categories in order to determine any existing correlation between the independent variables and the student's results.

Analysis is being conducted by the first author, a lab instructor for the introductory programming course. Each program snapshot is analyzed for whether goals of the test are achieved, and how the current snapshot differs from the previous one. The goal of the analysis is to create a model that denotes what problem-solving or programming steps the student has performed in his program during each 5-minute time interval. This analysis will be used to create a "state" which illustrates what steps the student has taken until the current time. Once the analysis is performed, we can illustrate one student's progress as a series of states, with actions taken to move the student from one state to the next. For example, we have analyzed one student's program to consist of the following states: S1, Action1, S2 Action2, S3, We are currently performing this analysis by hand. However, once salient features of states are determined, we will write a program to label each snapshot with state information and construct graphs for each student's performance.

Once these performance graphs are constructed, we will combine them into a single large graph that shows how all students have progressed in writing the target program. If we are successful in this process, we will then assign a large positive value to achieving the test goals and perform value iteration to assign values to every state in the solutions graph, resulting in a Markov decision process that represents the different approaches to writing the target program, with states closer to the solution having higher values and those further away with lower values. Once this is constructed, we can use it to visualize student progress through the programming task to better understand what strategies students are using to solve problems. We also hope to be able to construct ways to use this MDP to automatically provide intelligent hints and feedback to students in the course of writing a program.

3 Future Work

With the analyses, we plan to develop a learning model that can be used to predict a student's performance in the course based on the trends that are found, and applied to

subsequent semesters of students. This model will require extensive testing and revision over the course of several semesters in order to determine its viability. If the model is found viable, it can be used to create models for student performance in programming and automated support and feedback.

References

1. Aleven, V., McLaren, B., Sewall, J., Koedinger, K.R.: The Cognitive Tutor Authoring Tools (CTAT): Preliminary Evaluation of Efficiency Gains. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 61–70. Springer, Heidelberg (2006)
2. Aleven, V., McLaren, B., Roll, I., Koedinger, K.R.: Toward Meta-cognitive Tutoring: A Model of Help Seeking with a Cognitive Tutor. *International Journal of Artificial Intelligence in Education* 16, 101–128 (2006)
3. Barnes, T., Stamper, J.: Automatic hint generation for logic proof tutoring using historical data. *Educational Technology & Society, Special Issue on Intelligent Tutoring Systems* (2009)
4. Jadud, M.C.: Methods and Tools for Exploring Novice Compilation Behaviour. In: *Proceedings of the Second International Workshop on Computing Education Research*, pp. 73–84. ACM, New York (2006)
5. Lane, H.C., VanLehn, K.: Teaching the Tacit Knowledge of Programming to Novices with Natural Language Tutoring. *Computer Science Education* (2005)

Using Expert Models to Provide Feedback on Clinical Reasoning Skills

Laura Naismith and Susanne P. Lajoie

ATLAS Laboratory, Department of Educational and Counselling Psychology
McGill University, 3700 McTavish St, Montreal, QC H3A 1Y2, Canada
{laura.naismith, susanne.lajoie}@mcgill.ca

Abstract. Effective feedback is necessary to support expertise development in clinical reasoning. Technology-rich environments (TREs) often use expert models as one means of providing this feedback. A review of empirical studies showed 3 different types of expert models in TREs: outcome models, process models, and dynamic models. This paper presents examples of each of these models and discusses their implications for the future design of feedback mechanisms to support clinical reasoning development through self-assessment.

Keywords: expert model, feedback, clinical reasoning, medical education, self-assessment.

1 Introduction

Clinical reasoning is a process that physicians and health professionals use to make decisions about the diagnosis and treatment of their patients. In order to develop the multidimensional knowledge base that underlies clinical reasoning expertise [1], novice clinicians must engage in sustained deliberate practice [2]. Deliberate practice requires an experienced teacher or coach to assign structured activities that gradually increase in complexity, and to provide feedback on performance [3].

Technology-rich environments (TREs) [4] can supplement increasingly limited clinical opportunities [5] by providing additional opportunities for such practice. Based primarily on a cognitivist model of learning, TREs support the development of clinical reasoning expertise by simplifying and constraining the problem space to a finite set of clinical variables, identifying and correcting errors, and making “expert models of performance and competency more visible to learners in the context of the problem solving” [4:805].

This paper examines how different TREs have used expert models as a feedback mechanism to support the development of clinical reasoning skills.

2 Using Expert Models to Provide Feedback

Based on a review of empirical studies, 3 types of expert models were found to promote clinical reasoning skill development in TREs: outcome models, process models, and dynamic models.

Outcome models refer to the display of the expert's final answer to a clinical reasoning task, which is typically expressed as a diagnosis. Outcome models may also include information about the relationship between different diagnostic cues and the final diagnosis. MEDICL [6] is an example of a TRE that uses an expert outcome model to provide feedback on diagnoses related to acute abdominal pain.

Process models include information about the steps that the expert took during the problem-solving process and may include the consideration of information that is not associated with the final diagnosis. The Diagnostic Pathfinder (dP) [7] is case-based TRE for veterinary students. Students have to identify abnormal patient history and laboratory results and construct a diagnostic path that relates these abnormal findings to causal mechanisms (disease processes). After submitting a diagnostic path, students receive feedback in the form of a static process model of an expert's diagnostic path. Process models may also utilize information about expert outcomes. In BioWorld [8], a case-based TRE for medical students, the student first reviews a patient summary, then formulates a hypothesis about the patient's disease and gathers supporting evidence through simulated medical tests. After submitting a final diagnosis, confidence level, prioritized evidence list and case summary, the student can then view and compare his or her solution to a prioritized list of evidence that an expert used to solve the case (outcome model component), as well as a text-based expert summary (process model component). A further study [9] looked at the effects of multimedia visual representations of expert's actions while solving the same case in BioWorld. These representations are interactive, in that students can click on various parts of the diagram to hear excerpts of an audio transcript of the expert's verbal think-aloud when solving the same case.

Finally, *dynamic models* use process information to provide hints and feedback in the process of problem-solving. SlideTutor [10] is an ITS that supports the development of visual classification problem-solving skills. Each time a student initiates a case, a dynamic solution graph (DSG) is generated. The DSG represents the current problem state and valid next steps, as determined by a sophisticated cognitive architecture and expert model. Error states, in which the DSG does not match completely with one of the 10 different types of nodes, trigger feedback responses in the form of text-based alerts. The system may also deliver additional recommendations or feedback after correct actions, if the DSG is determined to reflect an inconsistency.

3 Discussion and Conclusions

TREs can provide feedback to medical students by allowing them to compare their solutions to an expert model. These models can vary from displaying expert outcomes [6] to static process models upon completion of a case [7][8] to highly dynamic comparisons that are performed during the process of problem-solving [10]. Use of all types of models have been associated with learning gains, though inconsistencies in performance measures prevent direct comparisons.

While students appear to appreciate the opportunity to review expert solutions [7], it is not known how students actually interpret and utilize them to improve their own learning and performance. Follow-up on this issue is needed given that numerous studies have demonstrated that students are poor at self-assessment [6][10]. Possible

strategies to support medical students in self-assessing and comparing their performance with experts include: a) providing opportunities for students to review a variety of performance models at different expertise levels, b) providing indicators of the students' progress towards explicit goals and sub-goals and c) providing instructor tools to support the aggregation of multiple student solutions to support classroom review sessions. Further research is necessary to demonstrate the utility and effectiveness of these strategies for the design of feedback within technology-rich environments that promote clinical reasoning expertise.

References

1. Norman, G.A.: Research in clinical reasoning: past history and current trends. *Med. Educ.* 39, 418–427 (2005)
2. Ericsson, K.A.: Deliberate practice and the acquisition and maintenance of expert performance in medicine and related domains. *Acad. Med.* 79(10), S70–S81 (2004)
3. Ericsson, K.A., Krampe, R.T., Tesch-Römer, C.: The role of deliberate practice in the acquisition of expert performance. *Psych. Rev.* 100(3), 363–406 (1993)
4. Lajoie, S.P., Azevedo, R.: Teaching and learning in technology-rich environments. In: Alexander, P.A., Winne, P.H. (eds.) *Handbook of educational psychology*, 2nd edn., pp. 803–821. Erlbaum, Mahwah (2006)
5. Gordon, J., Hazlett, C., ten Cate, O., Mann, K., Kilminster, S., Prince, K., et al.: Strategic planning in medical education: enhancing the learning environment for students in clinical settings. *Med. Educ.* 34(10), 841–850 (2000)
6. Schwartz, S., Griffin, T.: Comparing different types of performance feedback and computer-based instruction in teaching medical students how to diagnose acute abdominal pain. *Acad. Med.* 68(11), 862–864 (1993)
7. Danielson, J.A., Mills, E.M., Vermeer, P.J., Preast, V.A., Young, K.M., Christopher, M.M., et al.: Characteristics of a cognitive tool that helps students learn diagnostic problem solving. *Educational Technology, Research and Development* 55(5), 499–520 (2007)
8. Lajoie, S.P.: Developing professional expertise with a cognitive apprenticeship model: Examples from avionics and medicine. In: Ericsson, K.A. (ed.) *Development of professional expertise: Toward measurement of expert performance and design of optimal learning environments*, pp. 61–83. Cambridge University Press, Cambridge (2009)
9. Gauthier, G., Naismith, L., Lajoie, S.P., Wiseman, J.: Using expert decision maps to promote reflection and self-assessment in medical case-based instruction. In: Aleven, V., Ashley, K., Lynch, C., Pinkwart, N. (eds.) *Intelligent tutoring systems for ill-defined domains*, Workshop conducted at the 9th International Conference on Intelligent Tutoring Systems, Montreal, June 2008, pp. 68–80 (2008)
10. Crowley, R.S., Medvedeva, O.: An intelligent tutoring system for visual classification problem solving. *Artifl Intell. in Med.* 36(1), 85–117 (2006)

Algorithms for Robust Knowledge Extraction in Learning Environments

Ifeyinwa Okoye, Keith Maull, and Tamara Sumner

Institute of Cognitive Science, University of Colorado Boulder, Colorado 80309 USA
{okoye,maull,sumner}@colorado.edu

Abstract. This paper presents preliminary results on a generalizability study that was carried out to evaluate the robustness of a knowledge extraction algorithm.

Keywords: multi-document summarization, automatic knowledge base creation.

1 Introduction

Assessing learners understanding and supporting learners to improve their understanding are some of the goals of intelligent tutoring systems (ITS). This paper describes how knowledge extraction algorithms can automate building a knowledge base for an ITS and presents preliminary results on a generalizability study that was carried out to evaluate the robustness of a knowledge extraction algorithm.

2 Knowledge Extraction

Developing knowledge bases for structured educational knowledge most often used in ITS systems, is expensive, time consuming and frequently requires highly trained experts to help develop rules that encode the most important and relevant sentences within a domain [3]. To advance the efficiency of knowledge base creation and hence ITS systems, automated knowledge discovery algorithms are being developed to extract substantive conceptual information from domain-specific resources.

Our knowledge extraction algorithm called COGENT is based on MEAD [4], a document summarization tool. COGENT uses MEAD with its default configuration and adds other features such as educational standards, hypertext and content word density to determine which sentences to extract from digital library resources for use in building a knowledge base. The educational standards feature assigns a score to a sentence based on its similarity to the text contents of the American Association for the Advancement of Science (AAAS) learning goals on plate tectonics. The hypertext feature assigns a higher score to sentences contained under higher level HTML headings while the content word density feature retains sentences that exceed the 50% cutoff ratio of content words to function words. The final score for each sentence is computed by adding up the scores from all the features. COGENT selects the top scoring sentences until the selected text reaches 5% of the total input text - a value determined from empirical studies of experts performing the same task. COGENT eliminates redundancy along the way based on a cutoff cosine similarity of 0.7 [1].

COGENT was embedded within a personalized learning environment - the customized learning service for concept knowledge (CLICK) as discussed in [1]. Results to date indicate that CLICK and hence COGENT works well in the earthquake and plate tectonics (EPT) domain [2]. While these initial results are encouraging, the next step is to test the robustness of COGENT. Key to testing this robustness is evaluating performance at domain independence - that is how well COGENT generalizes across other topics and domains.

3 Generalization Study

We chose weather and climate (WCL), a near domain to earthquake and plate tectonics (i.e. a topic within earth science) as our first test bed to evaluate the generalizability of COGENT. The design and result of the study are as follows; an expert in EPT selected twenty resources that supported national learning goals in EPT for high school students from the Digital Library for Earth System Education. Another expert in the WCL domain did the same for WCL. Then, using COGENT, sentences were extracted from these two groups of resources. 326 sentences were extracted from the EPT resources and 272 from the WCL resources. Next, two experts in each domain were asked to score each extracted sentence twice; first as a standalone sentence and second, in context with related sentences because it is possible for a sentence to not be useful by itself but to be useful when presented with related sentences. The scores were based on a 5-point Likert scale from strongly agree to strongly disagree. During analysis, we collapsed responses of *strongly agree* and *agree* to mean the sentence was *relevant*, while *strongly disagree* and *disagree* meant the sentence was *irrelevant*. Because the data was ordered, weighted kappa was computed to evaluate the inter-rater reliability. The

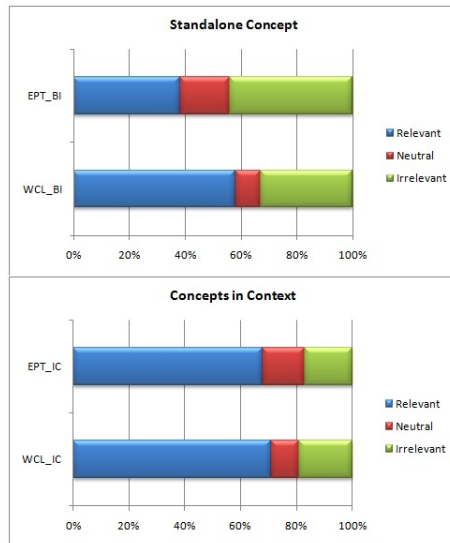


Fig. 1. Comparison of the knowledge extraction algorithm on EPT and WCL

Table 1. Sample errors in extracted sentences

Type of Error	Example
Reference to objects not extracted as part of the sentence	Click here for a list of references about plate tectonics
News articles	In the aftermath of the Oct. 8 earthquake, residents wander in disbelief through a city in Pakistan
Questions	What blocks the sun's energy from escaping from the Earth?
Unresolved co-reference	The products of this reaction are water and carbon dioxide
Synopsis and topic listing	The lesson concludes with descriptions of the location and types of plate boundaries

inter-rater agreement for the EPT sentences as 'standalone' was 0.379, and 0.439 for the WCL data. For 'sentences in context', the agreement was 0.335 for EPT and 0.404 for WCL. The low inter-rater agreement shows how difficult it is to objectively select sentences from digital library resources to build a domain knowledge base.

Figure 1 shows that COGENT worked in the WCL domain. Moreover, the algorithm extracted slightly more relevant stand-alone sentences in the WCL domain than in the EPT domain - the domain for which the algorithm was developed. Preliminary error analysis of the irrelevant sentences provides a sampling of the kinds of mistakes the algorithm produces as shown in Table 1. Reducing the number of irrelevant sentences extracted is a tractable problem that can be solved by fine-tuning COGENT to check for unresolved co-references and questions.

Deeper analysis of the irrelevant and neutral sentences is necessary to fully understand how to improve COGENT. However, the encouraging results already seen in our near domain experiment weakly imply the feasibility of COGENT performing at similar levels in far domains such as Biology. A full round of experiments is underway to validate that with updated parameters, the algorithm will generalize to a far domain like Biology.

References

1. de la Chica, S., Ahmad, F., Martin, J.H., Sumner, T.: Pedagogically useful extractive summaries for science education. In: Proceedings of the 22nd International Conference on Computational Linguistics, vol. 1, pp. 177–184. Association for Computational Linguistics (2008)
2. Gu, Q., Chica, S., Ahmad, F., Khan, H., Sumner, T., Martin, J.H., Butcher, K.: Personalizing the Selection of Digital Library Resources to Support Intentional Learning. In: Christensen-Dalsgaard, B., Castelli, D., Ammitzbøll Jurik, B., Lippincott, J. (eds.) ECDL 2008. LNCS, vol. 5173, p. 255. Springer, Heidelberg (2008)
3. Murray, T., Blessing, S., Ainsworth, S.: Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software. Kluwer Academic Pub., Dordrecht (2003)
4. Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., Zhang, Z.: MEAD - a platform for multidocument multilingual text summarization. In: LREC 2004, Lisbon, Portugal (May 2004)

Integrating Sophisticated Domain-Independent Pedagogical Behaviors in an ITS Framework

Luc Paquette, Jean-François Lebeau, and André Mayers

Université de Sherbrooke, Québec, Canada
{Luc.Paquette, Andre.Mayers}@USherbrooke.ca

Abstract. ITS authoring frameworks are useful to reduce the efforts needed to create tutors, but the resulting tutors are usually more limited than domain-specific ones. The sophistication of the pedagogical behaviors they can produce depends on the knowledge components available to model a task domain. In this paper, we present research focused on integrating sophisticated domain-independent pedagogical behaviors in an ITS framework.

Keywords: Problem-solving ITS, Pedagogical behavior, Well-defined domains.

1 Introduction

Different ITS authoring frameworks allow the generalization of pedagogical behaviors in a domain-independent context using a model-tracing (MT) approach (Cognitive Tutors [1] and Andes [2]), a constraint-based approach (ASPIRE [3]) or an example-tracing approach [4]. The most common pedagogical behaviors offered include [5]: immediate feedback indicating the correctness of a step (including flag feedback); next-step hints; error-specific feedback on anticipated incorrect steps and task selection. In most implementations, only immediate feedback and task selection are truly domain-independent, while next-step hints and error-specific feedback require additional domain-specific data (e.g., message templates) to be linked to the knowledge components.

Our hypothesis, in the context of problem-solving tasks in well-defined domains, is that, to express more sophisticated domain-independent behaviors, the model of the task domain must be defined more explicitly. With ASTUS, our MT framework, we want to increase the set of pedagogical behaviors that can be implemented independently from a task domain. To that end, the task model is defined using knowledge components that are examinable by the framework's different modules [6, 7] and interaction templates that match the learner's actions in the learning environment [8]. The tutor's pedagogical module can use those features to produce its behaviors, hence allowing the implementation of sophisticated interactions with the learner and the environment. For example, in ASTUS's procedural model, the knowledge required to execute a task is defined using goals (intentions) and procedures (dynamic plans). Procedures satisfy goal using domain-independent algorithms (sequence, iteration, condition) examined by the tutor to produce feedback.

2 Objective

We want to show that it is possible to implement sophisticated domain-independent behaviors that are closer to those found in domain-specific tutors. In particular, our goal is to illustrate that ASTUS's representation of a task domain (the knowledge and the interactions with the learning environment) can be used to implement such behaviors in an authoring framework. The research presented in this paper concerns the behaviors' implementation. Empirical studies would be required to evaluate the learning gains resulting from more sophisticated pedagogical behaviors.

3 Methodology

We look at the ITS's literature to find pedagogical behaviors that are widespread or that have been shown to improve learning. We then generalize these behaviors to a domain-independent context and describe them in terms of ASTUS's modeling approach. The expected result is the functional implementation of the pedagogical behaviors in a domain-independent ITS framework. Failure to implement a specific behavior gives us insights about 1) how to improve ASTUS to allow the integration of additional pedagogical behaviors and 2) the limits of which types of behaviors can be generalized to such a framework.

An example of a pedagogical behavior that can be implemented in a domain-independent context is the generation of hints. It is the generalization of a widespread behavior (allowing the learner to ask for a hint about the next step) [5], that is rarely integrated using a truly domain-independent implementation. For example, in the Cognitive Tutors, hints are message templates associated to a specific production rule that must be entered by the tutor's author [1]. Barnes and Stamper [9] worked on hint generation in their logic proof tutor in which every hint given to the learner is generated. Their approach has two main downsides: 1) the hint generation is specific to the logic proof task and 2) it cannot generate hints for every problem solving alternatives (approximately 80% are covered) since the model is built by examining the learners' previous solutions. With ASTUS, we are able to generate hints independently from the modeled task domain, for every pedagogically relevant problem-solving path. To achieve this behavior, we use domain-independent message templates that are associated to each of the knowledge component types. When a hint is requested, the tutor can instantiate the templates by examining the knowledge contained in the model of the task domain. ANDES [2] has previously done similar work where hints are generated using the knowledge components ("Problem Solving Methods"), but this feature has not been the object of a publication. Additional work on hint generation would be to replace our template by natural language generation.

In addition to hint generation, we intend to implement other domain-independent pedagogical behaviors such as: generating worked examples or self-explanation hooks from problem-solving task; mastery learning based task selection and error recognition according to the task's model. Previous work on interactions with the learning environment [8] allowed problem-solving demonstrations where one or more steps are performed with full visual support (e.g., the mouse is moved on the screen).

4 Conclusion

Implementing sophisticated domain-independent pedagogical behaviors in an ITS framework reduces the costs of authoring tutors since 1) it keeps the efforts centered on modeling the task domain and 2) it reduces the number of pedagogical behaviors requiring domain-specific encoding (e.g., domain-specific message templates). The research presented in this paper will allow the members of the ITS community to benefit from these features since ASTUS is designed to be available for them to develop their own tutors and conduct their own experiments [10].

ASTUS's implementation of sophisticated pedagogical behaviors depends on its explicit model of the task domain. Previous study [6] has shown that the workload required to create such a model is similar to creating a Cognitive Tutor using CTAT. While authoring in ASTUS currently require more time, efforts are made to offer tools that reduce authoring time [10].

References

1. Anderson, J.R., Corbett, A.T., Koedinger, K.R., Pelletier, R.: Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences* 4(2), 167–207 (1995)
2. VanLehn, K., Lynch, C., Schulze, K., Shapiro, J.A., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., Wintersgill, M.: The Andes Physics Tutoring System: Lessons Learned. *International Journal of Artificial Intelligence in Education* 15(3), 1–47 (2005)
3. Mitrovic, A., Suraweera, P., Martin, B., Zaharov, K., Milik, N., Holland, J.: Authoring Constraint-Based Tutors in ASPIRE. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) *ITS 2006*. LNCS, vol. 4053, pp. 41–50. Springer, Heidelberg (2006)
4. Alevan, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A New Paradigm for Intelligent Tutoring Systems: Example-Tracing Tutors. *IJAIED, Special Issue on Authoring Systems for Intelligent Tutoring Systemes*, 105–154 (2010)
5. VanLehn, K.: The Behavior of Tutoring Systems. *International Journal of Artificial Intelligence in Education* 16(3), 227–265 (2006)
6. Paquette, L., Lebeau, J.-F., Mayers, A.: Authoring Problem-Solving Tutors: A Comparison Between ASTUS and CTAT using the Mutli-Columns Subtraction Task Domain. In: *Advances in Intelligent Tutoring Systems*, Springer, Heidelberg (to appear)
7. Lebeau, J.-F., Fortin, M., Paquette, L., Mayers, A.: From Cognitive to Pedagogical Knowledge Models in Problem-Solving ITS Frameworks. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Graesser, A. (eds.) *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, pp. 731–733 (2009)
8. Fortin, M., Lebeau, J.-F., Abdessamed, A., Courtemanche, F., Mayers, A.: A Standard Method of Developing User Interfaces for a Generic ITS Framework. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 312–322. Springer, Heidelberg (2008)
9. Barnes, T., Stamper, J.: Toward Automatic Hint Generation for Logic Proof Tutoring Using Historical Student Data. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *Proceedings of the 9th International Conference on Intelligent Tutoring Systems*, pp. 373–382. Springer, Berlin (2008)
10. Lebeau, J.-F., Paquette, L., Fortin, M., Mayers, A.: An Authoring Language as a Key to Usability in a Problem-Solving ITS Framework (YRT). In: *ITS (2010)*

Delivering Tutoring Feedback Using Persuasive Dialogues

Amin Rahati and Froduald Kabanza

Department of Computer Science,
University of Sherbrooke
Sherbrooke, Quebec J1K 2R1
Canada

{amin.rahati,kabanza}@usherbrooke.ca

Abstract. We are developing a general argumentation framework for implementing tutoring feedback in the form of persuasive dialogues. The objective is to have an intelligent tutoring system capable of arguing with the student to convince him of the rationale of the feedback provided to him. The application domain is that of medical diagnosis skill learning.

Keywords: intelligent tutoring system, argumentation, persuasive dialogue.

1 Introduction

Many current intelligent tutoring systems (ITS) implement tutoring feedback using dialogues [1,2,3]. It has already been argued that engaging the students in argumentative dialogues would efficiently foster their knowledge construction, by making them think about the content and sequence of arguments they put forward [4]. Through argumentation, the ITS could provide persuasive tutoring feedback by giving students the opportunity to challenge feedback provided by the ITS as well as the opportunity to defend their position in problem solving. However, so far argumentation was involved in ITS only as a skill to learn [2,5] –that is, the ITS teaches how to argue – as opposed to using argumentation as a persuasive dialogue delivering tutoring feedback on a given problem solving or skill learning task.

In our approach, every problem solving action (PSA) performed by a student to solve a problem is considered as an argument. The ITS intervenes to help the students also by making arguments. Errors made by the student are considered as disagreements and the ITS tries to help the student remedy them through an argumentation. Our focus is on the structural level of dialogues, and we are not concerned with speech generation, speech recognition and natural language understanding, even though we acknowledge the important contribution of these approaches in learning.

2 Argumentation Framework

As figure 1(a) presents, the framework has three main components represented by the black circles. The first component defines the representation of dialogue moves. A typical dialogue move specifies the content of an argument or a propositional attitude in the exchange of arguments. Any PSA or utterance has a predefined argument template. A move type is a template operator described by a precondition specifying when the move is feasible and an effect specifying the update of the dialogue state.

At any point during the interaction, each arguer (ITS or student) is committed to his own arguments he has previously asserted (either by making PSAs or by making utterances during verbal exchanges) and has not withdrawn yet; and to his opponent’s arguments that he has accepted. A structure called the “Commitment Store” keeps track of the current commitments. As argued by [6], the commitment concept provides a means to settle a conflict between arguers by making the opponent commit to the proponent’s assertion or the proponent withdraws of its assertion.

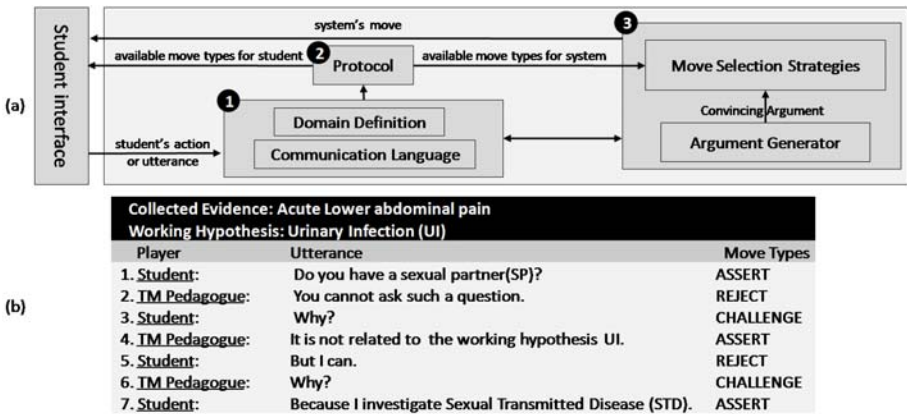


Fig. 1. (a)Architecture of the system; (b)An example scenario

The second component, the protocol, uses hierarchical state diagrams to regulate the moves and define the permitted move sequences in an argumentation dialogue. The third component calculates arguments which are persuasive for the student. For this, given an assertion made by the student, we follow Walton’s argumentation theory [7] by specifying rules expressing how to respond to arguments made by the opposing party in a two-participant argumentation. These argument generation rules (AGR), called test questions by Walton, specify arguments that can challenge assertions made by the student - PSAs as well as utterances during a conflict settling dialogue. A counterargument is also subjected to AGRs and this process continues until no match is found for AGRs.

A convincing argument exists among this set of generated arguments if it defeats the student's argument but it is not defeated by any counterargument. To calculate such argument we adapted a decision-theoretic argumentation method from [8].

Given a convincing argument and a set of moves, the ITS uses a dialogue strategy to select its current move. We specify a dialogue strategy by following a three-level methodology, inspired from [9]. The Higher level includes some strategic rules that we use as domain dependent tactic to select a move based on a pedagogic goal. For instance, figure 1(b) is an excerpt from a medical diagnosis learning scenario adapted from [3] and modified to reflect the argumentative capability that we are aiming for. In this scenario, the student is in the process of diagnosing a simulated patient and she has so far gathered a number of evidences and has formulated a list of hypotheses. At line 1 of this scenario ITS notices inconsistencies between the evidences and the hypotheses formulated by the student. This matches a counter-argument which spawns further argumentation with the student (line 1 to 7) to settle the disagreement.

3 Conclusion

Our paper appearing in the 2010 Intelligent Tutoring Systems conference provides details on the current implementation of our approach and its present capabilities.

References

1. Graesser, A.C., VanLehn, K., Rosé, C.P., Jordan, P.W., Harter, D.: Intelligent tutoring systems with conversational dialogue. *AI Magazine* 22(4), 39–51 (2001)
2. Yuan, T., Moore, D., Grierson, A.: A human-computer dialogue system for educational debate, a computational dialectics approach. *International Journal of Artificial Intelligence in Education* 18, 3–26 (2008)
3. Kabanza, F., Bisson, G., Charneau, A., Jang, T.S.: Implementing tutoring strategies into a patient simulator for clinical reasoning learning. *Journal of Artificial Intelligence In Medicine (AIIM)* 38, 79–96 (2006)
4. Karsten, S., Weinberger, A., Fischer, F.: Facilitating argumentative knowledge construction with computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning* 2(4), 421–447 (2007)
5. Pinkwart, N., Alevén, V., Ashley, K., Lynch, C.: Evaluating legal argument instruction with graphical representations using largo. In: *Proceeding of the 2007 conference on Artificial Intelligence in Education*, pp. 101–108 (2007)
6. Walton, D.: *Argument Structure: A Pragmatic Theory*. University of Toronto Press (1996b)
7. Walton, D.: *Argumentation Schemes for Presumptive Reasoning*. Erbaum, Mahwah (1996a)
8. Bench-Capon, T.J.M.: Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation* 13(3), 429–448 (2003)
9. Moore, D.: *Dialogue game theory for intelligent tutoring systems*. Ph.d. dissertation, Leeds Metropolitan University, Leeds, UK (1993)

Coordinate Geometry Learning Environment with Game-Like Properties

Dovan Rai, Joseph E. Beck, and Neil T. Heffernan

Computer Science, Worcester Polytechnic Institute
{dovan, josephbeck, nth}@wpi.edu

Abstract. We want to create coordinate geometry learning environment with game-like properties, that is, elements of games that are engaging such as cover story, graphical representation, and animated feedback. This paper proposes that adding game-like properties to a computer tutor results in more student engagement and interest in the material. However, in addition to taking instructional time away, adding such properties imposes new limitations and difficulties in constructing content. Therefore, we have taken a measured and minimalist approach to making the original environment more game-like making a balance between stimulation and overload.

1 Introduction

Although games are engaging to motivate students, they tend to take up time that could have been used for instruction, and have been empirically shown to be less effective than intelligent tutors when it comes to learning gains [1]. Hence, instead of completely integrating educational content into a game framework, we instead choose to incorporate into the tutor those features of games that are motivational but do not overly detract from learning. Based on this choice, we created a learning environment for coordinate geometry with *game-like properties*. We define game-like properties as elements of games that are responsible for their engaging nature such as points and rewards, graphics, fantasy, interactive feedback, speeded trials, leveling up, etc. Our reason for taking this viewpoint is that we wish to make computer tutors more engaging and game-like, but do not want to start by trying to design an educational game. Instead, we are taking a measured and minimalist approach by incrementally making a complete tutor more game-like by weighing each additional component in terms of retaining all the learning features of a tutor and minimizing the limitations, while exploiting the benefits of games. Our hypothesis is that the “sweet spot” of educational efficacy and engagement is closer to computer tutors than it is to educational games. Therefore, we feel that a strategy of gradually and incrementally adding game-like properties to a tutor is productive.

Our learning environment consists of a series of 8th grade (approximately 13-year olds) coordinate geometry problems wrapped in a visual cover story. We started with a story Mily’s World [4], where students have to help a 9-year old girl to solve her problems. Based on feedback that Mily was for younger students, we are revising our design and are making use of humor and mischief, which are hopefully more

appealing to the age-group, in a new story called *Monkey's Revenge*: A boy is thrown outside of his class for playing a game on his cell phone and encounters a monkey and they become friends. He builds a house for the monkey, but the monkey is not eager to become domesticated and destroys the house, steals his phone and runs away. The boy tries to get back his phone by throwing balls to the monkey. To move the story forward, the students have to solve coordinate problems like calculating distance between the boy and the monkey, slope of the roof and walls of the house, finding points where the monkey tied to a rope cannot find bananas and finally figure out slopes, intercepts and equations of the line of the path of the ball.

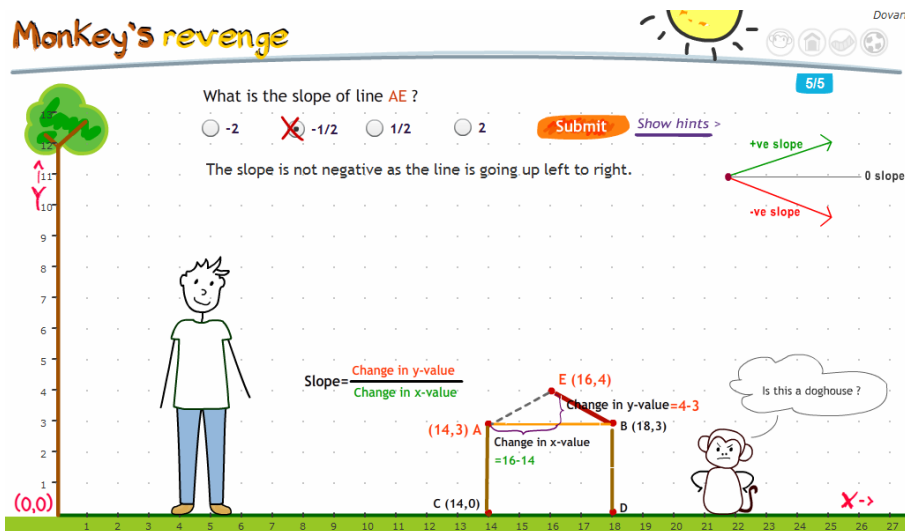


Fig. 1. Screenshot of Monkey's revenge

2 Conceptual Framework

Game-like properties are the elements of game that make it engaging, for example: graphics, animated feedback, story, etc. Educational software already uses combination of these properties, but we categorize them as game-like properties for their engaging, interactive and imaginative elements. Besides the benefit of engagement, the addition of these properties can add new costs like taking instructional time away and increasing cognitive load among students. Hence, we want to analyze the utility of these individual elements in the framework of stimulation and overload. To minimize the costs of cognitive load, we are using the following strategies:

Simple environment and minimalistic presentation: Regular games can afford to have new and complex environment with complicated rules. But for educational games, especially the ones based on conventional curriculum, the students should not be overwhelmed by too many details.

Concreteness fading: As the problems get harder, they tend to be more abstract and it is harder and counterintuitive to have concrete representations. Therefore, we have used a strategy to make the representations more concrete at first (story characters shown as cartoon image) and less so as we proceed (story characters are abstracted to dots). Initial concrete grounding facilitates interpretation in later problems.

2.1 Game-Like Properties, Emotional Interest and Cognitive Interest

Games arouse sensory and emotional interest. We propose that the carefully integrated game components can make math problems more meaningful and challenging and also aid in more easily understanding the content, and thus arousing cognitive interest. We have included some game-like properties suited to our content:

- **Authentic activities:** Learning is more efficient and effective when it is embedded in realistic and relevant contexts [2].
- **Visual representation:** Graphics not only add appeal but they can help develop mental models, thus reducing the burden on working memory.
- **Storyline:** If we use a coherent cover story, the initial story context can be reused for multiple problems, thus saving effort to read context for each new word problem.
- **Animated immediate feedback:** With visual immediate feedback, students can tell what the error was and how it relates to the correct solution. For instance, a new house pops up; the monkey can eat a banana, etc.
- **Collection:** Students can collect badges after each level as they master a sub-skill. By tagging those badges with math skills, we want to make tighter bond between game-environment and content.
- **Building:** Students have to solve different problems to build a house. Using various sub-skills to create a single structure, students can see how different mathematical concepts can be integrated within a single entity.

In conclusion, while there have been many attempts to use game-like environment in tutors (e.g. *Wayang Outpost*), our approach is to make an iterative process in a space of ITS and games, within the framework of stimulation and overload. Our aim is to use game-like environment as a platform for rich learning experience arousing emotional as well as cognitive interest among students.

References

1. O'Neil, H., Wainess, R., Baker, E.: Classification of learning outcomes: Evidence from the computer games literature. *The Curriculum Journal* 16(4), 455–474 (2005)
2. Shaffer, D.W., Resnick, M.: “Thick” Authenticity: New Media and Authentic Learning. *Journal of Interactive Learning Research* 10(2), 195–215 (1999)
3. Harp, S.E., Mayer, R.E.: How Seductive Details Do Their Damage: A Theory of Cognitive Interest in Science Learning. *Journal of Educational Psychology*, American Psychological Association, Inc. 90(3), 414–434 (1998)
4. Rai, D., Beck, J.E., Heffernan, N.T.: Coordinate Geometry Learning Environment with Game-like Properties. In: Tenth International Conference on Intelligent Tutoring Systems, Pittsburgh, USA (in press 2010)

Long-Term Benefits of Direct Instruction with Reification for Learning the Control of Variables Strategy

Michael A. Sao Pedro, Janice D. Gobert, and Juelaila J. Raziuddin

Worcester Polytechnic Institute, 100 Institute Rd. Worcester, MA 01609, USA
{mikesp, jgobert, juelaila}@wpi.edu

Abstract. We compare three learning conditions on 57 middle school students' short- and long-term retention at applying the control of variables strategy. Collapsing over time, direct instruction with reification yielded more robust learning than either direct instruction without reification or discovery learning conditions as measured by skill at constructing unconfounded experiments.

Keywords: microworlds, virtual instruction, science education.

1 Introduction and Method

Currently, there is a debate in the science education community on the effectiveness of discovery versus direct instruction (e.g. [1]). In our previous work [2], we compared the effectiveness of two types of direct instruction, with and without reification (self-explanation), and discovery learning on middle school students' acquisition and transfer of the Control of Variables Strategy (CVS), a procedure for conducting controlled scientific experiments. Students in our study practiced CVS by designing experiments with two virtual ramp apparatus to determine if a given factor affected how far a ball would roll down a ramp. Ramp setups could initially be *singly confounded* (one extraneous variable is not controlled) or *multiply confounded* (more than one extraneous variable is not controlled), and were initially *uncontrasted* (the factor to test is unchanged). Students had to modify ramp setups to test a specified factor while controlling for the others. We found that in an immediate posttest, both direct conditions designed significantly more unconfounded experiments starting from a multiply confounded setup than the discovery condition. However, the direct conditions did not significantly differ from each other. In the present study, we examined if these findings were robust by retesting participants 6 months after our original study. We hypothesized participants in the direct+reify group would outperform the other conditions even though the direct conditions showed no significant differences at the immediate posttest, since self-explanation supports deep learning [3] and knowledge integration [4].

We used the Science ASSISTment System [5], a web-based intelligent tutoring system, to host our materials and run randomized controlled experiments. The pretest, immediate posttest, and delayed 6 month posttest required students to construct 4

unconfounded ramp setups, without receiving feedback, to determine different factors' effects on the outcome. Between the pretest and immediate posttest, students practiced CVS in a randomly assigned learning condition. In both direct conditions, students were first taught CVS in the context of the ramp and asked to evaluate if different ramp setups tested "for sure" that a factor affected the outcome. However, direct+reify students explained their reasoning whereas direct-no reify students did not. Discovery students continued attempting to construct unconfounded experiments receiving no feedback. For more details on the original procedure, see [2].

2 Results

We analyzed 57 students' immediate and delayed posttest scores to determine which condition(s) yielded better performance on skill in constructing unconfounded ramp setups using a repeated measures ANCOVA with pretest score as a covariate. Four students who did not take the original ramp pretest were excluded from this analysis. Means and standard errors for ramp performance over time are shown in Figure 1. Ramp setups demonstrating CVS for a given factor were scored 1 point, 0 otherwise. Within-subjects tests revealed no significant main effects for time, Wilks $\lambda=.97$, $F(1,49)=1.36$, $p=.249$, partial $\eta^2=.027$ and no significant interaction between time and condition, Wilks $\lambda=.92$, $F(2,49)=2.03$, $p=.143$, partial $\eta^2=.076$. However, condition was a significant between-subjects factor, $F(2,49)=3.26$, $p=.047$, partial $\eta^2=.117$, controlling for ramp pretest score. Post hoc comparisons revealed that the direct+reify condition constructed significantly more unconfounded experiments than the discovery condition ($M=0.92$, $SE=0.43$, $p=.037$, 95% $CI=[0.06, 1.79]$) and the direct-no reify condition ($M=0.98$, $SE=0.45$, $p=.033$, 95% $CI=[0.84, 1.88]$). No significant difference was found between the direct-no reify and discovery conditions, $p=.903$.

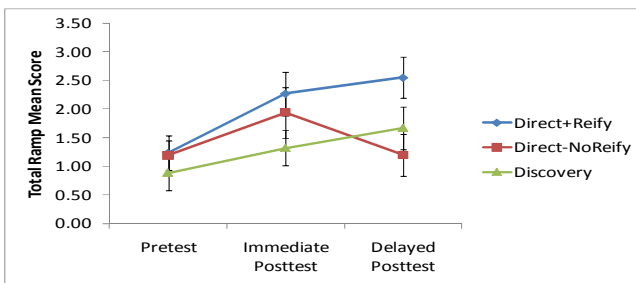


Fig. 1. Means and standard errors for total ramp score by condition, maximum score = 4

Since there was a significant between-subjects effect on condition, we analyzed group differences solely at the delayed posttest using an ANCOVA with ramp pretest score as a covariate. We found a main effect on condition, $F(2,49)=3.36$, $p=.043$, partial $\eta^2=.121$ with ramp pretest not being a significant covariate, $F(1,49)=1.38$, $p=.247$. Post hoc tests revealed that the direct+reify condition constructed significantly more unconfounded experiments in the delayed posttest than the direct-no reify

condition ($M=1.34$, $SE=0.53$, $p=.014$, 95% $CI=[0.29, 2.40]$). There were no significant differences between direct+reify and discovery, $p=.131$ nor direct-no reify and discovery, $p=.306$. As shown in Figure 1, the direct+reify condition maintained its higher performance at the immediate and delayed posttest. Though the direct-no reify condition improved at the immediate posttest compared to the pretest, the improvement is lost 6 months later at the delayed posttest. Also, the discovery condition's skills on this CVS authentic inquiry task increased as time progressed.

3 Conclusions and Future Work

On the Science Assistments project [5] we aim to assess inquiry skills such as hypothesizing, designing controlled experiments, collecting and analyzing data and formulating conclusions as they engage in inquiry using microworlds across different domains such as physics, chemistry, biology, and earth science. As a start, we researched the acquisition of a particular skill, CVS, which we feel is a necessary cornerstone for reasoning via inquiry. Our research suggests that the best way to attain timely and long-term skill in constructing unconfounded experiments is to combine direct instruction with reification. On a broader scale, we are using student action log files to create detectors of haphazard inquiry behavior as students engage in more open-ended inquiry tasks requiring the use of many inquiry skills. Example behaviors include running uncontrolled experiments and not testing stated hypotheses. These detectors will enable us in real-time to auto-score inquiry and determine those students requiring support. Leveraging these detectors, we can compare scaffolding strategies to identify those that are most effective for different kinds of students.

Acknowledgments. This research was funded by the National Science Foundation (NSF-DRL#0733286; NSF-DGE# 0742503) and the U.S. Department of Education (R305A090170). Any opinions expressed are those of the authors and do not necessarily reflect those of the funding agencies.

References

1. Kirschner, P., Sweller, J., Clark, R.: Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching. *Educational Psychologist* 41(2), 75–86 (2006)
2. Sao Pedro, M., Gobert, J., Heffernan, N., Beck, J.: Comparing Pedagogical Approaches for Teaching the Control of Variables Strategy. In: Taatgen, N.A., van Rijn, H. (eds.) *Proceedings of the 31st Annual Meeting of the Cognitive Science Society*, Austin, TX, pp. 1294–1299 (2009)
3. Chi, M.: Constructing Self-Explanations and Scaffolded Explanations in Tutoring. *Applied Cognitive Psychology* 10, S33–S49 (1996)
4. Linn, M., Hsi, S.: *Computers, Teachers, Peers: Science Learning Partners*. Erlbaum Associates, Mahwah (2000)
5. Gobert, J., Sao Pedro, M., Krach, N., Montalvo, O., Toto, E.: A Framework for Adaptive Scaffolding Based on Content Knowledge, Inquiry Skills, and Learner Characteristics. In: To appear at the Annual Meeting of the American Educational Research Association, Denver, CO (2010)

Can Affect Be Detected from Intelligent Tutoring System Interaction Data? – A Preliminary Study

Elizabeth A. Anglo and Ma. Mercedes T. Rodrigo

Department of Information Systems and Computer Science,
Ateneo de Manila University, Loyola Heights, Quezon City, Philippines
{eanglo,mrodrigo}@ateneo.edu

Abstract. This study attempted to determine if it is possible to create an automatic affect detector using a combination of semantic and keystroke data. While the resulting models attained detection accuracies comparable with other studies, their reliabilities were not ideal. One model however shows that interaction logs may have a potential as a detector for confusion.

Keywords: Aplusix, affect, interaction data, learner modeling.

1 Introduction

There is growing evidence that learning and emotion are tightly related, e.g., in [1]. Hence affect-sensitive intelligent tutoring systems (ITSs) may become even more effective. Automatic affect detection in ITSs is a challenging area of research. Finding a technology free of the problems associated with the current detectors is still largely an open question [2]. This paper explores the effectiveness of system interaction and semantic features from system logs to implement a real-time affect detector for an ITS.

2 Methodology

Data on the affective states that occur during learning were collected in a separate study [3] involving first and second year high school students as they interacted with Aplusix II: Algebra Learning Assistant [4], an ITS for mathematics. The system interaction (e.g., usage counter for arrow keys) and semantic data (e.g., number of problems attempted in an observation and their difficulty) were gathered from the system logs. The system interaction and semantic data were then synchronized with the affective observations using an unsupervised action filter based on a variable time window, as discussed in [5]. The synchronized logs contained 3, 000 records with each record having 58 features. Three separate datasets were created, with each dataset corresponding to an affective state, i.e., *boredom*, *confusion*, and *frustration*. This paper focuses on these affective states since they have been associated with learning, e.g., in [5]. We next applied 10 data mining approaches, including Bayesian, functions, rules and decision trees from RapidMiner [6] after which the results were examined for classification accuracy, reliability as measured using Cohen's kappa [7], and agreement with results established from previous studies.

3 Results, Discussion and Conclusion

Table 1 presents the average accuracy, kappa values, and highest number of correct affect guesses obtained from the program runs. The classification accuracies were comparable to each other, that is, no data mining approach was significantly better than the other in classifying the records. However, there are differences in the kappa values and number of correct affect guesses, e.g., the Functional Tree model produced the highest kappa value and made the most number of correct confused state guesses. The accuracy values are relatively good compared to accuracies using other technologies. The small number of records for each observed affect may make the effectiveness of the models debatable. An examination of the confusion matrices revealed how the models work. For boredom, the models detected 99% of the non-bored instances and 4% of the bored instances; for confusion, the models detected 97% of the non-confused records and 13% of the confused instances; and for frustration, the models detected 99% of the non-frustrated instances and only 1% of the frustrated instances. From these results, we conclude that while still not impressively accurate, at least in the case of the model for confusion, given the detrimental effects of this affect, this model offers us some detection capability better than chance alone. While the reliability value was far from being impressive, one may note that it is also the confusion model that gave the highest kappa value. This may mean that the interaction logs indeed have potential for detecting at least the state of confusion.

Table 1. Average Accuracy, Kappa Values, and Highest Number of Correct Guesses

	Boredom	Confusion	Frustration
Number of Records for Affect (% to Total Number of Records in Dataset)	69 (3%)	393 (14%)	73 (3%)
Average Classification Accuracy	97.05%	84.08%	97.02%
Best Kappa Value	0	0.09	0.02
Highest No. of Correct Affect Guesses	3	50	1

Next we observe how the resulting confusion model agrees with other studies. The Functional Tree model characterized a student in the confused state as someone who attempts a smaller number of problems and who works on a bigger number of easy problems compared to the other students. Since confusion precedes gaming and co-occurs gaming [8], we can say that the latter rule in a way indirectly concurs with previous work, e.g., [5] which showed problem difficulty may lead to gaming.

Our ultimate goal is to determine whether system interaction logs are useful for affect detection. In this preliminary analysis, we showed that the system logs offer detection capability for confusion better than chance alone. The model that showed most promise of detection capability was that for confusion. We observe this affect has the most number of occurrences among the three states studied in this paper. This might suggest that by lessening the imbalance in the dataset, we might be able to create better detectors. This is one of the directions that this research will consider next.

Acknowledgments. We thank Jean-Francois Nicaud of the Laboratoire d'Informatique de Grenoble for the use of Aplusix. We thank Sheryl Ann Lim, Alexis Macapanpan, Sheila Pascua, Jerry Santillano, Jessica Sugay, Sinath Tep, and Norma Jean Viehland, and Dr. Ma. Celeste T. Gonzalez for their assistance in organizing and conducting the studies reported here. We also thank the Ateneo de Manila High School, Kostka School of Quezon City, School of the Holy Spirit of Quezon City, St. Alphonsus Liguori Integrated School and St. Paul's College Pasig for their participation in the studies conducted. This research undertaking was made possible by the Philippines Department of Science and Technology Engineering Research and Development for Technology Consortium under the project "Multidimensional Analysis of User-Machine Interactions Towards the Development of Models of Affect". We thank Dr. Ryan Baker of the Department of Social Science and Policy Studies, Worcester Polytechnic Institute, for his invaluable support and guidance.

References

1. Craig, S., Graesser, A., Sullins, J., Gholson, B.: Affect and learning: an exploratory look into the role of affect in learning with AutoTutor. *Journal of Educational Media* 29(3), 241–250 (2004)
2. Anglo, E.A.: Review – Automatic Affect Detection Technologies. *Loyola Schools Review, School of Science and Engineering VIII* (2009) (in press)
3. Rodrigo, M. M.T., Baker, R.S.J.d., D'Mello, S.K., Gonzalez, M. C.T., Lagud, M.C.V., Lim, S.A.L., Macapanpan, A.F., Pascua, S.A.M.S., Santillano, J.Q., Sugay, J.O., Tep, S., Viehland, N.J.B.: Comparing learners' affect while using an intelligent tutoring system and a simulation problem solving game. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008. LNCS*, vol. 5091, pp. 40–49. Springer, Heidelberg (2008)
4. Nicaud, J.-F., Bouhineau, D., Chaachoua, H.: Mixing microworld and CAS features in building computer systems that help students learn algebra. *International Journal of Computers for Mathematical Learning* 9, 169–211 (2004)
5. Walonoski, J.A., Heffernan, N.T.: Detection and Analysis of Off-Task Gaming Behavior in Intelligent Tutoring Systems. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) *ITS 2006. LNCS*, vol. 4053, pp. 382–391. Springer, Heidelberg (2006)
6. Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., Euler, T.: Yale (now: RapidMiner): Rapid Prototyping for Complex Data Mining Tasks. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2006* (2006)
7. Cohen, J.: A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20(1), 37–46 (1960)
8. Rodrigo, M.M.T., Baker, R.S.J.d., Lagud, M.C.V., Lim, S.A.L., Macapanpan, A.F., Pascua, S.A.M.S., Santillano, J.Q., Sevilla, L.R.S., Sugay, J.O., Tep, S., Viehland, N.J.B.: Affect and usage choices in simulation problem-solving environments. In: Luckin, R., Koedinger, K.R., Greer, J. (eds.) *Artificial Intelligence in Education: Building Technology Rich Learning Contexts that Work*. IOS Press, Amsterdam (2007)

Comparing Disengaged Behavior within a Cognitive Tutor in the USA and Philippines

Ma. Mercedes T. Rodrigo¹, Ryan S.J.d. Baker², Jenilyn Agapito¹, Julieta Nabos¹,
Ma. Concepcion Repalam¹, and Salvador S. Reyes Jr.¹

¹ Department of Information Systems and Computer Science,
Ateneo de Manila University, Quezon City, Philippines
mrodrigo@ateneo.edu, {jen_agapito, julietnabos}@yahoo.com,
{conrepalam, jayr02}@gmail.com

² Department of Social Science and Policy Studies, Worcester Polytechnic Institute,
Worcester MA, USA
rsbaker@wpi.edu

Abstract. We study how student behaviors associated with engagement differ across different school settings. We present a study to investigate the variation in gaming the system and off-task behavior in schools in the USA and Philippines, using quantitative field observations on students using the same Cognitive Tutor lesson on scatterplots. We find that students in the Philippines go off-task significantly less but game the system significantly more than our sample of students in the USA. This study suggests that ITS designed for different settings or used in different settings will need to emphasize adaptation to different disengaged behaviors.

Keywords: gaming the system, off-task behavior, school context.

1 Introduction

In recent years, intelligent tutoring systems have left the research laboratory, expanded beyond the research classroom, and have started to see large-scale use worldwide [3, 5], creating the potential to use intelligent tutors to study cross-cultural differences in learners [cf. 5]. Beyond enabling scientific discoveries in this domain, greater attention to cross-cultural and cross-setting student differences in intelligent tutors has the potential to enable culturally-sensitive intelligent tutors that are educationally effective for a broader community of learners.

In this paper, we study how student behaviors associated with engagement differ across different school settings, comparing the frequency of gaming the system and off-task behavior in the USA and Philippines. Off-task behavior is much less common in East Asia than in the USA [1, 7], including in classrooms using educational software [7]. However, it remains unclear why students go off-task to such different degrees in East Asian and Western classrooms. Thus far, the primary hypothesis for this difference is that cultural factors explain the difference in incidence of off-task behavior [1]. However, it is also known that curricula are very different between East

Asia and Western countries. We control for this possibility by using the exact same intelligent tutor and study design in both East Asian and American classrooms.

2 Methods

53 students in two public schools in the suburbs of Pittsburgh, PA, and 60 students in a public school in an urban area of Quezon City, Manila, participated in this study. The participating schools in both countries consisted predominantly of students from the local ethnic majority (e.g. Filipino students in the Philippines, white students in the USA). Students in both countries were in mainstream mathematics classes (e.g. neither gifted nor special needs).

In both studies, student ages ranged from approximately 12 to 14. The schools in the USA regularly use intelligent tutoring systems and other types of educational software, whereas the schools in the Philippines do not typically use these technologies (rather than a confound, we consider this an inherent attribute of two settings, as educational software remains rare in Philippines public schools [cf. 6]).

All students used a short Cognitive Tutor unit on scatterplot generation and interpretation [2], a topic not previously covered in class, for 80 minutes. We collected data on each student's pattern of behavior during tutor usage, using the exact quantitative field observation procedure from [2]. Our coding scheme consisted of six categories: on-task, on-task conversation, off-task conversation, off-task solitary behavior, inactivity, and gaming the system.

3 Results

The incidence of both categories of behavior was highly different between the schools in the two countries. Students in the USA were off-task an average of 19.7% of the time ($SD=17.8\%$), within the typical range reported in traditional classrooms in the USA. Students in the Philippines were off-task an average of 2.7% of the time ($SD = 5.2\%$), in line with previous observations of student off-task behavior in classrooms using educational software in the Philippines [7]. The difference between off-task behavior in the two countries was statistically significant, $t(111)=7.02$, $p<0.0001$, effect size = 3.26 SD.

Students in the USA gamed the system about 5.3% of the time ($SD= 9.9\%$), in line with previously observed gaming frequencies in previous studies of the scatterplot tutor lesson [cf. 2]. Students in the Philippines gamed the system about 10.7% of the time ($SD=15.3\%$). The difference in gaming the system frequency between the two countries was statistically significant, $t(111)=2.17$, $p=0.03$, effect size = 0.54 SD.

4 Discussion and Conclusions

In this paper, we have presented a study examining the prevalence of student behaviors associated with disengagement in the USA and Philippines. This study controlled for method and learning environment – hence, differences found can be attributed to differences between the schools and/or their populations. The study found that

off-task behavior was significantly higher in the USA, but that gaming the system was significantly more frequent in the Philippines. One possible account for this finding is that Filipinos value social acceptance, respect for elders, and discretion more than Americans do [4]. In the school context, students might have considered off-task behavior as indiscreet and disrespectful to the teacher, and therefore socially unacceptable. Gaming has the appearance of being on-task, at least from a distance. Replicating the analyses presented here, to see whether the same pattern is seen in other East Asian countries and other Western countries, will be an important area of future work. In the long-term, this work may enable understanding of how intelligent tutors should differ in different countries.

Acknowledgements. This research was supported by NSF grant REC-043779 to “IERI: Learning-Oriented Dialogs in Cognitive Tutors: Toward a Scalable Solution to Performance Orientation”; by the Pittsburgh Science of Learning Center (National Science Foundation) via grant “Toward a Decade of PSLC Research”, award SBE-0836012; and by the Philippines Department of Science and Technology Engineering Research and Development for Technology Consortium under the project “Multidimensional Analysis of User-Machine Interactions Towards the Development of Models of Affect”.

References

1. Abiko, T., George, P.S.: Education for early adolescents in Japan, U.S.: Cross cultural observation. *NASSP Bulletin: Official Journal of the National Association of Secondary School Principals* 70, 74–81 (1986)
2. Baker, R.S., Corbett, A.T., Koedinger, K.R., Wagner, A.Z.: Off-Task Behavior in the Cognitive Tutor Classroom: When Students Game The System. In: *Proceedings of ACM CHI 2004: Computer-Human Interaction*, pp. 383–390 (2004)
3. Koedinger, K.R., Corbett, A.T.: Cognitive tutors: Technology bringing learning sciences to the classroom. In: Sawyer, R.K. (ed.) *The Cambridge handbook of the learning sciences*. Cambridge University Press, New York (2006)
4. Lynch, F.: Social acceptance reconsidered. In: Yengoyan, A.A., Makil, P.Q. (eds.) *Philippine Society and the Individual: Selected Essays of Frank Lynch (1949-1976)*, Center for South and Southeast Asian Studies, University of Michigan, USA (1984)
5. Nicaud, J.F., Bittar, M., Chaachoua, H., Inamdar, P., Maffei, L.: Experiments With Aplusix In Four Countries. *International Journal for Technology in Mathematics Education* 13(1) (2006)
6. Rodrigo, M.M.T.: Quantifying the divide: A comparison of ICT usage of schools in Metro Manila and IEA-surveyed countries. *International Journal for Educational Development* 25, 53–68 (2005)
7. Rodrigo, M.M.T., Baker, R.S.J.d., Lagud, M.C.V., Lim, S.A.L., Macapanpan, A.F., Pascua, S.A.M.S., Santillano, J.Q., Sevilla, L.R.S., Sugay, J.O., Tep, S., Viehland, N.J.B.: Affect and Usage Choices in Simulation Problem Solving Environments. In: *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, pp. 145–152 (2007)

Adaptive Tutorials for Virtual Microscopy: A Design Paradigm to Promote Pedagogical Ownership

Dror Ben-Naim¹, Gary Velan², Nadine Marcus¹, and Michael Bain¹

¹ School of Computer Science and Engineering

² School of Medical Sciences, Faculty of Medicine
University of New South Wales, Sydney, Australia

Abstract. A key factor in the successful involvement of teachers in the development of intelligent tutoring systems (ITS) is a development paradigm that accommodates teachers' skills and goals. In this paradigm, a mental model that is meaningful from the teacher's perspective must be created for each task. We have focused on supporting teachers throughout the process of developing, deploying and analyzing Adaptive Tutorials that use Virtual Slides (ATuVS), which were created to assist learning of microscopic morphology. This was facilitated by the Virtual Apparatus Framework (VAF) – an ITS architecture that enables development of online learning activities analogous to real-world laboratory activities. VAF allows us to develop authoring tools that follow a well-established pedagogical process, which teachers can easily work with. In order to evaluate the effectiveness of VAF as a teacher-oriented design paradigm, we introduce the concept of “pedagogical ownership”. We argue that mainstream adoption of ITS in general, and ATuVS in particular, is only possible if teachers can assert pedagogical ownership over them.

Keywords: ITS architecture, Adaptive Tutorials, Virtual Apparatus Framework, Pedagogical Ownership, Virtual Microscopy.

1 Introduction

While researchers in the field of ITS are concerned with different aspects of how to build systems that adapt to learners, there is surprisingly little discussion with regard to bridging what seems to be a growing gap between the ITS research community and the wider educational community. The long-standing approach to involve teachers in the ITS process is through the development of Authoring Tools ([1],[2]). However, to-date the overall success of this approach is questionable. We suggest that in order to improve acceptance of ITSs by teachers, the *mental model* [3] for the task of creating them should be in line with teachers' expectations and understanding of similar processes and systems.

2 Adaptive Tutorials Using Virtual Slides

Since 2002, the School of Medical Sciences at the University of New South Wales (UNSW) has progressively replaced microscopic examination of glass slides in

practical classes with computer-based virtual microscopy[4]. Virtual Slides (VS) are high-magnification digital scanned images of tissue sections that can be viewed via a web browser in a manner that closely simulates examination of glass slides with a real microscope. In the past two years we developed Adaptive Tutorials that use Virtual Slides. These created a learning environment within which students can be asked to answer questions and perform tasks that involve *interacting* with VS, while being provided with individualized, adaptive remediation. Such interactions include: annotating, labeling, region marking and more.

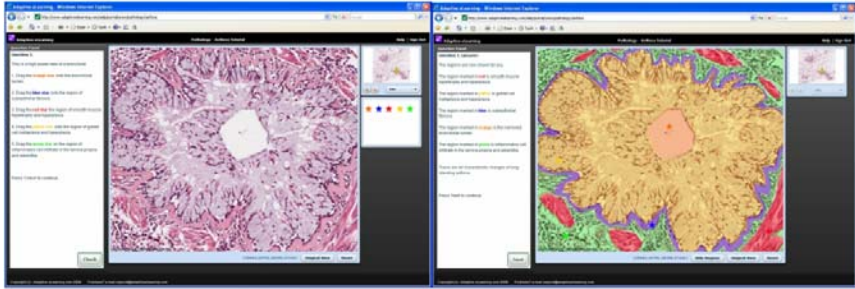


Fig. 1. A question from an Adaptive Tutorial on Asthma requiring identification of salient features in an affected airway wall by dragging colored markers onto the virtual slide (*left*), and the system's informative and intervention feedback where the ROIs in the problem were colored for the user (*right*)

Based on successfully piloting ATuVS [5], the Faculty of Medicine is in the process of developing over 500 ATuVS for all Pathology practical classes in Medicine and Science programs. Such a large content development project presents technical, pedagogical and administrative challenges. We are therefore interested in developing tools that facilitate rapid content development. Such tools must enable teachers to create their own ATuVS, use them in their teaching, and adapt and refine them based on an evaluation of their effectiveness. Our motivation is to enable teachers to “pedagogically own” ATuVS in the same way they own any other educational resources such as lecture notes, assignments, exam questions, laboratory exercises and so forth. We define Pedagogical Ownership as: *understanding of the content of instructional materials, the delivery mechanism(s) and the pedagogy underpinning the process*. The owner is able to develop and deliver the content to learners, to reflect on the effectiveness of that content, and therefore adapt it to better suit the learning needs of students.

In order to facilitate pedagogical ownership of ATuVS, we developed these resources using the Virtual Apparatus Framework (VAF). VAF is an eLearning content design paradigm inspired by the simplicity and elegance of the teaching laboratory[6]. Its premise is that teachers should be able to develop electronic courseware in a way that is analogous to how they develop laboratory activities. In other words, they need not be concerned about building the software or understanding exactly how it works, but rather they should be able to import prefabricated “apparatus” into a learning environment, and then author lesson plans that guide students through interaction with the apparatus. VAF's basic building blocks - Virtual Apparatus (VA) - are virtual

equivalents to real-world laboratory equipment. As such they include simulations or software tools with which students interact in the context of an online, laboratory-like activity.

Based on our work developing over 15 ATuVS and more than 20 Adaptive Tutorials in other domains using VAF, evidence has emerged that the teacher's role in defining tasks, specifying what constitutes a correct state, defining incorrect states, and attaching adaptive feedback to them can be seen as analogous to the task of training laboratory demonstrators. Such demonstrators might be "naive" or entirely unfamiliar about the topic for a particular laboratory class. In our experience, teachers readily engage with this mental model of the task. Teachers reported high levels of pedagogical ownership over ATuVS. As a consequence of this task analysis, work has begun to develop authoring tools that facilitate each step in the authoring process. This will enable teachers to create and edit ATuVS independently.

While concepts developed within this framework have been tested using ATuVS, we suggest that this framework is appropriate for Adaptive Tutorials or ITS in any domain. Furthermore, we argue that evaluating the level of pedagogical ownership teachers can assert over ITS content could be a benchmark of whether an ITS authoring paradigm is teacher-oriented, and thus whether it is likely to be effectively utilized.

References

1. Brusilovsky, P., Knapp, J., Gamper, J.: Supporting teachers as content authors in intelligent educational systems. *Int. J. Knowledge and Learning* 2(3/4), 191–215 (2006)
2. Razzaq, L., Patvarczki, J., Almeida, S.F., Vartak, M., Feng, M., Heffernan, N.T., Koedinger, K.R.: The ASSISTment Builder: Supporting the Life Cycle of Tutoring System Content Creation. *IEEE Transactions on Learning Technologies* 2, 157–166 (2009)
3. Johnson-Laird, P.N.: *Mental models: towards a cognitive science of language, inference and consciousness*. Cambridge University Press, Cambridge (1983)
4. Kumar, R.K., Velan, G.M., Korell, S.O., Kandara, M., Dee, F.R., Wakefield, D.: Virtual microscopy for learning and assessment in pathology. *The Journal of Pathology* (204), 613–618 (2004)
5. Velan, G., Ben-Naim, D., Kumar, R., Bain, M., Kan, B., Marcus, N.: Adaptive Tutorials Using Virtual Slides to Enhance Learning of Microscopic Morphology. In: Richards, G. (ed.) *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2009*, Chesapeake, VA (2009)
6. Ben-Naim, D., Marcus, N., Bain, M.: Virtual Apparatus Framework Approach to Constructing Adaptive Tutorials. In: Hamid, R., Arabnia, A.B. (eds.) *The 2007 International Conference on E-Learning, E-Business, Enterprise Information Systems, and E-Government*, pp. 3–10. CSREA Press, Las Vegas (2007)

The Online Deteriorating Patient: An Adaptive Simulation to Foster Expertise in Emergency Decision-Making

Emmanuel G. Blanchard, Jeffrey Wiseman, Laura Naismith,
Yuan-Jin Hong, and Susanne P. Lajoie

ATLAS Laboratory, Faculty of Educational and Counselling Psychology
McGill University, Montreal, Canada

{emmanuel.blanchard, jeffrey.wiseman, susanne.lajoie}@mcgill.ca,
{laura.naismith, yuan-jin.hong}@mail.mcgill.ca

Abstract. The deteriorating patient activity (DPA) is a low-fidelity educational simulation that prepares medical students to effectively approach emergency situations. This paper outlines how we have captured and represented expertise in rapidly changing emergency situations to develop a dynamic and adaptive computerized model of the DPA to enhance medical teaching and learning.

Keywords: emergency simulation, practice environment, cognitive task representation, adaptive system, case-based learning.

1 Introduction

Designing an ITS to support teaching and learning in emergency medicine requires special consideration of the high-pressure nature of the live clinical environment [1]. Low-fidelity simulations that replicate some aspects of the clinical environment have been shown to improve diagnostic accuracy and develop clinical skills [2]. Systems that combine low-fidelity simulations with intelligent tutoring techniques have been developed to train medical personnel to care for patients during heart attacks [3] and to develop cardiopulmonary resuscitation skills [4].

In this paper, we present the design of an online version of a simulation-based activity for medical education called the Deteriorating Patient Activity (DPA) [5]. The objective of the DPA is to help medical students learn how to apply a framework to stabilize a patient with symptoms and vital signs that worsen as the simulation continues. In a normal DPA activity, a medical expert plays the role of the deteriorating patient and simulates the positive/negative evolution of his status according to medical actions taken by students. Furthermore, if no action is taken, the patient naturally deteriorates after a certain period of time.

The DPA is a safe and easily controlled environment that allows for repetitive practice with feedback and supports a variety of valid clinical situations with varying levels of difficulty. The online DPA (ODPA) is an addition to the DPA-based instructional approach that aims at allowing students to perform the activity at their own pace without the need for an available human instructor. The idea is to address

simpler cases with the ODPAs and potentially to obtain remote and asynchronous debriefing by a human expert. More complex cases can thus be explored during classic DPA sessions where a human instructor is available. The instructor can also use trace analyses of students' actions in the ODPAs session to inform his or her teaching. Finally, the ODPAs can also be used as a supplemental practice environment to reinforce learning during a classic DPA session.

2 Implementation

The ODPAs application is an online-loadable, platform-independent player for ODPAs cases. Medical students can access the application from any computer connected to the Internet. When loaded, the application retrieves the learner's student model and initializes the *deteriorating patient model* using a remote XML file that matches the case selected by the learner. This file describes the pathways in which the patient state can evolve and provides information to the application to load appropriate *multimedia resources* and adapt the *graphical user interface* accordingly. All pertinent information (learner actions as well as deteriorating patient reaction or natural evolution) is logged and is transmitted to the medical instructor by email. The *remote student model database* is also updated for use in future activities.

Fig. 1 presents the Graphical User Interface (GUI) of the ODPAs.

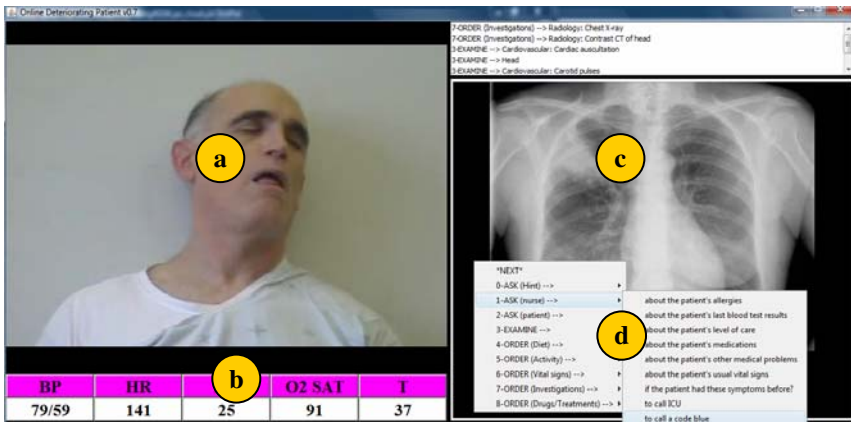


Fig. 1. Sample screen shot of the Graphical User Interface

The GUI is organized using three panels. The main panel (a) displays a video expressing the patient's current state; the second panel (b) displays vital signs and their values, determined by the case creator, that evolve in a time-realistic fashion and the final panel (c) displays additional information that the learner requests, such as laboratory test results, x-rays or expert consultations. Clicking on the right mouse button in the GUI displays a popup menu with a list of general medical actions that the user can perform (d).

The ODPAs is an event-based and time-constrained system. Events can either occur at a certain moment (for instance to generate an automatic state deterioration after a

certain amount of time) or be the result of a user action. All actions occurring in the GUI as well as the patient's state evolution are logged and sent to the medical instructor by email, who can then use such information to evaluate the performance of his/her students and inform the design of upcoming educational activities.

The solution we chose for addressing the open-ended problem of modeling a patient's deterioration can be summarized as a two-step algorithm:

- a) Elicit a limited number of basic deterioration states. A definition of a basic deterioration state includes information about multimedia resources to be displayed by the interface when the patient is in this state, and vital sign values that the patient shall evolve towards.
- b) Determine a set of rules explaining how the patient reacts to user actions depending on the state when the action occurs. State-transition links are the most frequent kind of rules but more complex ones can also be elicited for dynamically modifying the predefined model (i.e. disabling/creating transition links, or provoking the dynamic update of the current state by modifying a limited number of parameters) after certain events or combinations of events occurred.

3 Discussion and Conclusions

The ODPa provides medical students with supplemental clinical practice opportunities and medical instructors with recorded trace data of student actions, which can be used to examine student misconceptions and adapt instruction. Even though the ODPa is a low-fidelity educational simulation, it tries to encourage immersion through videos filmed from the learner's point of view, and by providing realism in terms of dynamic changes to vital signs in terms of speed, and its corresponding display of patient actions that appear in the video. For example, the patient in the video deteriorates at the same rate as the student actions. Furthermore, the ODPa appears to induce strong affect in test users such as fear and anxiety as the patient deteriorates. In future work, the importance of immersiveness in achieving stated learning objectives will be assessed as well as the strength of induced affective reactions and their impact on decision making.

References

1. Lajoie, S.P., Wiseman, J., Poitras, E., Cruz-Panesso, I.: On-line based learning environment for fostering physicians' clinical skills necessary to early recognition of a deteriorating patient. Paper presented at the 2nd Annual Conference on What Really Works in Technology-Enhanced Health Education, Oshawa, Canada (2009)
2. De Giovanni, D., Roberts, T., Norman, G.: Relative effectiveness of high-versus low-fidelity simulation in learning heart sounds. *Med. Educ.* 43(7), 661–668 (2009)
3. Eliot, C., Woolf, B.P.: Iterative development and validation of a simulation-based medical tutor. In: Lesgold, A.M., Frasson, C., Gauthier, G. (eds.) *ITS 1996*. LNCS, vol. 1086, pp. 540–549. Springer, Heidelberg (1996)
4. Romero, C., Ventura, S., Gibaja, E.L., Hervás, C., Romero, F.: Web-based adaptive training simulator system for cardiac life support. *Artif. Intell. Med.* 38, 67–78 (2006)
5. Wiseman, J., Snell, L.: The deteriorating patient: A realistic but "low-tech" simulation of emergency decision-making. *Clin. Teach.* 5, 93–97 (2008)

DynaLearn: Architecture and Approach for Investigating Conceptual System Knowledge Acquisition

Bert Bredeweg¹, Jochem Liem¹, Floris Linnebank¹, René Bühling², Michael Wißner², Jorge Gracia del Río³, Paulo Salles⁴, Wouter Beek¹, and Asunción Gómez Pérez³

¹ University of Amsterdam, Informatics Institute, Amsterdam, Netherlands
{B.Bredeweg, J.Liem, F.E.Linnebank, W.G.J.Beek}@uva.nl

² University of Augsburg, Multimedia Concepts and Applications, Augsburg, Germany
{buehling, wissner}@informatik.uni-augsburg.de

³ Universidad Politécnica de Madrid, Ontology Engineering Group, Madrid, Spain
{jgracia, asun}@fi.upm.es

⁴ University of Brasília, Institute of Biological Sciences, Brasília, Brazil
psalles@unb.br

Abstract. DynaLearn is an Interactive Learning Environment that facilitates a constructive approach to developing a *conceptual* understanding of how systems work. The software can be put in different interactive modes facilitating alternative learning experiences, and as such provides a toolkit for educational research.

Keywords: Qualitative reasoning, Conceptual knowledge, ILE architecture.

1 Introduction

DynaLearn allows learners to acquire conceptual knowledge by constructing and simulating computer-based qualitative models of how systems behave [4]. DynaLearn is based on Garp3 [2] and uses diagrammatic representations for learners to express their ideas. The environment is equipped with components capable of generating knowledge-based feedback, and virtual characters implementing the communicative interaction with learners (see Fig 1). DynaLearn is applied and evaluated in the context of environmental science.

2 Conceptual Knowledge and Use-Levels

Six use-levels have been realized In DynaLearn (see [3] for full description). **Concept map** is a graphical representation (entity-relation graph) that consists of two primitives: nodes (concepts) and arcs (relationships between concepts). A simple version of such a workspace is available in the DynaLearn software. **Basic causal model** focuses on quantities, how they change and how this change causes other quantities to change. Simulation means calculating for each quantity one of the following options: decrease, steady, or increase. Augmented with a teachable agent this use-level closely

relates to Betty's Brain [1]. **Basic causal model with state-graph** augments the previous level with the notion of quantity space. This has a significant impact on the simulation results (because quantities can now change values) and necessarily introduces concepts such as state-graph, behavior path, and value history. **Causal differentiation** refines the notions of causality. Processes are introduced requiring a differentiation between influences (I) and proportionalities (P). **Conditional knowledge**. Some facts only happen when certain conditions are satisfied (e.g. an evaporation process). This use-level introduces the possibility to specify conditions under which a specific set of details holds. The use-level **Generic and reusable knowledge** reflects Garp3 in its current status. The main difference with the other use-levels is the focus on 're-usable' knowledge.

When used in educational practice, use-levels can be used individually to focus on a particular phenomenon, or in a sequence to gradually refine someone's understanding of a phenomenon.

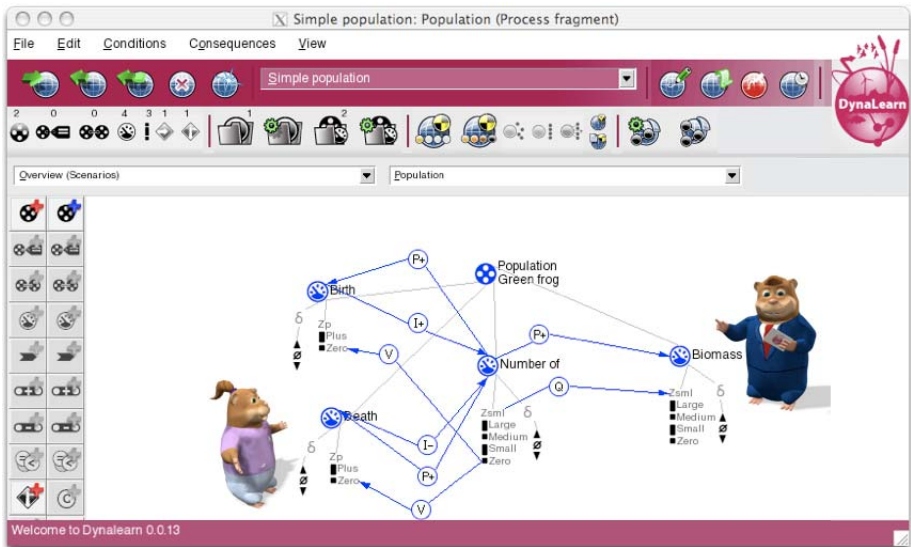


Fig. 1. Shown is a DynaLearn workspace with a diagrammatic expression and two interacting virtual characters, the Student / Learning companion (LHS) and the Quizmaster (RHS)

3 Knowledge-Based Feedback and Virtual Characters

One of the innovative features of DynaLearn is that a QR model created by a learner can be compared to QR models created by other learners and/or experts in order to automatically provide feedback and recommendations to that learner. This is made possible by converting the QR models into 'ontological' models. This conversion is performed in two main steps: (1) The QR model is automatically translated into the OWL language, identifying and extracting its relevant concepts and relationships, and defining them as ontology terms; and (2) a semiautomatic grounding process that

establishes explicit links between these ontology terms and other terms coming from external ontological models and background ontologies. In fact, due to the grounding process, the models created by learners can be related to external knowledge sources enabling the reuse of well-defined vocabularies as well as the inference of new knowledge not asserted in the learner's QR model explicitly.

In DynaLearn the communicative interaction is mediated by a set of virtual characters. Virtual characters lead to an increased sense of ease and comfort, and are expected to have motivating effects on learners [5]. In DynaLearn the characters become active following requests by learners, who have the initiative and control. When activated, the characters react to the diagrammatic expressions created by learners. What content the characters will communicate depends on role they have, and is further fueled by the knowledge-based feedback mechanisms. Thus far in DynaLearn we have established characters following the metaphor of a virtual classroom: student, teacher and quizmaster.

4 Concluding Remarks

The DynaLearn project is an ongoing activity. Currently the following components have been realized: the conceptual modeling environment (including the use-levels), grounding and a simple version of the quality feedback, and the teachable agent and quizmaster. Ongoing research addresses the remaining knowledge-based feedback, and the development of an integrated coherent dialogue (currently each character has its own interaction schema with the learner). Considerable effort will be put in classroom evaluation of the different modes of interaction. Particularly, blending in with ongoing classroom learning activities such that undesired disturbances are minimized as much as possible, while the positive impact and learning enhancements caused by the DynaLearn innovation are maximized.

Acknowledgements. The work presented in this paper is co-funded by the EC within the 7th FP, Project no. 231526, and Website: <http://www.DynaLearn.eu>. Thanks to Anders Bouwer, Richard Noble, Andreas Zitek, Yordan Uzunov and Ian Cowx.

References

1. Biswas, G., Schwartz, D., Leelawong, K., Vye, N.: TAG-V.: Learning by Teaching: A New Agent Paradigm for Educational Software. *Applied Artificial Intelligence, Special Issue on Educational Agents* 19(3), 363–392 (2005)
2. Bredeweg, B., Linnebank, F., Bouwer, A., Liem, J.: Garp3 — Workbench for Qualitative Modelling and Simulation. *Ecological Informatics* 4(5–6), 263–281 (2009)
3. André, E., Bee, N., Bühling, R., Gómez-Pérez, J.M., Häring, M., Liem, J., Linnebank, F., Thanh Tu Nguyen, B., Trna, M., Wißner, M.: Technical design and architecture. In: Bredeweg, B. (ed.) *DynaLearn, EC FP7 STREP project 231526, Deliverable D2.1* (2009)
4. Forbus, K.D.: Qualitative Modeling. In: van Harmelen, F., Lifschitz, V., Porter, B. (eds.) *Handbook of Knowledge Representation*, vol. 3, pp. 361–393 (2008)
5. Thomas, F., Johnston, O.: *The Illusion of Life: Disney Animation*. Abbeville Press, New York (1981)

Interfaces for Inspectable Learner Models

Susan Bull, Andrew Mabbott, Rasyidi Johan,
Matthew Johnson, Kris Lee-Shim, and Tim Lloyd

Electronic, Electrical and Computer Engineering, University of Birmingham, UK
{s.bull}@bham.ac.uk

Abstract. Inspectable (open) learner models have been in use for some time now. We present views of the learner model that have become more common, such as skill meters and concept maps; and introduce developments in less common interfaces for open learner models including the use of animation, audio and haptic feedback, and user-constructed learner model views.

Keywords: Open learner models, presentation of learner models.

1 Introduction

Inspectable open learner models (OLM) are learner models that can be viewed by the user to promote reflection and planning. Models have been opened in a range of domains, e.g. programming [1], maths [2], biology [3], language [4], text editor [5]; or may be domain independent [6],[7]. Learning gains have been observed [4],[8]. OLMs are used in individual [5] or group [7] learning; to promote collaborative or competitive interactions [9]; at various levels (school [3], college [4], university [6]).

2 Inspectable Learner Model Interfaces

Skill meters have become widespread ([1],[6],[8],[10]), illustrated on the top left of Fig. 1 [6]. Of more detailed model views, concept maps (top centre of Fig. 1 [11]) are the most common. Other structured views include trees [5],[11], conceptual graphs [12], Bayesian models [3]. The top right shows knowledge by colour, also giving haptic feedback using a force-feedback device (indicating knowledge by hard/softness of 'knowledge spheres' - a haptic form of skill meters). Audio can be used, and may be particularly applicable to language and music. Fig. 1 gives an example for language (speech sounds), which also uses images and text descriptions of mouth position (second row); and music (chords and spoken text), which also uses written text and music notation (third row left). The third row right of Fig. 1 portrays animations of concepts and misconceptions in chemistry (also implemented in programming). The bottom of Fig. 1 shows examples of user-constructed model views, where learners structure the display as suits their preferences. In some OLMs, users can directly compare their understanding to domain concepts, to help them recognise difficulties or gaps in their knowledge for themselves (see the language, music, chemistry examples in Fig. 1).

3 Summary and Conclusions

Presentation of an OLM may depend on the aims or goals of the OLM, and may be more or less applicable in a domain. Nevertheless, the extent of exploration of the potential for OLMs, and the fact that they are used in real learning settings, suggest an appropriate form of OLM could be found in many contexts. Furthermore, learners appear to trust a range of OLM views. If only an overview is required, widespread deployment of skill meters suggests they will be used by students. Structured model presentations have also been deployed, indicating that concepts and conceptual relationships can be shown. Less common interfaces to the learner model were introduced, e.g. user-created structures; OLMs with domain-specific representations (in music and language), including audio that may also be more generally applicable; animation of concepts and misconceptions (implemented in chemistry and programming); and haptic feedback. The latter may be harder to deploy given the equipment required. However, it could be usefully applied in contexts benefiting from haptic feedback during a task (e.g. engineering structures, medical/surgery).

References

1. Weber, G., Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-Based Instruction. *IJAIED* 12, 351–384 (2001)
2. VanLabeke, N., Brna, P., Morales, R.: Opening Up the Interpretation Process in an Open Learner Model. *IJAIED* 17(3), 305–338 (2007)
3. Zapata-Rivera, J.D., Greer, J.E.: Interacting with Inspectable Bayesian Models. *IJAIED* 14, 127–163 (2004)
4. Shahrour, G., Bull, S.: Interaction Preferences and Learning in an Inspectable Learner Model for Language. In: *AIED 2009*, pp. 659–661. IOS Press, Amsterdam (2009)
5. Kay, J.: The UM Toolkit for Cooperative User Modeling. *UMUAI* 4(3), 149–196 (1995)
6. Bull, S., Quigley, S., Mabbott, A.: Computer-Based Formative Assessment to Promote Reflection and Learner Autonomy. *Engineering Education* 1(1), 8–18 (2006)
7. Rueda, U., Larrañaga, M., Ferrero, B., Arruarte, A., Elorriaga, J.A.: Study of Graphical Issues in a Tool for Dynamically Visualising Student Models, In: *LeMoRe Workshop, AIED 2009*, pp. 268–277 (2003)
8. Mitrovic, A., Martin, B.: Evaluating the Effect of Open Student Models on Self-Assessment. *IJAIED* 17(2), 121–144 (2007)
9. Bull, S., Britland, M.: Group Interaction Prompted by a Simple Assessed Open Learner Model that can be Optionally Released to Peers. In: *PING Workshop, User Modeling (2007)*
10. Corbett, A.T., Anderson, J.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *UMUAI* 4, 253–278 (1995)
11. Mabbott, A., Bull, S.: Alternative Views on Knowledge: Presentation of Open Learner Models. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) *ITS 2004*. LNCS, vol. 3220, pp. 689–698. Springer, Heidelberg (2004)
12. Dimitrova, V.: StyLE-OLM: Interactive Open Learner Modelling. *IJAIED* 13, 35–78 (2003)

Conceptual Personalization Technology: Promoting Effective Self-directed, Online Learning

Kirsten R. Butcher¹, Tamara Sumner², Keith Maull², and Ifeyinwa Okoye²

¹ University of Utah, Department of Educational Psychology, 1705 E Campus Center Drive, MBH 327, Salt Lake City, UT, 84112, USA

`Kirsten.Butcher@utah.edu`

² University of Colorado at Boulder, Institute of Cognitive Science, UCB 344, Muenzinger Psych Building D414, Boulder, CO, 80309, USA

`{Sumner, Keith.Maull, Ifeyinwa.Okoye}@colorado.edu`

Abstract. This paper presents an empirical learning study using a prototype system designed to provide fully automatic, domain-independent conceptual personalization algorithms. The prototype system, the *Customized Learning Service for Concept Knowledge (CLICK)*, was implemented as an adaptive essay writing environment in a scientific domain. Results demonstrate that conceptual personalization promotes deep metacognitive strategies during online learning, and these strategies correlate with deep domain understanding.

Keywords: Personalization; Online Learning; Metacognition; Comprehension.

1 Introduction

Increasingly, learning opportunities are moving away from structured classroom environments toward unstructured, self-directed learning using digital resources. These unstructured learning experiences are characterized by student-driven decisions about what online resources to use, and when and how to use them. However, students typically lack the prior knowledge and metacognitive skills necessary to find appropriate online resources and to effectively learn from them [1]. A critical challenge for cyber-learning is developing tools that can support students in deploying strategies and processes necessary to integrate and synthesize multiple online sources [2]. Previous research using human tutors has shown that adaptive support during self-directed learning can facilitate effective metacognitive strategies and promote mental model development [3]. Intelligent tutoring systems also have shown great promise in using adaptive systems to support learning [4], but there is a critical need for tools that can be easily scaled to large populations and utilized by diverse individuals in a variety of self-directed and unstructured learning environments.

We describe the results of a learning study conducted to test the effects of a prototype cognitive personalization system: the *Customized Learning Service for Concept Knowledge* (referred to as *CLICK*). Cognitive personalization tools support learning by matching students with sets of online resources that they, as individual learners with a unique profile of prior knowledge and misunderstandings, need in order to develop a more complete and coherent understanding of the topic at hand.

2 The CLICK System

A brief overview of CLICK is provided here; details on the technical and computational underpinnings of the CLICK system are published elsewhere [5, 6]. CLICK consists of three major algorithms. The first algorithm produces knowledge map representations of current student knowledge and idealized domain knowledge. A knowledge map is a concept map where the nodes contain complex knowledge propositions (phrases, sentences, etc.). CLICK operates on student work products – in this case, essays – to create student knowledge maps. CLICK also generates a domain knowledge map, which is drawn *automatically* by extracting key science concepts and their relationships from a set of age- and domain-relevant web-based learning resources. The domain knowledge map depicts what an informed person of the target age group might be expected to know about a scientific topic. The second CLICK algorithm compares the student knowledge map and the domain knowledge map using graph-theoretic techniques to diagnose current student understanding [5]. CLICK diagnoses three types of conceptual problems: incorrect statements (students contradict concepts in the domain map), incomplete understanding (students provide a partial description or fail to mention a concept), and fragmented knowledge (students fail to connect two related science concepts). The third algorithm selects specific, interactive digital library resources to address each identified knowledge problem [6]. In the current prototype, a personalized recommendation engine draws resources from a test bed collection containing 796 age and topic-appropriate learning resources drawn from the Digital Library for Earth System Education (www.DLESE.org).

3 Learning Study: Description and Results

Thirty undergraduate students participated in the two-session study. In session 1, students' prior knowledge was assessed using a true/false test (targeting factual knowledge) and a short answer test (targeting domain understanding). Each student also wrote an initial essay that served as the basis for CLICK assessment. In session 2, students were randomly assigned to one of two conditions before revising their essays: 1) a Digital Library condition, in which students received CLICK-generated essay feedback with information on revision strategies, and used a digital library interface to search over the DLESE resource test bed; 2) a CLICK personalization condition, in which students received CLICK-generated feedback accompanied by a metacognitive prompt (e.g., "What makes you say that?"), as well as CLICK's automatically-chosen, personalized resource recommendations (drawn from the DLESE test bed) for each identified knowledge problem. The key difference between the conditions was whether or not students received *personalized resource recommendations* tied to identified knowledge problems. Following revision, students completed a metacognitive questionnaire (in which they analyzed essay feedback and explained their essay revision strategies) and completed knowledge posttests.

Results demonstrated that students in the CLICK condition were significantly *less* likely to report preserving the same ideas in their revisions ($F_{(1, 27)} = 12.5, p = .002$) and significantly *more* likely to report seeking online resources to augment or revise the scientific content of their essays ($F_{(1, 27)} = 5.6, p = .026$). CLICK students were

more likely to report using deep metacognitive processes (focusing on revising and integrating ideas) as they analyzed and responded to feedback [1]. Students in the CLICK condition reported deep revision processes more often than control condition students ($F_{(1, 26)} = 4.9, p < .04$) and tended to report fewer shallow revision processes ($F_{(1, 26)} = 3.1, p = .09$; see Table 1). Though not statistically significant ($F_s < 1.4, p_s > .25$), results for self-reported analysis of essay feedback followed the same pattern.

Table 1. Self-Reported Metacognitive Processes (% of Total): Means (Standard Deviations)

Process	Examples	Condition	<i>M (SD)</i>
Deep Analysis	Missing/incorrect content	Digital Library	67% (34%)
		CLICK	80% (27%)
Shallow Analysis	Poor grammar/spelling	Digital Library	33% (34%)
		CLICK	20% (27%)
Deep Revision Strategies	Add/explain content	Digital Library	51% (28%)
		CLICK	62% (21%)
Shallow Revision Strategies	Reword/delete sentence	Digital Library	49% (28%)
		CLICK	38% (21%)

Correlational analyses showed that improvement on the test of domain understanding (the short answer test) was significantly ($p < .05$) and positively correlated to deep analysis ($r = .37$) and to deep strategies ($r = .43$), but negatively (not significantly) correlated to shallow processes. The opposite pattern was true for the factual knowledge test (the true/false test), suggesting that conceptual personalization can support the development of deeper domain understanding, by promoting effective metacognitive strategies during self-directed, online learning.

References

1. Butcher, K.R., Sumner, T.: Self-directed learning and the sensemaking paradox. *Human Computer Interaction* (in press)
2. Lynch, C.: Digital libraries, learning communities, and open education. In: Iiyoshi, T., Kumar, M.S.V. (eds.) *Opening up education*, MIT Press, Cambridge (2008)
3. Azevedo, R., Cromley, J.G., Seibert, D.: Does adaptive scaffolding facilitate students' ability to regulate their learning with hypermedia? *Contemporary Educational Psychology* 29, 344–370 (2004)
4. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education* 8, 30–43 (1997)
5. de la Chica, S., Ahmad, F., Sumner, T., Martin, J.H., Butcher, K.R.: Computational foundations for personalizing instruction with digital libraries. *International Journal of Digital Libraries Special Issue on Educational Digital Libraries*, 3–18 (2008)
6. Gu, Q., de la Chica, S., Ahmad, F., Khan, H., Sumner, T., Martin, J.H., Butcher, K.R.: Personalizing the selection of digital library resources to support intentional learning. In: Christensen-Dalsgaard, B., Castelli, D., Ammitzbøll Jurik, B., Lippincott, J. (eds.) *ECDL 2008*. LNCS, vol. 5173, pp. 244–255. Springer, Heidelberg (2008)

Learning to Identify Students' Relevant and Irrelevant Questions in a Micro-blogging Supported Classroom

Suleyman Cetintas¹, Luo Si¹, Sugato Chakravarty², Hans Aagard³, and Kyle Bowen³

¹ Department of Computer Sciences
Purdue University, West Lafayette, IN, 47907, USA
{scetinta, lsi}@cs.purdue.edu

² Department of Consumer Sciences and Retailing,
Purdue University, West Lafayette, IN, 47907, USA
sugato@purdue.edu

³ Rosen Center for Advanced Computing
Purdue University, West Lafayette, IN, 47907, USA
{hans, kbowen}@purdue.edu

Abstract. This paper proposes a novel application of text categorization for two types questions asked in a micro-blogging supported classroom, namely relevant and irrelevant questions. Empirical results and analysis show that utilizing the correlation between questions and available lecture materials in a lecture along with personalization and question text leads to significantly higher categorization accuracy than i) using personalization along with question text and ii) using question text alone.

1 Introduction

Micro-blogging, a Web 2.0 technology, is a type of blogging that lets the users post short text messages to their community in real time. Recently, micro-blogging tools have been used in classroom environments as a communication tool between students with the instructor [2]. An important issue with large micro-blogging supported classrooms is that the number of questions/comments an instructor receives from the students can be many more than what s/he can answer in a limited time.

To the best of our knowledge, there is no prior research on the categorization of relevant and irrelevant micro-blogging messages or questions in classroom environments. Prior work on teacher agents has a question ranking capability that utilizes the questions text as well as a personalized approach to differentiate between the relevant and irrelevant questions [5]; but ignore the educational materials that are available in most classrooms.

This paper proposes a text categorization approach that can automatically identify relevant and irrelevant questions asked in a lecture by utilizing multiple types of evidence including question text, personalization, correlation between questions and lecture materials. We show that i) utilizing personalization along with question text is more effective than using question text alone, ii) utilizing the correlation between question and available lecture materials improve the categorization accuracy and iii) tf-idf weighting scheme is more effective than okapi while estimating the correlation between questions and available lecture materials.

2 Data

Data collected from a micro-blogging supported personal finance class (a 300 level u.g. course) during Fall 2009 has been used in this work. The study was conducted in a large classroom with 243 students during 24 lectures (each of them 50 minutes long). Data from first 4 lectures are used for training and the remaining 20 lectures are used for testing. Each lecture has an average of 26.9 relevant questions with a standard deviation of 9.4 and 10.8 irrelevant questions with a standard deviation of 8.5 (i.e. totally 645 relevant and 260 irrelevant questions). We employ two human annotators (the first author and an expert in finance) and ask them to annotate each question as either being relevant or irrelevant. The annotators reach a Kappa of 0.868 on 162 questions (i.e. questions of first 4 lectures) and therefore the rest of the data was annotated by the first annotator only. Every lecture has 1 publicly available presentation file relevant to the lecture (which are used as the available relevant lecture materials) and the course has a syllabus file that discusses about course policy, exams, projects, etc. (which is used as the available irrelevant lecture material).

3 Techniques: Support Vector Machine and Cosine Similarity

Micro-blogging questions/messages are in textual format; therefore we use the widely used Support Vector Machines (with a linear kernel) as our text classifier [4]. The categorization threshold of each SVM classifier is learned by 2-fold cross validation in the training phase (i.e. 2 of the 4 training lectures for each fold).

Cosine similarity is a measure of similarity between two vectors by calculating the cosine of the angle between them which is commonly used in text mining to compare text documents. In this work, the similarity scores between questions and lecture materials are calculated as a measure of the correlation by the common Cosine measure [1]. We use and compare the two common weighting schemes: tf-idf [1] that uses term frequency and inverse document frequency (i.e. favoring discriminative terms that only reside in a small number of documents) and okapi [3] that additionally considers document sizes by favoring shorter but relevant documents.

4 Performances of Several Modeling Approaches

SVM_TermsOnly & SVM_TermsPers: Using individual features (i.e. terms) of questions along with personalization to select the best questions to respond to has been shown to be a useful approach in a recent prior work [5]. In this work, we use two features for personalization: i) percentage of relevant questions asked by a student and ii) percentage of irrelevant questions asked by a student. An SVM classifier that only uses the terms of questions is used along with another SVM classifier that uses personalization along with the terms. The two baseline classifiers will be referred as SVM_TermsOnly and SVM_TermsPers respectively.

SVM_TermsPersLMSim: In a lecture, it is intuitive that most relevant questions asked in a class will be related with the lecture being covered in class. This modeling approach makes use of this fact and adds 3 new features about the correlation between

Table 1. Results of the SVM_TermsOnly, SVM_TermsPers and SVM_TermsPersLMSim classifiers in comparison to each other for two main configurations (while correlations between questions and available lecture materials are calculated) i) with tf-idf and ii) okapi is used as weighting schemes. The performance is evaluated with the F_1 measure [1] and “ F_1 (precision, recall)” triplets are reported for each configuration of each classifier.

Methods	Tf-Idf	Okapi
SVM_TermsOnly	0.7252 (0.8797, 0.6169)	
SVM_TermsPers	0.7830 (0.8675, 0.7135)	
SVM_TermsPersLMSim	0.8627 (0.9070, 0.8225)	0.7948 (0.8514, 0.7452)

a question and available lecture materials (that may be relevant or irrelevant) to the set of baseline (i.e. terms and personalization) features. Particularly the added features are the cosine similarity score i) between a question and available relevant lecture material(s) of the current lecture, ii) between a question and all available relevant lecture materials of that course (i.e. in this work, sum of the similarity scores of top 3 most similar relevant materials are used), iii) between a question and all available non-relevant lecture materials (i.e., in this work, the similarity score with the only irrelevant material is used: if there are more irrelevant materials, the approach in (ii) can be used).

It can be seen in Table 1 that SVM_TermsPers classifier significantly (with p-value of less than 0.01, for paired t-test) outperforms the SVM_TermsOnly classifier. This shows that utilizing personalization along with terms is a better approach than using only terms of questions and this is consistent with prior research [5]. SVM_TermsPersLMSim classifier is also shown to significantly (i.e. with p-value much less than 0.01) outperform SVM_TermsPers and SVM_TermsOnly classifiers. Utilizing the correlations among questions and available lecture materials along with terms and personalization is a better approach than using only personalization and terms of questions. To assess the similarity between questions and available lecture materials, two common weighting schemes are used with the cosine similarity measure. It can be seen that tf-idf weighting scheme significantly outperforms (with p-value less than 0.01) okapi weighting scheme.

5 Conclusions

This paper proposes a novel application of text categorization to identify relevant and irrelevant micro-blogging questions asked in a classroom. Several modeling approaches and weighting configurations are studied for this application. Empirical results show that utilizing the correlation among questions and available lecture materials along with personalization and question text significantly outperforms i) using personalization and question text and ii) using question text only. Furthermore, it is found to be significantly more effective to use tf-idf weighting scheme rather than okapi while calculating the correlations among questions and available lecture materials.

Acknowledgements. This research was partially supported by the following grants IIS-0749462, IIS-0746830 and STC-0939370. Any opinions, conclusions expressed in this paper are the authors', and do not necessarily reflect those of the sponsor.

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press Series/Addison Wesley (1999)
2. Grossek, G., Holotescu, C.: Can we use Twitter for educational activities? In: Proceedings of the 4th Int. Scientific Conf., eLearning and Software for Education (2008)
3. Robertson, S., Walker, S., Beaulieu, S., Gull, A., Lau, M.: Okapi at TREC. In: Proceedings of 1st TREC Conference (1992)
4. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1–47 (2002)
5. Soh, L., Khandaker, N., Jiang, H.: I-MINDS: A Multiagent System for Intelligent Computer-Supported Collaborative Learning and Classroom Management. Int. J. Artif. Intell. Ed. 18(2), 119–151 (2008)

Using Emotional Coping Strategies in Intelligent Tutoring Systems

Soumaya Chaffar and Claude Frasson

Département d'informatique et de recherche opérationnelle
Université de Montréal C.P. 6128, Succ. Centre-ville
Montréal, Québec Canada H3C 3J7
{chaffars, frasson}@iro.umontreal.ca

Abstract. Successful individuals appear to have developed efficient method for reducing and coping with stress and anxiety. There is some evidence that emotions of this kind are correlated negatively with performance. Furthermore, it is important that new Intelligent Tutoring Systems (ITS) involve emotional coping strategies in order to improve the learner's performance. In this paper, we show that problem-focused coping strategies are more efficient than emotion-focused ones for inducing positive emotions.

Keywords: ITS, tutorial actions, emotional coping strategies.

1 Introduction

A number of studies showed that user interactions with computers are close to human relationships. For example, Reeves and Nass (1996) argue that users treat computers like real people [7]. In addition, Klein and colleagues (2002) concluded that computers are strongly able to regulate negative emotions, even if they are the sources of these emotions [5]. Emotion regulation is defined by Gross (1998) as the ability to reduce the high intensity of a given emotion (negative or positive) and to change it [3]. Coping differs from emotion regulation by focusing only on negative emotions aiming to reduce negative emotional experiences [1].

In this research work we attempt to discuss the benefit of using emotional coping strategies in ITS. We try to compare the effect of emotion-focused strategies with the problem-focused one on the learner's emotional state.

2 Experiment and Discussion

Three kinds of emotion-focused and two kinds of problem-focused strategies have been used in this experiment (see Table 1) depending on stressful situations facing the learner. During his interaction with an ITS, a learner might be subject to many stressful experiences. For instance, evaluation tests were always regarded as negative emotional experiences for learners. They might feel various negative emotions dominated by anxiety, fear, etc. Once obtaining their evaluation mark, learners might feel

negative or positive emotions. In addition, a non-suited course presentation for learning styles could affect learners’ emotional states. For example, presenting Canadian history in text format to auditive students (those for whom the sense of hearing is the strongest perceptual preference) could generate boredom. Similarly, a very difficult course for learners could produce anxiety. However, a very easy one could induce boredom. It is therefore important to act following an emotional situation to alleviate possible negative emotions by adapting instruction to learners’ needs or reassuring them after receiving their marks.

In this research study, a total of twenty-nine graduate students (17 male/12 female, aged 22-40 years) in computer science were experienced in two emotional situations: misunderstanding of a data structure course and obtaining marks in a data structure evaluation test. These two situations might significantly affect students’ emotional state and their performances [6].

Table 1. Emotional coping strategies used in the experiment

Initial actions	Final actions
Definition	Encouragement
Example	Recommendation
Encouragement	Congratulation

} **(problem-focused)** } **(emotion-focused)**

A virtual tutor was used in this experiment attempting to influence the learner’s emotional state in the two situations (as shown in the table above): the first one when the learner needs help to understand a sorting algorithm (initial actions) and the second one when the learner obtained his mark after passing an evaluation test (final actions). As some participants randomly experienced several emotional situations depending on their request for help in understanding different sorting algorithms, our sample data is composed of 73 instances.

EMG is the most studied and validated physiological measure for indicating the valence even in the absence of facial expressions [4]. Thus, we have used this signal for analysing in more detail the learner’s emotional changes (see Table 2) after the tutor’s actions [2].

Table 2. ANOVA’s Results of EMG signals for different actions

Actions	Time	Average of peaks	F	P
<i>Definition</i>	Before	1.83	10.10	0.03
	During and after	15.51		
<i>Example</i>	Before	1.544	29.174	0.00
	During and after	7.986		
<i>Encouragement_Comprehension</i>	Before	2.045	3.289	0.088
	During and after	7.375		
<i>Recommendation</i>	Before	2.046	6.036	0.025
	During and after	12.516		
<i>Encouragement_Note</i>	Before	1.763	20.359	0.00
	During and after	6.673		
<i>Congratulation</i>	Before	0.948	4.681	0.162
	During and after	15.432		

The table above showed that the problem-focused strategies have significant positive effects on the variation of the EMG signals. However, the results of the emotion-focused ones showed that the *Encouragement_Comprehension* has no effect on the learner's emotional state. So, the learner needs for helping him to understand the course (definition, example) rather than encouraging him when he didn't understand the course. During the comprehension activity, we can conclude that it is recommended to use a problem-focused strategy in order to induce positive emotions in the learner. This strategy attempts to change the problem that is causing the emotional reaction by explaining the course using an example or a definition for instance in order to change the learner's misunderstanding state and improve his emotional state.

3 Conclusion

In this paper, we conclude that there are significant positive effects after using emotional coping strategies on participants' emotional state, with 95% of confidence. Thus, we assume that including emotional coping strategies in ITS would be beneficial for learning.

Acknowledgments. We address our thanks to the Fond Québécois pour la Recherche sur la Nature et la Technologie (FQRNT) for supporting this research work.

References

1. Blair, K.A., Denham, S.A., Kochanoff, A., Whipple, B.: Playing it cool: Temperament, emotion regulation and social behavior in preschoolers. *Journal of School Psychology* 42, 419–443 (2004)
2. Chaffar, S., Derbali, L., Frasson, C.: Towards Emotional Regulation in Intelligent Tutoring Systems. In: *AACE World Conference on E-learning in Corporate, Government, Healthcare, & Higher Education: E-LEARN 2009*, Vancouver, Canada (2009)
3. Gross, J.J.: The Emerging Field of Emotion Regulation: An Integrative Review. *Review of General Psychology* 2, 271–299 (1998)
4. Hazlett, R.L., Benedek, J.: Measuring emotional valence to understand the user's experience of software. *International Journal of Human-Computer Studies* 65 (2007)
5. Klein, J., Moon, Y., Picard, R.W.: This computer responds to user frustration: theory, design, and results. *Interacting with Computers* 14, 119–140 (2002)
6. Perry, P., Hechter, F.J., Menec, V.H., Weinberg, L.H.: Enhancing achievement motivation and performance in college students: An attributional retraining perspective. *Research in Higher Education* 34, 687–723 (1993)
7. Reeves, B., Nass, C.: *The media Equation: How people Treat Computers, Television, and New Media Like Real People and Places*. Cambridge University Press, Cambridge (1996)

Showing the Positive Influence of Subliminal Cues on Learner's Performance and Intuition: An ERP Study

Pierre Chalfoun and Claude Frasson

Département d'informatique et de recherche opérationnelle, Université de Montréal
2920 Chemin de la Tour, Montréal, Québec
{chalfoun, frasson}@iro.umontreal.ca

Abstract. This paper presents results from an empirical study conducted with a novel subliminal teaching technique aimed at enhancing learners performance in an ITS. We replicated previous findings with the same technique but in a 2D environment. Non intrusive physiological sensors were used to record affective and cerebral responses. A brain analysis technique called Event-Related Potential (ERP), known to describe and confirm cognitive functions in the brain, provided strong evidence that subliminal cues and miscues were cognitively processed even though reported as not seen. The obtained results showed that only subliminal cues, not miscues, could significantly increase learner performance and intuition in a logic-based problem solving task.

Keywords: unconscious cognition, subliminal priming, ERP, intelligent tutoring systems, intuition.

1 Introduction

The aim of intelligent Tutoring Systems (ITS) has been to properly adapt learning material to the learner. However, a major component of learning and decision making when solving problems has been neglected: human intuition. In fact, a large body of work in neuroscience and other fields has put forth compelling evidence that learning simple to complex information can be done without perception or complete awareness to the task at hand [1, 2]. The existence of perceptual learning without perception has not only been proven [3], but replicated in a recent study two years ago [4]. In the present study, we hope to prove the usefulness of our novel subliminal learning technique by (1) replicating our previous findings, (2) presenting data to assess the presence of cerebral *endogenous* processing, that is processing *inside* the brain when subliminal stimuli is projected and (3) showing that positive subliminal cues can enhance the learner's intuition in decision making without his conscious perception.

2 Experiment and Results

The focus of the experiment is to visually teach the construction of an odd magic square of any order without using mathematical operations (see [4] for complete details). EEG data and misdirection cues were added in this experiment with regards to

the last study [4]. The experiment was divided in 5 steps and 3 groups as follows: in **step 1**, a series of neuropsychological tests were administered. Learners then proceeded to **step 2** where the three required tricks to complete a magic square were taught by using either no subliminal cues (control group), subliminal positive cues (answer group) or subliminal misdirection (miscue group). Each stimulus was preceded by a 50 ms pre-mask of random geometrical figures, a 33.33 ms prime (2 frames of a 60Hz SVGA screen) and a 50 ms post-mask of random geometrical figures. We decided to show the learners multiple examples of each trick without explaining how the trick works. It was up to them to *deduce* the inter-workings of each trick. Learners reported how they deduced each trick by choosing between multiple-choice answers (intuition, logic or a mix of both). When all the three tricks were deduced, learners were instructed, in **step 3**, to respond to a series of 13 questions. The first 10 questions, Q1 to Q10 tested their knowledge of each learned trick. The last 3 questions however tested their knowledge of all three tricks combined. Learners reported how they answered each question by choosing between multiple-choice answers (guess, intuition, logical deduction, mainly by intuition or mainly by logical deduction). After answering all the questions, a series of post-tests were administered in **step 4** to test for prime awareness and overall system evaluation. Lastly, **step 5** displayed learner’s performance (completion time and mistakes). A total of 46 healthy volunteers, 23 men and 23 women, took part of the experiment. The mean age was 27 (SD = 3.51). Each received a 10\$ CAN as compensation.

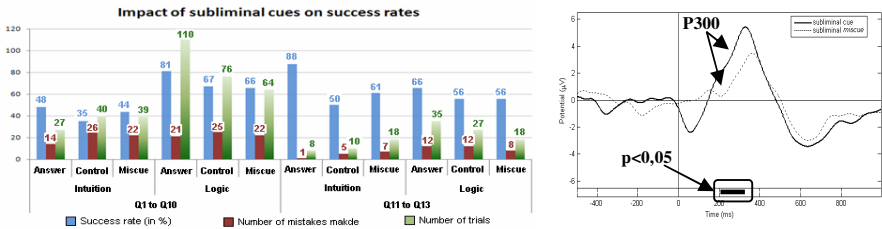


Fig. 1. Subliminal cues can enhance learner’s intuition and evoke a P300

Subliminal cues enhance performance based on intuitive decision making. At first glance, the numbers in fig. 1 seem to indicate a higher success rate for the answer group across all questions. However, a more fine grained analysis revealed a strong significant difference (one way ANOVA, $p=0.007$, $\alpha=0.05$) for the answer group when solving questions based on their intuition. This difference is much more pronounced for the last three questions where the knowledge of all the three tricks was required to answer properly. These questions are supposed to have a 50% success rate if one chooses to answer randomly. We can indeed observe that the success rate for the miscue and control group is approximately 55% whereas the answer group has an astounding rate of 88%. It is also important to note that perception of subliminal cues was assessed at the end of the experiment and none of the participants indicated seeing or noticing anything peculiar about the experiment. Furthermore, a P300 component was found for the subliminal groups (cue and miscue). The P300 component has been of specific interest for cognitive science since it has recently been involved

in tasks where decision making is involved [5]. The results indicate that a P300 component is elicited by subliminal priming. This finding suggests that mental allocation is taking place after the perception of the stimulus, thus the cues taken into consideration at an unconscious level. Since both amplitude and latency differences in P300 components in the past have been linked to reward valence and decision making [6], we can assume that both primes (answers and miscues) are perceived but only the positive cues (answers) carry more ‘salience’ and eventually more ‘pertinent’ information to the learner.

3 Conclusion

We have discussed in this paper the use of subliminal priming for learning. We have showed that our learning technique can be used in different learning environments and that it can lead to very encouraging and promising results. The use of ERP in the analysis of the impact of subliminal primes has revealed that both cues and miscues have been registered by the attention centers of the brain and elicit in the process a P300 component. We have demonstrated that subliminal cues can have an important impact on learning but more specifically on decision making when using one’s intuition. These results seem to indicate that positive subliminal cues can enhance learner’s performance with regards to intuitive decision making when answering questions in a learning session. Further in depth analysis is required in order to examine other sources of information such as response time and emotional manifestations to help establish the optimal cerebral conditions for subliminal learning to occur.

Acknowledgements. We would like to thank the FQRSC for funding this research.

References

1. Strahan, E.J., Spencer, S.J., Zanna, M.P.: Subliminal priming and persuasion: Striking while the iron is hot. *Journal of Experimental Social Psychology* 6 (2002)
2. Watanabe, T., Nanez, J.E., Yuka, S.: Perceptual learning without perception. *Nature* 413 (2001)
3. Kouider, S., Dehaene, S.: Levels of processing during non-conscious perception: a critical review of visual masking. *Philosophical Transactions Of The Royal B Society* 362 (2007)
4. Chalfoun, P., Frasson, C.: Subliminal priming enhances learning in a distant virtual 3D Intelligent Tutoring System. *IEEE Multidisciplinary Engineering Education Magazine: Special Issue on Intelligent Tutoring Systems* 3 (2008)
5. Hajcak, G., Moser, J.S., Holroyd, C.B., Simon, R.F.: It’s worse then you thought: The feedback negativity and violations of reward predictions in gambling tasks. *Psychophysiology* 44 (2007)
6. Wu, Y., Zhou, X.: The P300 and reward valence, magnitude, and expectancy in outcome evaluation. *Brain Research* 1286 (2009)

Exploring the Relationship between Learner EEG Mental Engagement and Affect

Maher Chaouachi and Claude Frasson

Département d'informatique et de recherche opérationnelle
Université de Montréal
C.P 6128, succ. Centre-Ville
Montréal, Québec Canada H3C 3j7
{chaouacm, frasson}@iro.umontreal.ca

Abstract. This paper studies the influence of learner's affective states on the EEG-mental engagement index during a problem solving task. The electrical activity of the human brain, known as electroencephalography or EEG was registered according to an acquisition protocol in a learning environment specifically constructed for emotional elicitation. Data was gathered from 35 healthy subjects using 8 biosensors and two video cameras. The effect of learners' emotional states on the engagement index was analyzed as well as their impact on response time variability.

Keywords: emotion, EEG, mental engagement, ITS, response time.

1 Introduction

During the last decade there has been a growing research interest about student motivation and engagement in intelligent tutoring systems (ITS). Several approaches for engagement tracing and detection have been suggested, ranging from Item response theory models to Bayesian networks [1]. Moreover learning activity is also fundamentally related to emotions. In fact emotions play an important role in creative thinking, inspiration as well as concentration and motivation [2]. Hence, intelligent tutoring systems should adapt their communication and interaction with learners according to changes in the affective dimension as well as their engagement level. Researchers from various scientific communities have made great improvements in methodologies and technologies that give insight into the brain and the learner's physiological activity. Pope, Bogart, and Bartolome at NASA developed an EEG-engagement index based on brainwave band power. They reported a performance improvement in a vigilance task when this index was used as a criterion for switching between manual and automated piloting mode [3]. In this poster we focus on the interaction between learners' engagement level and emotional dimensions and their impact on learners' response time variability.

2 Experimental Design and Results

Thirty-five learners (13 women) with a mean age of 27.2 ± 6.91 years, ranging from 19 to 46 years, took part in the experiment. Learners were asked to respond to three series of ten successive questions while their EEG and physiological signals were recorded. During data acquisition, learners wore an electro-cap and data was recorded from six active sites, four located on the scalp at locations P3, C3, Pz, Fz as defined by the international 10-20 system and referenced to Cz. The EEG data served to compute the engagement index using a ratio of three EEG bands: Theta (4–8 Hz), Alpha (8–13 Hz) and Beta (13–22 Hz). The ratio used was: Beta / (Alpha + Theta). In order to detect affective states, learners were also equipped with blood volume pressure sensor (BVP) and skin conductance sensor (SC). BVP signals were used to derive the heart rate (HR) whereas SC sensors computed galvanic skin response (GSR). We established four quadrants, labeled Q1 to Q4, with regards to signal variations in both HR and GSR according to Lang's 2D affective model (figure 1) [4].

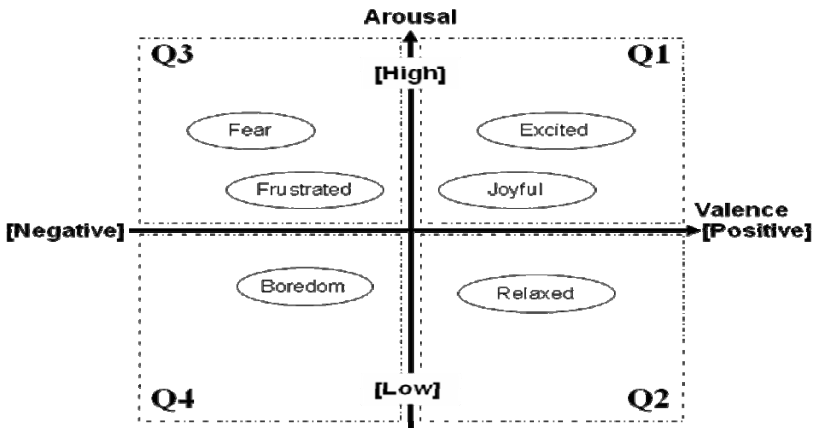


Fig. 1. Lang's 2D affective space labeled by quadrants

An engagement level variable with two states (high and low) was established during each problem solving task (i.e. answering a question). The engagement level was considered *high* if the mean engagement index value was above the learner's engagement baseline value and *low* otherwise. According to mean HR and GSR, one of the four quadrants of the emotional states' space was assigned to the learner during answering each question. A two-way ANOVA performed on response time ($N=1050$) with engagement level and quadrant as grouping factors yielded a significant effect of emotional state on RT variability, $F(3, 1042)=6.638, p < 0.01$. Results indicated that learners took significantly more time resolving a problem when they were on Q2 ($M = 5.926, SD = 0.354$) and Q1 ($M = 5.380, SD = 0.309$) compared to the time spent when they were on Q3 ($M = 4.378, SD = 0.220$) and Q4 ($M = 4.441, SD = 0.222$). The main effect of engagement level on RT variability was non significant, $F(1, 1042)=2.005, p = n.s.$ nevertheless, the interaction effect was significant $F(3, 1042)=49.705, p < 0.05$. Results depicted on Fig. 2, showed that compared to all four

quadrants, the slowest RT was registered for high and low engagement level in Q1 and Q2 respectively (High engagement: $M_{RT} = 6.202$, $SD_{RT} = 0.493$, Low engagement: $M_{RT} = 6.203$, $SD_{RT} = 0.538$). However fastest RT for high and low engagement level was registered in Q4 and Q3 (High engagement: $M_{RT} = 4.305$, $SD_{RT} = 0.279$, Low engagement: $M_{RT} = 3.987$, $SD_{RT} = 0.346$).



Fig. 2. Mean learners' response time according to their engagement levels and emotional states

Results showed also the impact of the arousal dimension on the engagement index. In fact, as shown on Fig. 2, two trends are visible: (1) High arousal subspace: When learner emotional state during a question is Q1 or Q3, his RT is higher when his engagement level is higher compared to when his engagement level is low. (2) Low arousal subspace: When learner emotional state during a question is Q2 or Q4, his RT is higher when his engagement level is low compared to when his engagement level is high.

3 Conclusion

In this poster we studied the relationship between learners' engagement level, emotional states and response time variability. Results showed that learners' response time variability is strongly influenced by the engagement level and emotions.

Acknowledgement. We thank the FQRSC and the CRSNG for funding this work.

References

1. Koedinger, K.R., Corbett, A.T., Ritter, S., Shapiro, L.J.: Carnegie Learning's Cognitive Tutor: Summary research results. Carnegie Learning, Pittsburgh (2000)
2. Guilford, J.P., Höpigner, R.: The Analysis of Intelligence. McGraw-Hill Book Company, New York (1971)
3. Pope, A.T., Bogart, E.H., Bartolome, D.S.: Biocybernetic system evaluation indices of operator engagement in automated task. *Biological psychology* 40(9) (1995)
4. Lang, P.J.: The emotion probe: studies of motivation and attention. *American Psychologist* 50(14) (1995)

MiBoard: Creating a Virtual Environment from a Physical Environment

Kyle Dempsey, G. Tanner Jackson, and Danielle S. McNamara

Department of Psychology, University of Memphis,
38152 Memphis, TN
{kdempsey, gtjacksn, dsmcnamr}@memphis.edu

Abstract. Increasing both user enjoyment and persistence, or engagement, is a challenge in ITS development. The current study investigates engagement in ITSs through the implementation of games and game-based elements. To investigate this possibility, we use both a physical version and a computerized version of the same educational board game. We compare user experience and reading strategy data for the implementation of two games – iSTART: The Board Game and MiBoard. We discuss game design implications for virtual environments and ITSs.

Keywords: Games, Engagement, Reading Comprehension, Reading Strategies, ITS.

1 MiBoard

The current study is a modified replication of Rowe [1], which investigates user experiences with MiBoard [2]. MiBoard (Multiplayer interactive Board) Game was developed as a computer-based proxy for iSTART: The Board Game (iSTART:TBG) [1]. Using the same goals as iSTART:TBG, MiBoard was also intended to be an alternative to iSTART extended practice. We expected students' attitudes and experiences in MiBoard to mirror that of iSTART:TBG, and we expected that MiBoard could be even more effective in promoting learning, by incorporating motivational components, such as individual achievements that sustain concentration for long periods of time [3,4].

The gameplay requirements in MiBoard are a computerized translation of those from iSTART:TBG. MiBoard requires players to read a portion of a text, produce a self-explanation using an assigned strategy, identify the strategies used in other players' self-explanations, and to resolve any disagreements through a chatroom debate. The strategies included within MiBoard are the same as those taught within iSTART training (i.e., Comprehension Monitoring, Paraphrasing, Prediction, Elaboration, and Bridging). As in iSTART:TBG, feedback on a player's self-explanation comes from the voting of the other players and through discussion.

While MiBoard was intended to directly mirror iSTART:TBG, there are noticeable differences between the two. First, MiBoard is completely computer-based, meaning players are required to use a computer interface to complete the game interactions, discussions, and voting. Second, iSTART:TBG allows the reader to choose between

two strategies while MiBoard specifies a single strategy. This feature allows us to control the strategies being practiced by the players. Finally, the most noticeable difference between iSTART:TBG and MiBoard is that within MiBoard, the debate portion of the game occurs via a chat room.

2 Current Study

Twenty-two native English speakers at an urban university participated in the study in exchange for credit in their Introductory Psychology course. Participants were not familiar with iSTART before their exposure during this study. Items similar to those used in Rowe [1] were developed to assess the participants' attitudes and opinions toward MiBoard. These responses were then compared to the results from Rowe [1].

Participants received an abbreviated version of iSTART training (lectures only). This is the same training received by participants in the Rowe study [1]. Participants were then asked to self-explain one of two counterbalanced texts. The texts used in this study were the same texts used in the Rowe study. Self-explanations were scored using the overall iSTART algorithm, which scores on a 0 to 3 scale and awards higher values for more elaborative self-explanations [5]. Participants were divided into anonymous groups of either three or four players and interacted with MiBoard for 30 minutes. After playing MiBoard, they were prompted to self-explain the second counterbalanced text and answer questions about their experience with the system.

Unlike the findings for iSTART:TBG, participants who interacted with MiBoard reported that they did not enjoy using the system. The results of the attitudinal questionnaire for both studies are shown in Table 1. Participants' ratings indicate that MiBoard was not fun to play, not helpful in understanding reading strategy knowledge, and not fun to use. In addition, participants found the game to be slow and thought it was somewhat frustrating.

Table 1. Evaluation Responses from iSTART: The Board Game (Rowe, 2008) and MiBoard (Current Study) (ratings from 1 to 6, higher scores indicate stronger agreement)

Statement	iSTART: TBG		MiBoard	
	M	SD	M	SD
The game was fun	5.51	0.78	2.14	1.39
The game improved strategy knowledge	5.71	0.52	3.77	1.60
The game was easy to use	5.46	0.85	3.81	1.79

Regarding efficacy of training, similar to Rowe [16], using the iSTART scoring algorithm we found there to be no difference between the pretest self-explanation scores ($M = 1.58$, $SD = 0.58$) and the posttest self-explanation scores ($M = 1.47$, $SD = 0.51$), $F(1,19) = 2.069$, $MSE = 0.069$, $p = 0.17$. Indeed, the downward trend of the scores indicates that MiBoard may even detract from the learning process.

3 Discussion

The results of this MiBoard evaluation contrasts negatively with results found with iSTART:TBG. First, the efficacy results are similar, in that neither of the games

produced an improvement in self-explanation score. This is not surprising because the amount of training was relatively short compared to typical iSTART studies. Nonetheless, it is a concern that MiBoard led to a marginal decrease in self-explanation quality.

The enjoyability results also paint a bleak picture. Even though MiBoard was intended to be a reasonable proxy for iSTART:TBG, participants simply did not react in the same way to the two games. Participants in Rowe's study reported that they enjoyed iSTART:TBG and would play it again, whereas the students who interacted with MiBoard did not enjoy the experience and were not interested in playing it again. We believe that the differences between the versions (e.g., input methodology, social cues, etc.) can potentially account for the differing results. Though the input methodology is intended to increase distribution through the removal of physical copresence, it seems that this lack of copresence may have contributed to the negative effects. MiBoard incorporated real time user statuses, individual and global system messages, etc., however these features did not seem to alleviate the loss of visual cues.

Acknowledgements. This research was supported by the Institute for Education Sciences (IES R305A080589; IES R305G020018-02). 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 IES. Special thanks to Justin Brunelle and John Meyers for their instrumental help in coding and conducting this study.

References

1. Rowe, M.P.: *Alternate Forms of Reading Comprehension Strategy Practice: Computer and Game-Based Practice Methods* (Doctoral Dissertation). University of Memphis, Memphis (2008)
2. Brunelle, J.B., Dempsey, K.B., Jackson, G.T., Boonthum, C., Levinstein, I.B., McNamara, D.S.: *MiBoard: iSTART Metacognitive Training through Gaming*. In: SCiP Conference (2009)
3. Graesser, A.C., Chipman, P., Leeming, F., Biedenbach, S.: *Deep Learning and Emotion in Serious Games*. In: Ritterfield, U., Cody, M., Vorderer, P. (eds.) *Serious Games: Mechanisms and Effects*, Routledge, pp. 81–100. Taylor and Francis, Mahwah (2009)
4. Moreno, R., Mayer, R.E.: *Role of Guidance, Reflection and Interactivity in an Agent-Based Multimedia Game*. *Journal of Educational Psychology* 97, 77–128 (2005)
5. McNamara, D.S., Boonthum, C., Levinstein, I.B., Millis, K.: *Evaluating Self-Explanations in iSTART: Comparing Word-Based and LSA Algorithms*. In: Landauer, T., McNamara, D.S., Dennis, S., Kintsch, W. (eds.) *Handbook of Latent Semantic Analysis*, pp. 227–241. Erlbaum, Mahwah (2007)

Players' Motivation and EEG Waves Patterns in a Serious Game Environment

Lotfi Derbali and Claude Frasson

Département d'informatique et de recherche opérationnelle, Université de Montréal
C.P. 6128, Succ. Centre-ville, Montréal, Québec, Canada H3C 3J7
{derbalil, frasson}@iro.umontreal.ca

Abstract. This study investigated players' motivation during serious game play. It is based on a theoretical model of motivation (John Keller's ARCS model of motivation) and EEG measures. Statistical analysis showed a significant increase of motivation during the game. Moreover, results of power spectral analysis showed EEG waves patterns correlated with increase of motivation during different parts of serious game play.

Keywords: Motivation, Serious game, Learning, ARCS model, EEG waves.

1 Introduction

Serious games are computer applications that combine a serious intent, learning and training using video environment or computer simulation. They have become an important social trend. Indeed, game environments have great potential to support immersive learning experiences. Several researches have shown that serious games can provide a suitable context via interactive, engaging and immersive tasks [1, 2]. Additionally, game experience is a psychological and physiological process. During video game, specific physiological and neurological reactions take place in players. Some researchers have examined electrophysiological players' responses such as galvanic skin response and electroencephalography (EEG) activity [3, 4].

In this paper, we study players' motivation during different parts of serious game play. For this purpose, we examine players' motivation using (1) a theoretical model of motivation (John Keller's ARCS model of motivation) and (2) their EEG activity.

2 Procedure

This study involved thirty three volunteer subjects (11 female), who ranged from 19 to 42 years of age. We used the serious game called FoodForce which intended to educate about the problem of world hunger. We also adopted the Keller's ARCS Learning Motivational Model [5] to measure players' motivation. Each subject was placed in front of two computers: one computer is used for playing the serious game and the other one is used to answer ARCS questionnaires. EEG electrodes were placed on four selected sites (F3, Fz, C3 and Pz) according to the international 10-20

system. According to standard EEG definitions, the Fast Fourier Transformation (FFT) divided the EEG into common frequencies: delta (1-4 Hz), theta (4-8 Hz), alpha (8-12 Hz), low-beta (12-20 Hz), high-beta (20-32 Hz) and gamma (33-42 Hz). During our off-line analysis, we resorted to a power spectral density (PSD) of EEG measures to dissect the EEG patterns correlated with the high motivation of subjects.

3 Motivation and ARCS Scores

By using ARCS questionnaires results, we evaluated differences between initial and final motivations during the game. Results of paired sample t-test showed a significant increase of motivation during experiment ($t(32)=-2.650$, 2-tailed $p=0.012<0.05$). More specially, significant effects were found for “Attention” component ($t(32)=-4.950$, 2-tailed $p=0.000<0.05$) and for “Confidence” component ($t(32)=-2.677$, 2-tailed $p=0.012<0.05$) of the Keller’s ARCS model of motivation. According to Keller’s ARCS motivation theory, the learner’s attention is gained possibly by “novel, surprising, or incongruous events...” called perceptual arousal. Furthermore, Keller’s ARCS motivation theory tells us that the learner’s curiosity is aroused by the mean of “solving problem or resolving an open issue...” called inquiry arousal.

4 Motivation and EEG Waves

We firstly noticed that alpha wave’s power tends to decrease and theta wave’s power to increase along of the game (Figure 1). This change of alpha and theta waves was detected at the four selected sites but it is clearer at the middle frontal cortex (Fz) for the majority of subjects.

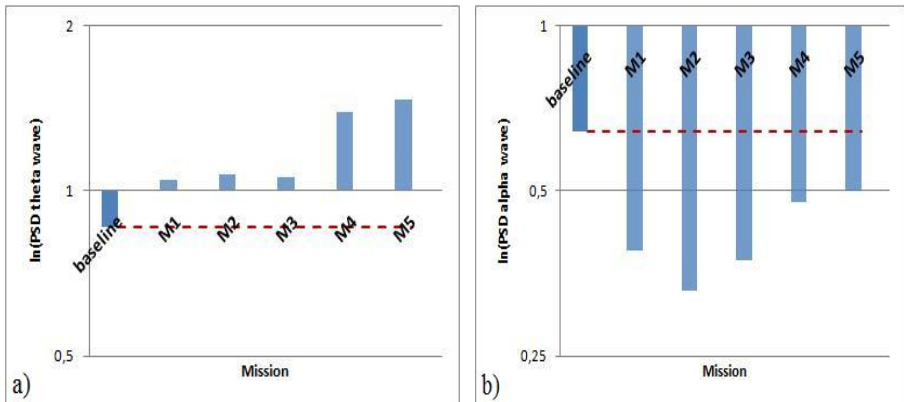


Fig. 1. Power spectral analysis in the Fz site for subject 11 during the game: a) Increase of theta wave power from the baseline, b) Decrease of alpha wave power from the baseline

We also noticed a general increase of power for high frequencies (high-beta and gamma waves) during the game (Figure 2). This finding was again detected for all selected sites but it was mainly related to the left motor cortex (C3).

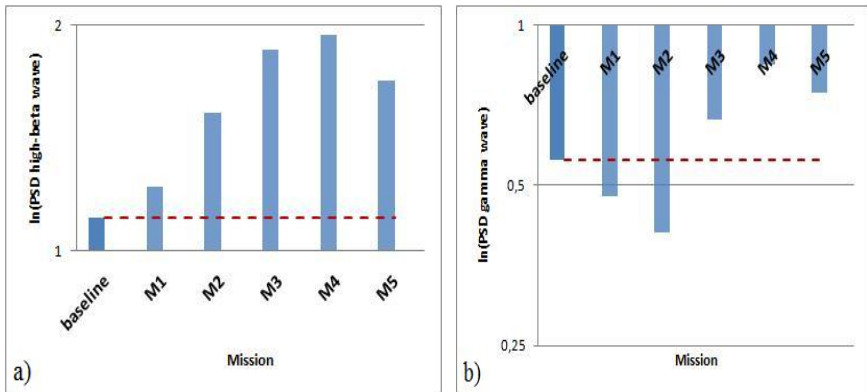


Fig. 2. Power spectral analysis in C3 site for subject 14: a) Increase of high-beta wave from the baseline during the game, b) Increase of gamma wave from the baseline during final missions

We assumed that EEG can provide a valid and objective index for players' motivation during game. Therefore, we studied changes of each EEG wave power during different parts of serious game play. Power spectral analysis showed EEG waves patterns correlated with the increase of motivation: significant increase of high frequencies (high-beta and gamma waves) was detected, theta wave activity increased, and alpha wave activity decreased. Our results confirm some previous studies indications [6] that brain activity during video game playing has a close relation with alpha and theta changes.

Acknowledgments. We acknowledge the support of the FQRSC (Fonds Québécois de la Recherche sur la Société et la Culture), the NSERC (National Science and Engineering Research Council), and the Tunisian Ministry of Higher Education and Scientific Research for this work. We also address our thanks to Pierre Chalfoun for his participation in the experiment setup and his useful comments.

References

1. Prensky, M.: *Digital Game-Based Learning*. McGraw Hill, New York (2001)
2. Johnson, W.L., Wu, S.: Assessing Aptitude for Learning with a Serious Game for Foreign Language and Culture. In: *Proceedings of Intelligent Tutoring Systems*, pp. 520–529 (2008)
3. Kramer, D.: Predictions of Performance by EEG and Skin Conductance. *Indiana Undergraduate Journal of Cognitive Science* 2, 3–13 (2007)
4. Salminen, M., Ravaja, N.: Oscillatory brain responses evoked by video game events: The case of Super Monkey Ball 2. *CyberPsychology & Behavior* 10, 330–338 (2007)
5. Keller, J.M.: *IMMS: Instructional materials motivation survey*. Florida State University (1987)
6. Pellouchoud, E., Smith, M.E., McEvoy, L., Gevins, A.: Mental effort related EEG modulation during video game play: Comparison between juvenile epileptic and normal control subjects. *Epilepsia* 40(4), 38–43 (1999)

Predicting the Effects of Skill Model Changes on Student Progress

Daniel Dickison¹, Steven Ritter¹, Tristan Nixon¹, Thomas K. Harris²,
Brendon Towle¹, R. Charles Murray¹, and Robert G.M. Hausmann¹

¹ Carnegie Learning, Inc., 437 Grant Street, Pittsburgh, PA 15219
{ddickison, sritter, tnixon, btowle, cmurray, bhausmann}@carnegielearning.com
<http://carnegielearning.com>

² EDalytics, LLC
thomas@edalytics.com
<http://edalytics.com>

Abstract. We describe a methodology for simulating student behavior to predict the effects of skill-learning parameter changes on system behavior. Validation against data collected after the changes were made shows that accurate predictions can be made despite a different cohort of students. Furthermore, deviations from the predictions may help explain unexpected effects of other changes made to the tutoring system.

Keywords: data mining, student simulation, model validation.

As we work to optimize the knowledge-tracing parameters in Carnegie Learning's Cognitive Tutors, we are concerned about the effects these changes have on student experience. These parameters are used in Cognitive Tutor's knowledge-tracing system to model student learning [3].

Optimizations of these parameters [7] are intended to ensure that students receive the optimal amount of instruction on a particular topic. The basic form of this work is to find a best fit to user data, modifying parameters that model the speed of learning each underlying knowledge component in the system (see [5]). Because the system presents new problems to a student until every skill has been mastered (estimated $p_{\text{known}} > 0.95$), optimizing these parameters can result in the system presenting a larger or smaller number of problems, as needed by each student (for example, see [2]).

Predicting the effect of skill parameter changes on a student population is complex, and some important factors may not immediately be obvious. For example:

- Some skills have more influence on problem selection and mastery than others. For example, a skill that is very often performed correctly will be judged as mastered relatively quickly under any skill model. Parameter fitting may improve the modeling of that skill, but there will likely be no net effect on problem selection for students or on system behavior.
- Different problems address different subsets of skills, so, given a problem-selection algorithm that favors unmastered skills, changes in one skill may lead to an increase or decrease in opportunities for another skill.

- Problem-selection algorithms are occasionally modified — for example, to increase the variety of problems. Even when the desired variety is not related to skill modeling, such changes may lead to changes in skill opportunities.

Student data from a previous release of the tutor (2007) was used to drive a simulation of individual students. The student outcome data is a binary correct/incorrect for the first attempt at each opportunity to use each monitored skill. The simulation tallies up the skill opportunities for each selected problem and updates the knowledge tracing state according to the skill outcomes of the simulated student. Further problems are selected until mastery or the maximum number of problems is reached. (See [4].)

Because the simulated sequence of problems may differ from that experienced by the student, the simulated sequence of problem step and skill encounters may also differ. The underlying assumption is that a skill encountered in a particular problem step is equivalent to the same skill embedded in a different problem and problem step. This assumption is a consequence of the Decomposition Thesis [1] underlying the cognitive model. However, independence of skills is assured only if the cognitive model is a perfect representation of student learning. In any implemented model, this assumption is likely to be violated, so one question addressed in our work is whether such violations have strong consequences on our ability to predict student performance.

The version of Cognitive Tutor containing the optimized skill parameters was released to customers during the summer of 2009. We have now collected several months worth of actual student usage data from the 2009 version which we can use to validate our predictions of the impacts of these skill parameter changes in several units of instruction. Due to the skill parameter changes, we expect problem counts in 2009 to differ from those observed in 2007. More importantly, if the simulations are reliable, then we expect the predictions to closely mirror the distribution observed in 2009. We analyzed nine units from the Bridge to Algebra curriculum that used new parameters for 2009 and contained data from more than 100 students.

A linear model using only the simulation as a predictor was selected with $AIC = 12.01$ over the models using 2007 data with and without the simulations, with $AIC = 12.55$ and $AIC = 23.72$, respectively. The resulting linear model had a non-significant x -intercept (coeff = 1.2, $p = 0.37$) and a significant coefficient for the simulation predictor (coeff = 0.97, $p < 0.001$). These results suggest that the underlying assumptions in the simulations hold true regarding the median problem counts per section. Cumulative distribution plots of the problem counts (see poster) show that the overall distribution of individual student problem counts match closely for each unit as well. In general, the assumptions underlying the simulation such as the independence of skills appears to hold for generating accurate predictions of student experience.

The simulation does a relatively poor job of predicting performance on one unit: Picture Algebra. The simulation predicted that students would finish the unit in fewer problems than turned out to be the case. The task in Picture Algebra is based on using Singapore-math style diagrams to solve word problems [6]. While

we did not change the skill model for Picture Algebra, we do update various aspects of our system each year, and an analysis of the user interface changes made to Picture Algebra between 2007 and 2009 can explain the discrepancy between the predicted and actual results.

This section has seven tracked skills. Student performance on two skills is worse in 2009 than in 2007 and so these skills are the most likely candidates for understanding why students are doing more problems in 2009 than in 2007. Of the two skills, the one called "Drawing larger bar" was the only one that was a critical skill in a significant percentage of students (58% in 2009). We believe that user interface changes that made a drag-and-drop action less intuitive can account for decreased student performance for this skill and hence for the increased problem count.

By performing simulations as described in this paper, we can ensure that changes we make to skill parameters are in fact improvements before we deploy the product to thousands of customers. The validation presented here shows that the predictions of the simulation are reliable in most cases. The one unit where our predictions were inaccurate was likely due to changes made to the user interface. The accuracy of the simulation allows us to use failures of prediction to examine the impact of changes unrelated to skill parameters. We believe that this approach can be a very productive and efficient method for improving the effectiveness of intelligent tutoring systems.

References

1. Anderson, J.R.: Spanning seven orders of magnitude: a challenge for cognitive modeling. *Cognitive Science* 26, 85–112 (2002)
2. Cen, H., Koedinger, K.R., Junker, B.: Is Over Practice Necessary?—Improving Learning Efficiency with the Cognitive Tutor using Educational Data Mining. In: Lucken, R., Koedinger, K.R., Greer, J. (eds.) *Artificial Intelligence in Education, Building Technology Rich Learning Contexts That Work*, Proceedings of the 13th International Conference on Artificial Intelligence in Education, pp. 511–518 (2007)
3. Corbett, A.T., Anderson, J.R.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-Adapted Interaction* 4, 253–278 (1995)
4. Dickison, D., Ritter, S., Harris, T.K., Nixon, T.: A Method for Predicting Changes in User Behavior in Cognitive Tutors. In: *AIED 2009: 14th International Conference on Artificial Intelligence in Education, Workshop Proceedings. Scalability Issues in AIED Workshop*, vol. 4 (2009)
5. Harris, T.K., Ritter, S., Nixon, T., Dickison, D.: *Hidden-Markov Modeling Methods for Skill Learning*. Carnegie Learning Technical Report (2009)
6. Koedinger, K.R., Terao, A.: A cognitive task analysis of using pictures to support prealgebraic reasoning. In: Schunn, C.D., Gray, W. (eds.) *Proceedings of the Twenty-Fourth Annual Conference of the Cognitive Science Society*, pp. 542–547. Lawrence Erlbaum Associates, Mahwah (2002)
7. Ritter, S., Harris, T.K., Nixon, T., Dickison, D., Murray, R.C., Towle, B.: Reducing the Knowledge Tracing Space. In: Barnes, T., Desmarais, M., Romero, C., Ventura, S. (eds.) *Proceedings of Educational Data Mining 2009: 2nd International Conference on Educational Data Mining* (2009)

Data Mining to Generate Individualised Feedback

Anna Katrina Dominguez, Kalina Yacef, and James R. Curran

School of Information Technologies, University of Sydney, Australia
{adom0244, kalina, james}@it.usyd.edu.au

Abstract. Intelligent Tutoring Systems can be very expensive and complex to design, build and maintain. We explore the feasibility of adding automatic personalised feedback to an existing online learning system, by mining the student data collected by the system. This work was carried out on a web site in which students are taught programming basics in Python. Using 2008 and live 2009 data, the 2009 system generated hints to help students in topic areas they were found to be struggling with. We found that students who used the hinting system achieved significantly better results (26% higher marks) than those who did not, and stayed active on the site longer. A qualitative survey also revealed positive feedback from the students.

1 Introduction

Our goal is to investigate whether data mining can be used to provide tailored hints to users of a learning system without the overhead of building an ITS from scratch.

There are a number of successful programming tutors (e.g. [1-3]) which provide hints, often based on a finite set of rules, to help students solve problems. However these rules need to be encoded, require the formulation of pre-determined feedback and use domain specific methods for analyzing students' answers.

Our approach is to provide feedback to students through the introduction of data mining *directly into the system loop* to generate hints. We build on the NCSS Challenge¹, an annual online programming competition in which high school students are taught the basics of programming in Python over 5 weeks. The hinting system helps students access (i) parts of the notes explaining topics that they struggle with and (ii) other relevant questions. They are generated through the use of patterns that have been discovered in Challenge data from previous and current (live) years. Using these patterns as opposed to manually constructing solutions and feedback means the system is built with less costly overhead than traditional ITSs.

The hinting system is fully operational, and was used in the 2009 Challenge. We describe here its overall architecture and highlight some results of our experiment.

2 System Architecture

The overall operation of the Challenge system can be seen in Figure 1.

¹ <http://challenge.ncss.edu.au>

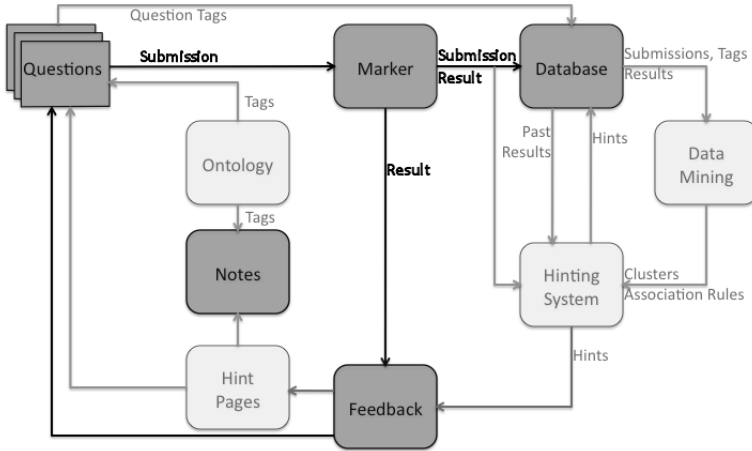


Fig. 1. Overall structure of the system. Light shaded sections correspond to the new components created to provide hints based on data mining.

Do you need help with Question 2.5: `encrypt.reverse()`?

The question covers the following topics which you might like to go over:

- [String slicing](#)
- [getting String lengths](#)
- We've found that students who find [String concatenation](#) and [String slicing](#) difficult can also find other topics difficult. Try to before moving on to [Matching characters/substrings](#).

Doing these similar questions you haven't finished might help you:

- [2.3: lerCrypto scamb](#)

You might like to finish some easier questions first:

- [1.5: Find the longest string](#)

Fig. 2. The topic and question hint page

The darker sections in Figure 1 outline the core components of the Challenge system, i.e. notes, questions, automatic marker, storage database and feedback given to students. The lighter sections outline the extended intelligent components and behaviour of the Challenge system, which we summarise below.

The *hints* provide users with links to specific sections of the notes. To allow this, we constructed a lightweight *ontology* organising all the Python topics covered in the notes into a tree structure. All notes sections and questions were then tagged with the relevant topics, allowing for the notes and questions to be associated with each other.

The *hinting system* uses the tags for the questions and notes, along with the student submissions and results data from the 2008 Challenge, and patterns extracted with *data mining*. We found sequences of topics students had trouble with using association rule mining. We also clustered students into groups of similar ability levels, and clustered questions by similarity of topics and of difficulty. Details of the data mining process can be found at [4].

Hints are created specifically for the question the student is attempting, and are based on the student's performance, the topics the question is tagged with, and the patterns mined from previous data as well as from the live data generated by the current participants of the Challenge. An example of a hint is shown in Fig 2.

3 Experiment Results

We tested our hinting system on 584 participants of the 2009 NCSS Challenge. We split them into a test group which received hints, and a control group which did not.

In the hinted group, the mean score of the students' submissions was 4.02 out of 10 (sd = 2.78), while the control group had a mean score of 3.18 (sd = 2.71). The difference of 0.84, i.e. an increase of 26.4% ($p < 0.0006$) suggests that the hints helped students. There were also consistently more users in the hinted group who made at least one submission per week, suggesting a longer participation from these users.

Lastly, we presented a questionnaire (five-point Likert scale) at the end of the course to measure the students' satisfaction with the hinting system. Hints and questions were judged "somewhat relevant" or "relevant" by 67% and 90% of students respectively. 71% of students appreciated the hints and were willing to have more of them. When asked to provide comments, many students stated that the hints had helped them with problem solving, with students giving extremely positive comments and requests for the hints to continue in future years of the Challenge. A student commented well our aims: *"I found the tips more helpful, because when we are using the notes to solve the problem we really don't know where to go and what to do or which formula to use. But after using the hint formula we know where to go and what to use for solving the problem. So I reckon that the hint boxes were a very smart way to access the notes that can help us to solve the problems."*

4 Conclusion

Our project aimed to integrate data mining into an e-learning system in order to create a system that provides individualised hints to users. These hints give users immediate help by directing them to parts of the course notes that are relevant to questions that they find difficult.

Our experiment suggests that hints can be generated dynamically through the use of data mining to provide hints, at a much lower designing cost than traditional ITSs, and that the resulting system is effective.

References

1. Anderson, J.R., Reiser, B.J.: The LISP tutor: it approaches the effectiveness of a human tutor. *Byte* 10, 159–175 (1985)
2. Mitrovic, A.: An intelligent SQL tutor on the web. *International Journal of Artificial Intelligence in Education (IJAIED)* 13, 173–197 (2003)
3. Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-based Instruction. *International Journal of Artificial Intelligence in Education* 12, 351–384 (2001)
4. Dominguez, A.K., Yacef, K., Curran, J.R.: Data Mining for Individualised Hints in eLearning. In: *Educational Data Mining (EDM 2010) Conference*, Pittsburgh, USA (2010) (submitted)

In the Zone: Towards Detecting Student Zoning Out Using Supervised Machine Learning

Joanna Drummond and Diane Litman

Department of Computer Science, Sennott Square,
University of Pittsburgh, Pittsburgh, PA 15260
{jmd73,litman}@cs.pitt.edu

Abstract. This paper explores automatically detecting student zoning out while performing a spoken learning task. Standard supervised machine learning techniques were used to create classification models, built on prosodic and lexical features. Our results suggest these features create models that can outperform a Bag of Words baseline.

Keywords: Zoning Out, Natural Language, Machine Learning.

1 Introduction

Recent investigations suggest detecting and adapting to student affect and other states could improve student learning and other performance measures for intelligent tutoring systems (e.g., [1,2,3,4,5,6]). Current detection methods include measuring response times [3] and using lexical, prosodic and other linguistic features [2,6].

Zoning out, a state defined as “thinking about other things while [performing a learning task]” [7], was shown to negatively impact student learning [7]. Thus, Moss calls for investigating intelligent tutoring systems that adapt to zoning out [7]. While little work has investigated detecting zoning out, detecting disengagement, a closely related phenomena, has been explored (e.g., [3,4]).

Given the promise of language-based affect-adaptive tutors [1,2] and the link between zoning out in a spoken learning task and normalized learning gains [7], we feel that spoken tutoring systems that adapt to students’ zoning out have the potential for improving student learning. Therefore, we wish to show it is feasible to build models to automatically detect zoning out.

2 Dataset and Features

Our corpus, a subset of Moss’s [7], contains novice undergraduate students reading aloud a biology paragraph, then performing a learning task (paraphrase, or self explain) aloud. Students’ audio was recorded and human-transcribed, with transcriptions including common spoken disfluencies. At set intervals, the student took a short survey with the text “I found myself zoning out and thinking about other things when reading this text” with a Likert scale underneath, with 1 being “All the time,” and 7 being “Not at all.” So, we will attempt to classify

at this granularity. We combine everything the student read in that interval into one **Text**, and what the student produced via the learning task into one **Task**.

Since we wish to show detecting zoning out is possible, we group student self-reports into two categories: “*High*” if the student reports 1-3, and “*Low*” if the student reports 5-7, discarding borderline reports of 4. We have 52 instances of students self-reporting *High*, 63 reporting *Low*, and 20 discarded due to reporting 4. Therefore, we have a total of 115 data points, from 37 students.

We only present features chosen by the feature selection algorithm used in the machine learning experiments. *Transcript-Based Features* use our human transcriptions. **WC Text** calculated the number of words the student read, and **WC Diff** counted the difference between **WC Text** and the number of words the student said in their **Task**. We also created wordlists by investigating students’ **Task** data. These wordlists generate word-count-based features. **Confusion** wordlist tried to capture when a student acknowledged that they were confused. **References** attempted to indicate when a student personalized the information. **Disfluencies** counted the number of human-annotated spoken disfluencies (filled pauses, unfilled pauses, and false starts) found in the student’s **Task**. **Bag of Words** counts the number of times each word in the **Task** vocabulary is said in this data point’s **Task**, making each word a unique feature. Our baseline model is built using only this feature.

Audio-based Features are commonly used to classify user states in spoken systems. We used an implementation previously developed for detecting student affect. **Percent Text Silence** is the amount of internal silence divided by the time the student is actively speaking and their internal silence. **Text** and **Task Min Pitch** is the student’s minimum pitch in that segment. **Text** and **Task Min Energy** describes loudness instead of pitch.

3 Machine Learning Experiments and Results

We present results from one machine learning algorithm, to show this task is feasible. The Bag of Words baseline and two experimental models were built using the J48 Decision Tree algorithm implemented by Weka, which includes a feature selection algorithm. Due to our small dataset, we used the leave-one-out cross-fold validation training/testing paradigm. We chose accuracy, precision and recall as our evaluation metrics. We then tested for differences between our models and the baseline using a two-tailed t-test.

The quantitative performance of our models can be found in Table 1. We evaluate our models using the three metrics, applied to both *High* and *Low*. We have highlighted the best performance for our experimental models in each metric in the table. This table also shows the results of the t-test. The Bag of Words row shows the performance of our baseline. The next row details our first experimental model, built with All designed features, a superset of those presented in Section 2, excluding **Bag of Words**. Qualitatively assessing this model, **Text Min Pitch**, **Disfluencies**, and **References** are the most important features. As **Text Min Pitch** was the root node of All’s decision tree, we built the Text Min Pitch model, using only this feature. This model performed best in all metrics except *Low* Recall.

Table 1. Leave-One-Out Cross-Fold Validated Performance (N = 115); * Significantly higher than baseline at $p = 0.05$, † at $p = 0.10$

Model	Accuracy	<i>High</i> Precision	<i>High</i> Recall	<i>Low</i> Precision	<i>Low</i> Recall
Bag of Words	0.522	0.474	0.519	0.569	0.524
All Features	0.583	0.548	0.442	0.603	0.698*
Text Min Pitch	0.643†	0.580	0.769*	0.739*	0.540

4 Conclusions and Future Work

Our long-term goal is to enhance spoken tutorial systems to detect and adapt to *High* zoning out students. We have shown that even with a small dataset, it is feasible to build a model to detect students' self-reported zoning out that outperforms a Bag of Words baseline. In addition, our automated audio-based features were very important in detecting zoning out, suggesting it's possible to automatically detect student zoning out in a real-time tutorial dialogue system.

To improve our results, we wish to explore different machine learning algorithms and different methods of feature selection. We also wish to explore fully automating all transcript-based features. In addition, we wish to apply our results to detecting disengagement in a spoken physics tutorial dialogue system [1].

Acknowledgements. We thank Dr. C. Schunn and Dr. J. Moss for our data, M. Lipschultz and ITSPROKE group for comments, and NSF grant #0631930.

References

1. Forbes-Riley, K., Litman, D.: A user modeling-based performance analysis of a wizarded uncertainty-adaptive dialogue system corpus. In: Proc. Interspeech, Brighton, UK (September 2009)
2. Pon-Barry, H., Schultz, K., Bratt, E., Clark, B., Peters, S.: Responding to student uncertainty in spoken tutorial dialogue systems. Intl. Journal of AIED (2006)
3. Beck, J.: Using response times to model student disengagement. In: ITS (2004)
4. Cocea, M., Weibelzahl, S.: Log file analysis for disengagement detection in e-Learning environments. User Modeling and User-Adapted Interaction (2009)
5. Lehman, B., Matthews, M., D'Mello, S., Person, N.: What are you feeling? In: Investigating student affective states during expert human tutoring sessions. LNCS (2008)
6. D'Mello, S., Craig, S., Witherspoon, A., Mcdaniel, B., Graesser, A.: Automatic detection of learners affect from conversational cues. User Modeling and User-Adapted Interaction (2008)
7. Moss, J., Schunn, C.D., VanLehn, K., Schneider, W., McNamara, D.S., Jarbo, K.: They Were Trained, But They Did Not All Learn: Individual Differences in Uptake of Learning Strategy Training. In: Proc. of 30th Annual Meeting of the Cognitive Society (2009)

Can We Get Better Assessment from a Tutoring System Compared to Traditional Paper Testing? Can We Have Our Cake (Better Assessment) and Eat It too (Student Learning during the Test)?

Mingyu Feng¹ and Neil Heffernan²

¹ SRI International, Menlo Park, CA 94025

² Worcester Polytechnic Institute, Worcester, MA 01609
mingyu.feng@sri.com, nth@wpi.edu

Abstract. Dynamic assessment (DA) has been advocated as an interactive approach to conducting assessments to students in the learning systems. Sternberg and others proposed to give students tests to see how much assistance it takes a student to learn a topic; and to use as a measure of their learning gain. To researchers in the ITS community, it comes as no surprise that measuring how much assistance a student needs to complete a task successfully is probably a good indicator of this lack of knowledge. However, a cautionary note is that conducting DA takes more time than simply administering regular test items to students. In this paper, we report a study analyzing 40-minutes data of totally 1,392 students from two school years. The result suggests that for the purpose of assessing student performance, it is more efficient for students to take DA than just having practice items.

Keywords: Dynamic assessment, assessment in learning system.

1 Introduction

In the past twenty years, much attention from the Intelligent Tutoring System community has been paid to improve student learning while the quality of assessment has not been emphasized as much. In the US, state tests are causing many schools to give extra tests. It would be great if ITSs could be used to do the tests, so that no time from instruction is taken away. Many psychometricians would argue that let students learn while being tested will make the assessment harder since you are trying to measure a moving target. Can ITSs, if given the same amount of time, be better assessors of students (while also providing the benefit of helping students learn during that time period)?

Assessing students accurately without interfering with learning is an appealing but also a challenging task. Dynamic assessment (DA, also called dynamic testing) [3] has been advocated as an interactive approach to conducting assessments to students. DA uses the amount and nature of the assistance that students receive to judge the extent of student knowledge limitations (e.g. [3], [4], [5]) or measures student learning potential (e.g. [2]). ITSs are perfect test beds for DA as they naturally lead students into a tutoring process to help students with the difficulties. We [1] have

collected extensive information to assess students dynamically in a computer)-based tutoring system (<http://ASSISTments.org>). In this system if a student has trouble solving a problem (the **main** item, the system provides instructional assistance by breaking the problem into a few **scaffolding** steps, or displaying **hint** messages on the screen upon request. Although DA has been shown to be effective predicting student performance, yet there is a cautionary note: since students are allowed to request assistance, it generally takes longer to finish a test using the DA approach than using a traditional testing approach.

2 Methods

Fundamentally, in order to find out whether DA was worth the time, we would want to run a study comparing the assessment value of the following two conditions: Static assessment condition (A): students were presented with one static test item and were requested to submit an answer; Dynamic assessment condition (B): students were presented with one static test item followed by a DA portion where they could request help. Since ASSISTments had collected data with the information needed, we chose to compare predictions made based on log data from 40 minutes of time across *simulated* conditions that were similar but not exactly the same as above: **Simulated static assessment condition (A')**: 40 minutes of student work selected from existing log data on only main items; **Dynamic assessment condition (B')**: 40 minutes of work selected from existing logged response data on both main items and the scaffolding steps and hints. Such a simulation study not only saved time, but also allowed us to compare the same student's work in different conditions, which naturally rules out the subject effect. We chose to use student's end of year state accountability test score as the measure of student achievement.

We considered two data sets, one from the 2004-2005 school year with 628 students, and the other from 2005-2006 school year of 764 students. The online metrics for dynamic testing that measures student accuracy, speed, attempts, and help-seeking behaviors are simply:

- Main_Percent_Correct – students' percent correct on main questions
- Main_Count - the number of main items students completed.
- Scaffold_Percent_Correct - students' percent correct on scaffolding questions.
- Avg_Hint_Request - the average number of hint requests per question.
- Avg_Attempt - the average number of attempts students made for each question.
- Avg_Question_Time - on average, how long it takes for a student to answer a question, whether original or scaffolding, measured in seconds.

The last five metrics are DA style metrics and were not measured in traditional tests. We ran stepwise linear regression to use the metrics described above to predict student state test scores (the dependent variable). For condition A', the independent variable of the simple linear regression model was *Main_Percent_Correct*; while for condition B', it changed to be the collection of the DA style metrics.

Looking at the results, we noticed that in both years, students finished more test items in the 40 minutes in static condition than in dynamic condition (22 vs 11 in the first year; 31 vs. 13 in the second year). We examined the parameters in the linear regression models esp. for the dynamic condition. The first three parameters entered the models were the same in both years (with the order changed a little bit).

Scaffold_Percent_Correct was the most significant predictor in the first year while in the second year, it was Main_Percent_Correct. Also, in the later year 2005-2006, Avg_Attempt was considered as a significant predictor while in the first year it was Avg_Hint instead. We also chose to use Bayesian Information Criterion (BIC) to compare the generalization quality of the model. In both years, the R squares of the model from the dynamic condition were higher, and BICs were significantly lower than those of the static condition, suggesting DA condition did a statistically significantly better job at predicting state test scores than the static condition did. We also conducted 5-fold cross validation on the 2004-2005 data and noticed variables in the trained regression models of the DA condition were consistent across the 5 folds validation. So we took the average of coefficients from the five trained regression models and applied the average model on the full data set. Mean absolute difference (MAD) was calculated as a measure of prediction accuracy. The average model from the simulated static condition and the DA condition produced MAD of 9.01 (out of 54) and 8.7 respectively. The paired t-test suggested that there was a marginally significant difference ($p=0.10$).

Based on the results, we conclude that dynamic assessment is more efficient than just giving practice test items. DA can produce more accurate assessment of student math performance, even limited by using the same amount of testing time. This is surprising as students in the dynamic assessment do few problems.

3 Conclusion

In this paper, we compared DA against a tough contrast case where students were doing assessment all the time in order to evaluate efficiency and accuracy of DA in a tutoring system. This paper eliminates the cautionary note about dynamic assessment that says DA will always need a longer time to do as well at assessing students, which further validates the usage of tutoring systems for assessment.

Acknowledgements

We would like to acknowledge funding from the US Department of Education, the National Science Foundation, the Office of Naval Research and the Spencer Foundation. All of the opinions expressed in this paper are those solely of the authors and not those of our funding organizations.

References

1. Feng, M., Heffernan, N.T., Koedinger, K.R.: Addressing the assessment challenge in an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research* 19(3), 243–266 (2009)
2. Fuchs, L.S., Compton, D.L., Fuchs, D., Hollenbeck, K.N., Craddock, C.F., Hamlett, C.L.: Dynamic assessment of algebraic learning in predicting third graders' development of mathematical problem solving. *Journal of Educational Psychology* 100(4), 829–850 (2008)
3. Grigorenko, E.L., Sternberg, R.J.: Dynamic testing. *Psychological Bulletin* 124, 75–111 (1998)
4. Sternburg, R.J., Grigorenko, E.L.: All testing is dynamic testing. *Issues in Education* 7, 137–170 (2001)
5. Sternburg, R.J., Grigorenko, E.L.: *Dynamic testing: The nature and measurement of learning potential*. Cambridge University Press, Cambridge (2002)

Using Data Mining Findings to Aid Searching for Better Cognitive Models

Mingyu Feng¹, Neil T. Heffernan², and Kenneth Koedinger³

¹ SRI International, Menlo Park, CA 94025

² Worcester Polytechnic Institute, Worcester, MA 01609

³ Carnegie Mellon University, Pittsburgh, PA 15213

mingyu.feng@sri.com, nth@wpi.edu, koedinger@cmu.edu

Abstract. One key component of creating an intelligent tutoring system is forming a model that monitors student behavior. Researchers in machine learning area have been using automatic/semi-automatic techniques to search for cognitive models. One of the semi-automatic approaches is learning factor analysis, which involves human making hypothesis and identifying difficulty factors in the related items. In this paper, we propose a hybrid approach in which we leverage findings from our previous educational data mining work to aid the search for a better cognitive model and thus, improve the efficiency of LFA. Preliminary results suggest that our approach can lead to significantly better fitted cognitive models fast.

Keywords: Data mining, cognitive model, learning factor analysis.

1 Introduction

One key component of creating an intelligent tutoring system (ITS) is forming a model that monitors student behavior. An ITS needs the construction of complex models to represent the skills that students are using and their knowledge states, tracks their progress and chooses what problems will be displayed next. Using a better cognitive model, a system should be able to do a better job of predicting which items students will get correct in real-time and thus do a better job of selecting the next best item. A better model would also help teachers adjust their instruction in a data-driven manner. Given the importance of cognitive models, their construction and improvement has been a major focus in the community. Researchers in machine learning area have been using techniques such as the rule space method [6], Q-matrices [1] learning factor analysis (LFA) [2], learning factor transfer [5] to build/search for cognitive models. We propose a hybrid approach to leverage findings from educational data mining to aid searching for a better cognitive model using LFA.

2 Methods

Cen, Koedinger & Junker [2] proposed a generic, computation intensive method called learning factor analysis (LFA) for cognitive model evaluation and refinement.

LFA aims to “combine statistics, human expertise and combinatorial search to evaluate and improve a cognitive model”. In LFA, a difficulty factor is a hidden feature in a problem that makes the problem easier or harder to solve. An example factor in math with two possible values is using a rule CIRCLE-AREA (e.g. $S = \pi * r^2$) *forward* (to calculate circle area given radius) or *backward* (to calculate radius given circle area). Here *forward* and *backward* are values of a difficulty factor. Given the difficulty factor, LFA could apply one of the three operators “split”, “add”, and “merge” on skills in current based model to generate sub-models and perform combinatorial search to look for the “best” model. By applying operator “add” to existing cognitive model, it is hypothesized that there is an unrepresented skill required by the items that are associated with a difficulty factor. The “merge” operator assumes students only need one representation for multiple skills; yet the “split” operator hypothesizes multiple representations be used to represent the variation in one piece of knowledge component. Various heuristics such as Akaike's information criterion (AIC), Bayesian Information Criterion (BIC), R-square and Log likelihood, have been considered as model evaluation and selection measures.

We [3] have conducted a focused item-level analysis of a subset of items to track how student performance on items changed during the same session in a web-based tutoring system (ASSISTments). Items that have same deep features or knowledge requirements, such as approximating square roots, but have different surface features, such as cover stories, were organized into a **Group of Learning Opportunity (GLOP)**. We reported how we could reliably tell which item) is most effective at causing learning. We found out the items vary in their instructional effectiveness of the skill(s) associated with the group of items.

Now that we could reliably tell difference of learning among items, we wanted to employ this information to improve existing cognitive models. As a basis of LFA, the identification of difficulty factors needs human expertise. They have always been found by subject experts through a process of “difficulty factor assessment” [4]. Based upon theory or task analysis, researchers hypothesized the likely factors that cause student difficulties, and by assessing performance difference on pairs of problems that vary by only one factor, the experts identified the hidden knowledge component that could be used to improve a cognitive model. Can we raise efficiency of LFA by suggesting difficulty factors automatically yet still get better models? We have found certain items in a random sequence cause significantly less learning than others. Intuitively, it is highly possible that there is certain factor inherited in the items, which makes it harder for the learning from this item to transfer to later items. This could be either because later items demand more skills than the current one, or because what a student learns from a current item does not help later items. In both conditions, there is probably “mis-tagging” with this item. Presumably, such a factor can be utilized by LFA to manipulate the original cognitive model to search for the best-fit model.

In order to test this idea, we create factor tables for each of the GLOPs. In each table, we use one factor with two values “High” and “Low” indicating the effectiveness of the items. The item that has caused least learning is associated with the factor value “Low” while all other items are associated with “High”. Given the factor tables, we ran LFA search over all the GLOPs. BIC was used as the heuristic to evaluate the models in that it balances simplicity and predictive power of models. Among the 38 GLOPs, LFA was able to find statistically significantly better models (a difference of 10 points or more on BIC) for 12 of them, using the factors as assigned in the factor tables. Among the 12 GLOPs, 5 of them included 2 items; 3 included 4 items; the rest

4 GLOPs had 5, 6, 8, and 9 items respectively. For 11 out of the 12 GLOPs, the application of the “add” operator led to a better fitted model, which suggests that there were more knowledge other than the current skills that needed to be represented in the cognitive model in order to better track student learning. We have also conducted a sanity check where we randomly assign one item each GLOP with the “Low” value of the factor, and then run the same searching process. Obviously, for the 2-item GLOPs, the results will be the same as before. But for GLOPs with more items, the search process using randomly assigned factor values only find better models for 2 out of the 27 GLOPs, which makes our previous results of 7 out of 27 somewhat impressive. The results show some validity for the very simple way of suggesting factors.

3 Conclusion

This paper describes one practice on how to use educational data mining findings to help improve cognitive modeling. A semi-automatic approach, LFA, is considered. Preliminary results show findings in item effectiveness can be used to assign difficulty factors for items, and thus, automate LFA so that it can efficiently search for superior cognitive models. In terms of future work, we would like to apply this approach on data collected from other tutoring systems to verify the generality.

Acknowledgements

We would like to acknowledge funding from the US Department of Education, the National Science Foundation, the Office of Naval Research and the Spencer Foundation. All of the opinions expressed in this paper are those solely of the authors and not those of our funding organizations.

References

1. Barnes, T.: Q-matrix Method: Mining Student Response Data for Knowledge. In: Beck, J. (ed.) *Educational Data Mining: Papers from the 2005 AAAI Workshop*. AAAI Press, Menlo Park (2005)
2. Cen, H., Koedinger, K., Junker, B.: Learning factors analysis: a general method for cognitive model evaluation and improvement. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) *ITS 2006*. LNCS, vol. 4053. Springer, Heidelberg (2006)
3. Feng, M., Heffernan, N., Beck, J.: Using learning decomposition to analyze instructional effectiveness in the ASSISTment system. In: Dimitrova, Mizoguchi, du Boulay (eds.) *Proceedings of the 14th International Conference on Artificial Intelligence in Education (AIED 2009)*. IOS Press, Amsterdam (2009)
4. Koedinger, K.: Research statement for Dr. Kenneth R. Koedinger (June 2000), <http://pact.cs.cmu.edu/koedinger/koedingerReserach.html>
5. Pavlik, P.I., Cen, H., Koedinger, K.R.: Learning factors transfer analysis: Using learning curve analysis to automatically generate domain models. In: Barnes, Desmarais, Romero, Ventura (eds.) *Proceedings of the 2nd International Conference on Educational Data Mining*, Cordoba, Spain, pp. 121–130 (2009)
6. Tatsuoka, K.: Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Education Measurement* 20(4), 345–354 (1983)

Generating Proactive Feedback to Help Students Stay on Track

Davide Fossati¹, Barbara Di Eugenio², Stellan Ohlsson³,
Christopher Brown⁴, and Lin Chen²

¹ College of Computing, Georgia Institute of Technology

² Department of Computer Science, University of Illinois at Chicago

³ Department of Psychology, University of Illinois at Chicago

⁴ Department of Computer Science, U.S. Naval Academy

Abstract. In a tutoring system based on an exploratory environment, it is also important to provide direct guidance to students. We endowed iList, our linked list tutor, with the ability to generate proactive feedback using a procedural knowledge model automatically constructed from the interaction of previous students with the system. We compared the new version of iList with its predecessors and human tutors. Our evaluation shows that iList is effective in helping students learn.

Keywords: Feedback, proactive interaction.

1 Introduction

In this paper, we address the challenge of automatic generation of student guidance in a problem-based, exploration-oriented Intelligent Tutoring System. In such systems, students are usually presented with problems and an environment in which to solve them. The system can provide support to students by means of feedback, on-demand help, or even interactive dialogue [1,5]. Systems that rely mostly on the exploration of a simulated environment try to encourage knowledge construction. However, recent evidence suggests that minimally guided instruction does not work as well as expected [4], and students do benefit from direct guidance from instructors or more experienced peers. In previous work, we explored the impact of increasingly sophisticated feedback in a mostly exploratory environment [2,3]. In this paper, we make a further step in providing guidance to students by generating a new form of *proactive feedback*.

2 Feedback in the iList Tutor

The iList tutor [1] helps students explore and learn about linked lists by providing graphical representations that can be interactively manipulated with programming language commands. The visualization of linked lists is updated in real time

¹ iList is freely accessible at <http://www.digitaltutor.net>

according to the actions performed by the students. The system provides a set of problems that can be solved by providing sequences of operations that transform the original lists into the desired configurations. The tutor can provide different types of feedback. In previous work we presented *syntax*, *execution*, *final*, and *reactive procedural* feedback [2,3]. Here we introduce *proactive procedural feedback*, an interactive tutor-student interaction composed of three parts:

1. A question from the tutor, including a statement of the goal to be achieved by the following move; the explicit question about how to accomplish that goal; and a set of up to four choices including the correct answer and some of the most frequent incorrect answers given by students. Example: “Let’s see what we can do now... Pointer T is pointing to node 5, we want it to point to null. How would you do that? (1) T = NULL; (2) delete T;”
2. An answer from the student, given by clicking on one of the given choices.
3. Feedback from the tutor. If the answer was right, the message is a positive statement such as “That sounds right! I suggest you try it now.” If the answer was incorrect, the message points out the mistake and illustrates the consequences of that choice. Example: “Uhhh... This is probably not a good idea. Here is what will happen if you do what you suggested. You will delete the node that is pointed by pointer T and that contains 2. Variable T is now pointing to node 2, then it will point to garbage.”

Students must complete the entire interaction before they can continue to work on the problem. To decide when to start it, iList monitors the student’s activity. If the situation is considered critical and enough time has elapsed since the last move, iList initiates the proactive interaction. To make this determination, the current state of the problem is matched against a probabilistic graph that assigns likelihoods to states and actions, evaluates the quality of students’ moves in terms of probability of eventual success, and records the time spent by students in different states. This graph was automatically built from the interaction of previous students with the system [3].

3 Evaluation and Future Work

We evaluated five versions of iList by measuring students’ learning gain as the difference between a pre-test and a post-test. Versions 1, 2, and 3 of iList could provide syntax, execution, final, and reactive procedural feedback to various degrees. Versions 4 and 5 could additionally generate proactive procedural feedback, although in version 4 it was very infrequent. In our comparison we also included a control group of students that did not receive any form of instruction, and a group of students that interacted with a human tutor (Table II). ANOVA revealed an overall significant difference among the seven groups ($F(6, 319) = 3.04$, $P = .007$). Tukey post-hoc tests revealed point-to-point significant differences only between the control group and the human tutored group ($P = .004$), and between the control group and iList-5 ($P = 0.021$). The progression of effect sizes indicates an overall positive trend. As iList is enhanced with additional features, its performance moves closer to that achieved by human tutors.

Table 1. Learning gain of students in seven conditions

Tutor	N	Pre-test		Post-test		Gain	
		μ	σ	μ	σ	μ	σ
None	53	.34	.22	.35	.23	.01	.15
iList-1	61	.41	.23	.49	.27	.08	.14
iList-2	56	.31	.17	.41	.23	.10	.17
iList-3	19	.53	.29	.65	.26	.12	.24
iList-4	53	.53	.24	.63	.22	.10	.16
iList-5	30	.37	.24	.51	.26	.14	.17
Human	54	.40	.26	.54	.26	.14	.25

A limitation of this comparison is that it was conducted over several semesters and across different institutions. Thus, it is difficult to factor out numerous confounding variables such as differences in student population. Future evaluations will be run in a way more conducive to controlling for such differences. Currently, iList covers only linked lists. We are planning on increasing the number of data structures covered by the system, and augmenting the interaction capabilities between the students and the system. More fundamentally, we would like to explore additional pedagogical strategies, to make the system more responsive to students that might be more sensitive to different modes of learning.

Acknowledgments. This work is supported by the ONR (N00014-07-1-0040, N00014-00-1-0640); the UIC Graduate College (Dean's Scholar Award 2008-2009); and the NSF (ALT-0536968, IIS-0133123, 0937060 to CRA for CIF-186).

References

1. Evens, M., Michael, J.: One-on-one Tutoring by Humans and Machines. Lawrence Erlbaum Associates, Mahwah (2006)
2. Fossati, D., Di Eugenio, B., Brown, C., Ohlsson, S., Cosejo, D., Chen, L.: Supporting computer science curriculum: Exploring and learning linked lists with iList. *IEEE Transactions on Learning Technologies*, Special Issue on Real-World Applications of Intelligent Tutoring Systems (2009) (in press)
3. Fossati, D., Di Eugenio, B., Ohlsson, S., Brown, C., Chen, L., Cosejo, D.: I learn from you, you learn from me: How to make iList learn from students. In: *AIED 2009, The 14th International Conference on Artificial Intelligence in Education*, Brighton, UK (July 2009)
4. Kirschner, P.A., Sweller, J., Clark, R.E.: Why minimal guidance during instruction does not work: an analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist* 41(2), 75-86 (2006)
5. VanLehn, K., Lynch, C., Schulze, K., Shapiro, J.A., Shelby, R.H., Taylor, L., Treacy, D.J., Weinstein, A., Wintersgill, M.C.: The Andes physics tutoring system: Five years of evaluations. In: McCalla, G.I., Looi, C.K. (eds.) *Artificial Intelligence in Education Conference*. IOS Press, Amsterdam (2005)

ITS in Ill-Defined Domains: Toward Hybrid Approaches

Philippe Fournier-Viger¹, Roger Nkambou¹,
Engelbert Mephu Nguifo², and André Mayers³

¹ Dept. of Computer Sciences, University of Quebec in Montreal, Canada
fournier_viger.philippe@courrier.ugam.ca, nkambou.roger@ugam.ca

² Dept. of Mathematics and Computer Sciences, Université Blaise-Pascal Clermont 2, France
mephu@isima.fr

³ Dept. of Computer Sciences, Université of Sherbrooke, Canada
andre.mayers@usherbrooke.ca

Abstract. Classical approaches for supporting tutoring services face several limitations for ill-defined domains. To overcome these limitations, we argue for the utilization of hybrid approaches for supporting tutoring services. In this paper, we describe a hybrid model that combines an expert system, the model-tracing paradigm, and a data mining approach in an ITS for learning to operate a robotic arm. The result is tutoring services that exceed what was possible to offer with each individual approach for this domain.

1 Introduction

Domains where classical approaches for building intelligent tutoring systems (ITS) are not applicable or do not work well have been termed "ill-defined domains" [1]. For these domains, classical approaches for supporting tutoring services face several limitations. An example is model-tracing [2], which consists in comparing a predefined task model with learners' solutions. Designing a task model by hand can be very hard and time-consuming for ill-defined domains. A second approach is constraint-based modelling [3]. Although, it is effective for some ill-defined domains, it cannot support tutoring services such as suggesting next problem-solving steps to learners, writing constraints can be difficult and time consuming, and a huge number of constraints can be required [1]. A third approach is to integrate an expert system in an ITS to generate expert solutions or for comparing learner solutions with ideal solutions [1]. Expert systems are appropriate for many ill-defined domains. But not all of them can explain their reasoning to learners. Therefore, we here argue for the use of hybrid approaches for ill-defined domains. The idea is to combine advantages of different approaches to avoid their limitations.

2 An Hybrid Model in CanadarmTutor

We illustrate this approach with CanadarmTutor [4], an ITS for learning to operate the Canadarm2 robotic arm, a 7 degrees of freedom, robotic arm, deployed on the International Space Station (ISS). The main learning activity in CanadarmTutor is to

move the arm from a given configuration to a goal configuration. It is a difficult task since operators do not have a direct view of the scene of operation on the space station and must rely on cameras mounted on the manipulator and at strategic places in the environment where it operates. To move the arm, an operator must select at every moment the best cameras for viewing the scene of operation, select and perform joint rotations for moving the arm, and avoid dangerous situations. The task of moving the arm is ill-defined because even if there are some general rules for moving the arm, there are no clear strategies for choosing joints rotations [6].

To support tutoring services in CanadarmTutor, we have initially applied the “expert system approach” by integrating a special path-planner which is based on a probabilistic roadmap approach [4]. The path-planner can automatically generate correct arm's moves avoiding obstacles, consistent with the best available camera views to achieve a given goal [4]. But, the generated paths are not always realistic or easy to follow, as they are not based on human experience, and they do not cover aspects of the task such as how to select/adjust cameras. Also, it cannot support tutoring services such as estimating knowledge of learners as there is no knowledge or skills representation.

To overcome these limitations, we applied the “model-tracing approach” [5]. To do so, we used a custom cognitive model, which is designed for taking into account aspects of spatial reasoning. With this cognitive model, we modelled the main steps for moving the arm as a set of rules with declarative knowledge. This modeling permits CanadarmTutor to automatically evaluate a learner's spatial representations and skills during arm manipulation, and generate personalized feed-back (see [5] for details). Although the task model specified by hand provided a fine cognitive assessment of a learner's knowledge for the main steps of the manipulation task, it does not go into finer details such as how to select joint rotations for moving Canadarm2. The reason is that at this level of details, it is very difficult to define a task model for generating the joint rotations that a human would execute.

To avoid this limitation, we developed a novel approach for supporting tutoring services for ITS [6]. It consists of recording user solutions for a task and then to apply a custom data mining algorithm for automatically extracting part of solutions that occur frequently. The idea is that even if there is a huge number of possibilities for a task and no clear strategies for finding solutions, there may be some parts of solutions that appear frequently and can be used for supporting tutoring services such as suggesting next problem-solving steps. We have applied this approach in CanadarmTutor and obtained what we call “partial task models”. This successfully allows CanadarmTutor to offer tutoring services such as suggesting joint rotations for moving the arm. Recently we have extended this approach by annotating user solutions with contextual information (skills required to perform the solution, success, expertise level, etc.), so that partial task models contain this information [6]. This is a very useful feature, as it allows for example to discover frequent parts of solutions that are common to experts possessing a particular skill, and that have completed the exercise successfully. This is used in CanadarmTutor to evaluate learner profiles based on the solution paths that they follow [6]. Although learning partial task models from user solutions in CanadarmTutor allows providing useful help to learners at the level of joint manipulations –which was impossible to achieve with the cognitive model or the path-planner [6], one problem is that no help can be offered to learners if part of a solution path was previously unexplored by other users.

We therefore took the decision to combine the three approaches in a hybrid model in CanadarmTutor. Because of space limitation, we only describe its main features, here. First, when a learner clicks on “what I should do next?” during an exercise, CanadarmTutor offers advices on the general procedure for moving the arm thanks to the cognitive model and suggest joint rotations based on the partial task models. If no help can be offered with the partial task model for the current task, a path is generated with the path planner and presented to the learner.

Second, the skills from the cognitive model are now used to annotate the recorded solutions taken as input by the data mining algorithm. Therefore partial task models now include skills from the cognitive model. CanadarmTutor uses this new information to assess skills of the cognitive model by looking at the solution path followed by the learner according to the partial task model. For example, if a learner follows several patterns demonstrated by people with a particular skill, CanadarmTutor will raise its confidence that the learner possesses the skill.

Other tutoring services offered thanks to the hybrid model are generating exercises tailored to the learner (cognitive model), generating demonstration (cognitive model + path planner + partial task models), offering proactive help about camera selection (cognitive model) and letting the learner explore freely the domain knowledge (cognitive model + partial task model). We have performed a preliminary experimentation with the hybrid model and users have been very satisfied.

In conclusion, the tutoring services now offered in CanadarmTutor with the hybrid approach greatly exceed what was possible to offer with each individual approach for the ill-defined task of the robotic arm manipulation. Note that we did not just put together tutoring services provided by the three approaches. Rather, we have integrated each one with each other to provide coherent and rich tutoring services.

Acknowledgments. We thank the Canadian Space Agency, NSERC and FQRNT for their logistic and financial support, and current/ past members of GDAC\PLANIART.

References

- [1] Fournier-Viger, P., Nkambou, R., Mephu Nguifo, E.: Supporting tutoring services in ill-defined domains. In: Nkambou, et al. (eds.) *Advances in intelligent tutoring systems*. Springer, Heidelberg (2010)
- [2] Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent Tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education* 8, 30–43 (1997)
- [3] Mitrovic, A., Mayo, M., Suraweera, P., Martin, B.: Constraint-based tutors: a success story. In: *Proc. of IEA AIE 2001*, pp. 931–940 (2001)
- [4] Kabanza, F., Nkambou, R., Belghith, K.: Path-Planning for Autonomous Training on Robot Manipulators in Space. In: *Proc. 19th Intern. Joint Conf. Artificial Intelligence* (2005)
- [5] Fournier-Viger, P., Nkambou, R., Mayers, A.: Evaluating Spatial Representations and Skills in a Simulator-Based Tutoring System. *IEEE Transactions on Learning Technologies* 1(1), 63–74 (2008)
- [6] Fournier-Viger, P., Nkambou, R., Mephu Nguifo, E.: Exploiting Partial Problem Spaces Learned from Users’ Interactions to Provide Key Tutoring Services in Procedural and Ill-Defined Domains. In: *Proc. AIED 2009*, pp. 383–390. IOS Press, Amsterdam (2009)

Analyzing Student Gaming with Bayesian Networks

Stephen Giguere¹, Joseph Beck¹, and Ryan Baker²

¹ Computer Science ² Social Science and Policy Studies
Worcester Polytechnic Institute
Worcester, MA, USA

Abstract. This paper examines the problem of modeling when students are engaged in “gaming the system.” We propose and partially validate an approach that uses a hidden Markov model, as is used in knowledge tracing, to estimate whether the student is gaming on the basis of observable actions. By doing so, we provide a common modeling approach that is applicable to gaming, or other constructs such as off task behavior. We find that our initial approach gave promising results, with parameter estimates that are plausible, and also exposed some weaknesses in our initial attempt. Specifically, that relying solely on response time is probably insufficient to construct a strong model of gaming.

1 Goal and Approach

This work’s goal is to adapt Bayesian knowledge tracing [1] to produce a similar Bayesian *gaming tracing* method that models student gaming [2] behavior. Knowledge tracing uses an observable action of a correct or incorrect response, and updates its knowledge of the student’s knowledge, which is not directly observable. Our model observes some visible indicator of the student’s gamingness, and updates its estimate of how likely the student is gaming the system. We learned a separate model of gaming for each of the thirty-five skills in the domain since gaming tendencies are related to the skill being practiced [3]. We used fast response times as our observable variable, and varied the threshold we considered “fast” as 2 seconds, 5 seconds, and 10 seconds.

We used data gathered by the Assistments system [4, 5], an online tutor designed to accommodate formative assessment of students in city schools (www.assistment.org). The tutor contains problems in middle school mathematics. These problems are organized by the various skills that they test, as determined by content experts working for Assistments. We had available over 1.75 million student responses to problems, but due to problems of execution time were restricted to using a random sample of 15% of the students (i.e. randomization and selection were at the student level). We fit the model to the data using BNT-SM [6], a Matlab package specifically designed for training knowledge tracing models. Since our models are structurally identical, and only add one additional parameter, using the BNT-SM simplified our analysis. Table 1 shows the results of our model-fitting process.

The first notable observation to make about these parameters is how as the threshold for fast responses become larger, the probability of a rapid response increases. Thus, we have a tradeoff of making our detector more sensitive by decreasing the

Table 1. Overview of model parameters separated by their definitions of a fast response

Model Parameter	Threshold for Fast Response		
	≤ 2 sec.	≤ 5 sec.	≤ 10 sec.
P(initially game this skill)	0.118	0.205	0.266
P(fast response nongaming)	0.009	0.022	0.072
P(fast response gaming)	0.658	0.647	0.479
P(become gamer)	0.031	0.084	0.116
P(stop gaming)	0.204	0.252	0.206

threshold at a cost of catching fewer gamers. Next, we notice that the probability of stopping gaming is substantial: approximately 0.2 for all thresholds. There are two things to note. First, this parameter does not exist in a knowledge tracing model, but is necessary to model a transient state such as gaming as it is this parameter that enables a student's level of gaming to naturally fall with time. Second, the consistency of the parameter across thresholds reinforces that our detector is measuring something real. Whether a student stops gaming should be relatively insensitive to the precise definition of gaming. We also note that the initial probability of gaming is fairly high: a 20% chance of a response of less than five seconds is not encouraging. However, at least as a trend, students are more likely to stop becoming gamers than they are to become gamers (last two rows).

The conditional probabilities (second and third model parameters in Table 1 reflect how the model is able to reason from a student's response time. Given that a student responds quickly, how much evidence is it of gaming behavior? The greater the difference between those two parameters, the more reliable our detector is. We conducted simulations and found that our model could fairly quickly determine if a student was gaming—at least if one believes our model. We did not conduct a validation against “gold standard” data labeled as gaming by, for example, human observers.

2 Future Work and Conclusions

This work has provided a proof of concept that gaming can be viewed in a similar statistical model as knowledge tracing (a hidden Markov model). It was necessary to tweak knowledge tracing to give students the ability to transition out of their gaming state; the analogy in knowledge tracing is “forgetting,” which does not occur in standard knowledge tracing (or most other student modeling approaches).

Moving forward from this study, there are many investigations that can be made to make our immediate findings more complete and others that further pursue the concepts presented in this analysis. We do not believe an immediate validation of accuracy at predicting gaming for this technique vs. other approaches would be beneficial for two reasons. First, this technique is a first attempt, and there are several possible optimizations (multiple observables, normalizing response times, etc.). Second, such an evaluation misses the major contribution of the work. By modeling gaming and knowledge under the same Bayesian framework, it becomes possible to simultaneously

model both constructs. Exploring such possibilities is a very rich area for future work, as it is plausible that by simultaneously estimating both gaming and knowledge, we can do a better job of both. As suggestive evidence, recent work [3] has generated useful results by putting a hand-crafted heuristic estimate of gaming in a Bayesian knowledge tracing framework.

Even if there is no synergistic result of combining the two, modeling gaming with a standard student modeling technique means that any research that strengthens knowledge tracing (such as a better parameter fitting approach) also strengthens our gaming detector. This positive result is in stark contrast to training a home-grown gaming detector. Unless the detector's designer pushes on it, it is unlikely to get better. We argue that by converging on a set of common models, the field will advance much more rapidly.

Acknowledgements

This research was made possible by the US Dept. of Education, Institute of Education Science, "Effective Mathematics Education Research" program grant #R305A070440, NSF CAREER award to Neil Heffernan, the Spencer Foundation, and a Weidenmeyer Fellowship from WPI.

References

1. Corbett, A., Anderson, J.: Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction* 4, 253–278 (1995)
2. Baker, R.S.J.d., et al.: Adapting to When Students Game an Intelligent Tutoring System. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) *ITS 2006*. LNCS, vol. 4053, pp. 392–401. Springer, Heidelberg (2006)
3. Gong, Y., et al.: The fine-grained impact of gaming (?) on learning in Intelligent Tutoring Systems (2010)
4. Razzaq, L., Heffernan, N.T.: Scaffolding vs. hints in the Assistment System. In: Ikeda, M., Ashley, H., Chan, T.-W. (eds.) *Proceedings of the 8th International Conference on Intelligent Tutoring Systems*, pp. 635–644. Springer, Berlin (2006)
5. Shrestha, P., et al.: Are Worked Examples an Effective Feedback Mechanism During Problem Solving? In: Taatgen, N.A., Rijn, H.v. (eds.) *Annual Conference of the Cognitive Science Society*, pp. 1294–1299 (2009)
6. Chang, K.-m., et al.: A Bayes Net Toolkit for Student Modeling in Intelligent Tutoring Systems. In: *Proceedings of the 8th International Conference on Intelligent Tutoring Systems*, Jhongli, Taiwan (2006)

EdiScenE: A System to Help the Design of Online Learning Activities

Patricia Gounon and Pascal Leroux

Computer Science Laboratory of the Le Mans University (LIUM)

Institut d'informatique Claude Chappe

Avenue René Laennec

72085 Le Mans cedex 9, France

{firstname.lastname}@univ-lemans.fr

Abstract. This paper deals with a system which helps the designer of on-line learning to specify pedagogical scenario and more particularly on the guidelines wished for a learning activity given. The first idea is to propose for a designer, who is neophyte or expert, an adapted assistance for the description of a pedagogical scenario from a model of tutoring organization. The second is to export the pedagogical scenario in the standardized form of his choice depending of training platform used. In this paper, we describe the cooperative system EdiScenE, based on Tutoring organisation model, which includes author and help tools for the development of pedagogical scenario.

Keywords: Cooperative System, Author Tools, Organization of Tutoring Activity, Pedagogical Scenario, Educational Modeling Language.

1 Introduction

The exchanges and reusable of pedagogical objects are important challenge in Technology Enhanced Learning (TEL). To obtain some reusable pedagogical components, a solution is to use an EML (Educational Modelling Language) for describing all pedagogical sequences can be expressed. We focus our researches on the authors' environments. These environments aim to program directly with the IMS-LD language to describe the different elements of a unit of learning like for example CopperAuthor [1] and Collage [2]. The main problem with these tools is the use of the IMS-LD which is not so easy to understand for designer. So, several environments propose a graphic interface to describe a pedagogical scenario like for example MOTPlus Editor [7]. Nevertheless, the existing authoring tools are often difficult to access for novice users and they don't offer adequate assistance to the user profile. Furthermore, when we study more precisely these environments, we observe that the description of the organization's tutoring activity is not detailed and does not describe in great detail the roles of actors, to make and determine the tools best suited to its tutoring activity.

2 Adaptation of Two EMLs Integrating the Triton Model

The Triton model is based on different work in Technology for Human Learning, Educational Sciences and of the results of exploratory experiments that we conducted. The main goal of **Triton** model (**Tutoring organization** model) [4] is to allow a description of the organization of the tutoring of a learner during a learning session. This model assists and guides the tutoring activity description in association with the learning activity. Triton model is used in the design and implementation of online media activity. The model becomes the basis for (1) reflection upon the desired accompaniment in a given learning activity and (2) the description of the tutoring organization in an e-learning platform.

We considered the reification of the Triton model in several EML, by (1) a more precisely granularity of actors' description, (2) the description of the interactions between the actors, and (3) the description of tutor recommendations and intervention tools associated. To prove the compatibility between the Triton model and the EMLs formalism and the interest of Triton model to improve the EML for the description of tutoring organization, we have chosen to integrate the Triton model components in the IMS LD [6] and Learning Design Language (LDL) [3]. Our propositions [5] contribute to enrich the accompaniment scenario already present in these standards. These adaptations aim to add a level of detail to the participating tutor and tutored person's description.

3 EdiScenE Prototype for Describing a Pedagogical Scenario

EdiScenE is a cooperative system which helps a designer of on-line learning to specify pedagogical scenario and more particularly on the guidelines wished for a learning activity given. This system must allow describing and verifying the coherence of the scenario in accordance with the choice of the EML. Our approach is to offer a system without prerequisites to a standard formalism. Another goal is the cooperation of the system with the user concerning the omissions or mistakes in the pedagogical scenario. We want to focus on methodological elements to construct a pedagogical scenario in terms of learning activity and the interactions between learners and tutors. The designer can choose an EML. Now, two EMLs are offered: IMS-LD and LDL. Once the user selects the EML, the designer can create a pedagogical scenario or import one to complete it. To complete tutoring components of pedagogical scenario, the user has a component allows to identify the list of actors who incarnating the tutor function and tutored persons. The designer also describes the recommendations of each tutor previously identified. Specifically, from the actors' description and the interaction style definition of each tutor, the designer can consult, modify or define the recommendation of the tutor (as defined in the Triton model). The designer can also describe the tools used to support tutoring. Before the construction or the modification of a pedagogical scenario, the designer describes some tutor tools by using the Triton model formalism. This description gives the characteristics of different tools that could be used during the tool selection step for supporting the tutor activity. All these tool descriptions are reusable for different learning designs in several pedagogical contexts. A list of tools, adapted to the different tutoring task characteristics, is

proposed to the designer. This list is obtained by a comparison of each characteristic of the specified task with each tool described beforehand using the same formalism. It should be noted that the system also assists the user on the consistency of the contents of each module. Moreover, at any time, the user can request assistance. EdiScenE aims to generate the described pedagogical scenario in accordance with the choice of the formalism recapitulating the organization of the tutoring and the learning activity description. This result could be integrated to a learning platform supporting the formalism chosen.

4 Conclusions and Perspectives

From the Triton model and the extended several EMLs proposition, we have conceived and developed the prototype EdiScenE, dedicated to a designer for describing a pedagogical scenario and specifying tools to support tutor activity during a learning session. This prototype is interesting because it aims to describe a pedagogical scenario without prerequisites to a standard formalism and it allows give some methodologies elements to construct a pedagogical scenario in terms of learning situation and the interactions between learners and tutors. One perspective of this research is to improve the EdiScenE prototype to provide to the designer with the possibility to add one or several formalism extensions in accordance with his needs. Another perspective is to improve cooperation between the user and the system from experiments conducted with novice and experts users.

References

1. CopperAuthor project site, <http://www.copperauthor.org/>
2. Hernández-Leo, D., Villasclaras-Fernández, E.D., Asensio-Pérez, J.I., Dimitriadis, Y., Jorín-Abellán, I.M., Ruiz-Requies, I., Rubia-Avi, B.: COLLAGE: A collaborative Learning Design editor based on patterns. *Educational Technology & Society* 9(1), 58–71 (2006)
3. Ferraris, C., Martel, C., Vignollet, L.: Modelling the “Planet Game” Case Study with LDL and Implementing it with LDI. *Journal of Interactive Media in Education*
4. Gounon, P., Dubourg, X.: A Descriptive model to Organise Tutoring for Learning Environments. In: IEEE Computer Society (ed.) ICALT 2004, Joensuu (Finlande), August 1-September 30 (2004)
5. Gounon, P., Leroux, P. (to appear) Design of Tutoring Activity: An Extension of two EMLs Based on an Organizational Model of Tutoring. In: IEEE Computer Society (ed.) ICALT 2010, Sousse (Tunisia), July 5-7 (2010)
6. IMSLD, Technology Standards Committee of the IEEE. IMS Learning Design v1.0. IMS Learning Design Information Model. Version 1.0 Final Specification (2003)
7. Paquette, G., Léonard, M.: MOT+LD Graphic Editor Workshop. In: UNFOLD Workshop, Barcelone, Espagne (avril 2005)

Critiquing Media Reports with Flawed Scientific Findings: *Operation ARIES!* A Game with Animated Agents and Natural Language Trialogues

Art Graesser¹, Anne Britt², Keith Millis², Patty Wallace²,
Diane Halpern³, Zhiqiang Cai¹, Kris Kopp², and Carol Forsyth¹

¹ Institute for Intelligent Systems, 365 Innovation Drive, University of Memphis,
Memphis, Memphis, TN 38152, USA

{a-graesser, zcai, cmfrsyth}@memphis.edu

² Department of Psychology, Northern Illinois University, DeKalb, IL 60115, USA

{kmillis, britt, pwallace}@niu.edu, kkopper@yahoo.com

³ Department of Psychology, Claremont McKenna College, 850 Columbia Avenue,
Claremont, CA 91711 USA

diane.halpern@claremontmckenna.edu

Abstract. *Operation Aries!* is a computer environment that helps students learn about scientific methods and inquiry. The system has several components designed to optimize learning and motivation, such as game features, animated agents, natural language communication, trialogues among agents, an eBook, multimedia, and formative assessment. The present focus is on a Case Study learning module that involves critiquing reports of scientific findings in news media that have flawed scientific methodology. After the human student lists the methodological flaws of a Case Study in natural language, a teacher agent and a peer agent hold a trialogue with the student that evaluates each listed flaw and that uncovers additional flaws that that student missed.

Keywords: Natural language processing, pedagogical agents, case-based reasoning, games.

1 Introduction

Case-based architectures are periodically adopted in the design of intelligent learning environments and models of cognition [1,2,3,4,5]. Specific *cases* (e.g., experiences, scenarios, authentic problems) provide a rich context and set of constraints that guide reasoning, problem solving, inferences, interpretations, and other cognitive processes. Memory can be viewed as a rich storehouse of specific cases that are retrieved as needed when a person encounters new tasks and problems.

One powerful method of demonstrating one's knowledge is to apply general principles in a knowledge domain to specific cases. This application of principles to cases is the focus of the present study. The knowledge domain is research methods whereas there are general principles of research design, statistics, and scientific reasoning. Students both learn and demonstrate their understanding scientific methods by

critiquing specific studies in the news that exhibit flawed scientific reasoning. This Case Studies module is one of many features of the learning environment we have developed, called *Operation ARIES!*.

2 Operation Aries!

Operation Aries! (Acquiring Research Investigative and Evaluative Skills, hereafter called Aries) is an educational game designed to teach scientific methods and inquiry to high school and college students. A game environment with fantasy and feedback was selected because it improves motivation and time-on-task for deeper learning [6]. Aries teaches several concepts that are required for evaluating whether research designs and findings support conclusions about causal relationships between variables (e.g., the need for control groups, experimenter bias, third variables) and other virtues of scientific methodology (e.g., validity, reliability). Aries has several components designed to optimize learning and motivation in addition to game features, such as animated agents, natural language communication, dialogues among agents, an eBook, multimedia, and formative assessment.

Aries has a game narrative that involves aliens (called Fuaths) who are disguised as human beings. The Fuaths disseminate bad science through various media (newspapers, advertisements, Internet) in an attempt to spread illiteracy and take over the planet. The goal of the student is to become a special agent of the Federal Bureau of Science (FBS), an agency with a mission to identify the Fuaths and save the planet. Aries has three major modules. A *Science Training* module is an e-Book on the scientific method, based on an academic book written by Halpern [7]. There is an on-line assessment of the chapters through multiple-choice tests and conversational dialogues among the human student, a teacher agent, and a peer student agent [8]. These dialogues attempt to comprehend the language of the student and adaptively respond with algorithms based on AutoTutor [9]. An *Interrogation Module* involves the human student asking questions that interrogates a suspect that might be an alien [10]. A *Case Studies* module presents flawed studies for the human to critique.

3 Case Studies Module

Each case study is a report on a scientific study presented by a newspaper, advertisement, Internet, or some other form of media. Each case study has between 0 and 4 flaws in the scientific methodology. The potential flaws are in different categories that address hypotheses, independent and dependent variables, control, sampling of observations, experimenter bias, and justifiable conclusions. Altogether, a flaw in a case study could potentially involve a violation in one of 20 scientific principles. An example case study involved an advertisement for a new diet pill that causes people to lose weight. The study was flawed because there was no control group and there was attrition of subjects in a single experimental condition. When considering all of the problems in the case repository, all of the scientific principles have associated cases with potential flaws. Consequently, case studies can be dynamically selected to handle principles that a particular student is having trouble with, based on past performance; alternatively, students can select cases to give them a sense of control.

When a particular case is selected by either the human or Aries, the student reads the case and then types in a list of flaws in natural language. After the student finishes listing the flaws, there is a dialogue among the human, a teacher agent, and the peer agent that evaluates each flaw, one at a time. The teacher agent indicates whether it is an actual flaw, explains why, asks clarification questions when it cannot understand the student's language, and expresses other speech acts. The teacher agent also gives hints to elicit other flaws that the human does not initially list. On some trials, the peer agent lists the flaws and the teacher agent responds. The student accumulates points in the game to the extent that the student can express the relevant flaws without hints and resists expressing critiques that are not bona fide flaws.

The Aries system has all of the modules completed. We are in the process of alpha testing the system on pilot students. The impact of the various Aries components will be evaluated with respect to motivation, learning gains, and verbal protocols.

Acknowledgments. This research was supported by the Institute of Education Sciences (R305B070349). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of IES.

References

1. Aleven, V.: Using Background Knowledge in Case-based Legal Reasoning: A Computational Model and an Intelligent Learning Environment. *A.I.* 150, 183–237 (2003)
2. Ashley, K.D., Brüninghaus, S.: Automatically Classifying Case Texts and Predicting Outcomes. *Artificial Intelligence and Law* 17, 125–165 (2009)
3. Kolodner, J., Cox, M., Gonzalez-Calero, P.: Case-based Reasoning-inspired Approaches to Education. *The Knowledge Engineering Review* 20, 299–303 (2005)
4. Leake, D.: CBR in Context: The Present and Future. In: *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, pp. 3–30. AAAI Press/MIT Press, Menlo Park (1996)
5. Schank, R.C.: *Dynamic Memory Revisited*. Cambridge University Press, Cambridge (2009)
6. Parker, L.E., Lepper, M.R.: Effects of Fantasy Contexts on Children's Learning and Motivation: Making Learning More Fun. *Jour. of Pers. and Social Psych.* 62, 625–633 (1992)
7. Halpern, D.F.: *Thought & Knowledge: An Introduction to Critical Thinking*. Erlbaum, Mahwah (2003)
8. Cai, Z., Graesser, A.C., Millis, K.K., Halpern, D., Wallace, P., Moldovan, C., For-syth, C.: ARIES!: An Intelligent Tutoring System Assisted by Conversational Agents. In: *Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling*, p. 796. IOS Press, Amsterdam (2009)
9. Graesser, A., Chipman, P., Haynes, B., Olney, A.: AutoTutor: An Intelligent Tutoring System with Mixed-initiative Dialogue. *IEEE Transactions on Education* 48, 612–618 (2005)
10. Millis, K., Cai, Z., Graesser, A., Halpern, D., Wallace, P.: Learning Scientific Inquiry by Asking Questions in an Educational Game. In: *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, pp. 2951–2956. AACE, Chesapeake (2009)

A Case-Based Reasoning Approach to Provide Adaptive Feedback in Microworlds*

Sergio Gutierrez-Santos, Mihaela Cocea, and George Magoulas**

Birkbeck College, London Knowledge Lab, University of London
23-29 Emerald St, WC1N 3QS, London (UK)
{sergut,mihaela,gmagoulas}@dcs.bbk.ac.uk

Abstract. This paper presents a case-based reasoning (CBR) approach to provide adaptive support in microworlds. Interaction in microworlds is complex and unstructured, making the analysis of student behaviour difficult and the provision of computer-based feedback challenging. Our approach starts with the elicitation of expected solutions to microworld tasks (both valid and common mistakes) to generate a case base. This is used to evaluate the actions of students and provide adapted feedback.

Keywords: case-based reasoning, exploratory learning environments, microworlds, adaptive feedback.

1 Introduction

Microworlds are a special kind of exploratory learning environments (ELE) where students are allowed to create their own models/constructions, and explore their properties and relationships. Providing appropriate support in such a situation is crucial for adequate learning [1], but this is an challenging endeavour due to the ill-definedness of the interaction. This paper proposes an approach to provide support for a microworld called eXpresser. The goal is to provide adaptive feedback on-demand to alleviate the workload of teachers in classrooms.

Most cases of analysis and support for ELE are related to systems for learning Physics [2,3,4]. There are also relevant works in the domain of Mathematics [5,6]. Although these systems grant some freedom to students, none of them allow to create new models/constructions. Our approach uses case-based reasoning to provide feedback to students interacting with a mathematical microworld that allows them to create new shapes and algebraic expressions from scratch.

2 The eXpresser Microworld

The eXpresser is a microworld that allows students to create figural patterns and link them with expressions. Creating and combining patterns and expressions, students can create many different structures in the computer [7]. The

* The authors would like to acknowledge the rest of the members of the MiGen team.

** This work is funded by TLRP (e-Learning Phase-II, RES-139-25-0381).

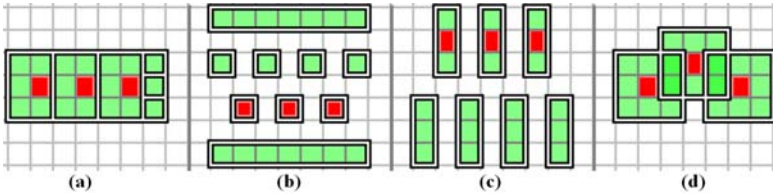


Fig. 1. Several examples of expected solutions for a 'footpath' task. Each solution can be constructed in many different ways (i.e. different actions lead to similar structures). The internal structure of the patterns is highlighted here for clarity. In eXpresser all constructions would look the same in the normal course of the task.

construction of these structures in the context of a classroom task scaffolds the development of algebraic skills of the students (our target age is 11–12 years).

The microworld grants a lot of freedom to students, who may construct their patterns in a multitude of ways, that range from valid ones (see Figure 1) to off-task behaviour. The complexity and variety of possible approaches makes it impossible to list all of them. Our approach is based on the identification of the main possibilities with the help of pedagogy experts, using this initial case base to judge the specific approaches of students and provide adapted feedback.

3 CBR for Adapted Feedback Generation

The most important containers of information in a CBR system are the cases themselves, each of them storing information about one *problem* and its *solution*. The other knowledge containers are the similarity measures (used to compare cases) and the adaptation mechanism (used to adapt solutions to a new problem). In our approach, *problems* are possible construction strategies on the microworld and *solutions* are feedback provided to learners on demand (see Figure 2). Construction strategies are represented as series of shapes. Shapes are defined by attributes such as position, colour, and relations to other shapes (e.g. there are as many green tiles as five times the number of red tiles).

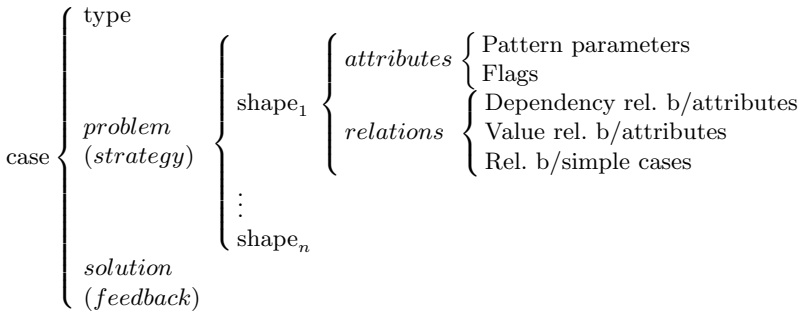


Fig. 2. Case structure. For a more detailed description of strategies and shapes, see [8].

When a learner working with the microworld asks for help, the learner's construction is processed into a new *problem*, i.e. strategy. This strategy is compared with all the strategies in the case base to find the most similar one. Details about the similarity metrics used can be found in [8].

If a perfect match is found, feedback is immediately generated as the *solution* of the matching case. Otherwise, the case solution will need to be adapted to generate the feedback. There are two types of cases in our approach, so there are two different adaptation processes.

If the retrieved case is an "expected solution" (i.e. the student is probably working in the right direction), the text of the feedback changes to give the student an encouraging message and additional information is put in place to highlight those aspects on which the student should reflect upon. This information can be extracted from the similarity comparison between the case and the student's result. If the retrieved case can be a "common mistake" (i.e. the student has a misconception frequently observed in practice according to pedagogy experts) the feedback provided contains a common message developed by the pedagogical team. This message makes the students reflect on past actions and realise their misconception. In this case, there is usually no further adaptation of the feedback because it is already specifically targeted towards one well-known misconception. Lastly, it can happen that the student's construction is not similar to any of the problems in the case base (e.g. the student has made little progress on the task, or she might have found a new perfectly valid approach, which had not been considered by the design team). This situation is beyond the scope of the system, so a message is handled to the human teacher.

The proposed approach makes it possible to provide adapted feedback in microworlds, where the unstructured nature of the interaction poses an important challenge. We plan to extend the approach to other microworlds.

References

1. Kirschner, P., Sweller, J., Clark, R.: Why minimal guidance during instruction does not work: an analysis of the failure of constructivist, discovery, problem-based, experiential, & inquiry-based teaching. *Educational Psychologist* 41, 75–86 (2006)
2. Van Lehn, K., Lynch, C., Schultz, K., Shapiro, J., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., Wintersgill, M.: The ANDES physics tutoring system: Lessons learned. *Int. Journal of Artificial Intelligence and Education* 15 (2005)
3. Veermans, K., van Joolingen, W.R.: Combining heuristics and formal methods in a tool for supporting simulation-based discovery learning. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) *ITS 2004*. LNCS, vol. 3220, pp. 217–226. Springer, Heidelberg (2004)
4. Stathacopoulou, R., Magoulas, G., Grigoriadou, M., Samarakou, M.: Neuro-fuzzy knowledge processing in intelligent learning environments for improved student diagnosis. *Information Sciences* 170, 273–307 (2005)
5. Bunt, A., Conati, C.: Probabilistic student modelling to improve exploratory behaviour. *User Modeling and User-Adapted Interaction* 13, 269–309 (2003)

6. Mavrikis, M., Lee, J.: Towards contingent and affective microworlds. In: Int. Conf. in Artificial Intelligent in Education (2003)
7. Pearce, D., Geraniou, E., Mavrikis, M., Gutierrez-Santos, S., Kahn, K.: Using pattern construction and analysis in an exploratory learning environment for understanding mathematical generalisation: The potential for intelligent support. In: Int. Workshop on Intelligent Support for Exploratory Environments (2008)
8. Cocea, M., Gutierrez-Santos, S., Magoulas, G.: Enhancing modelling of users strategies in exploratory learning through case-base maintenance. In: Workshop on Case-based Reasoning, SGAI Int. Conf. on Artificial Intelligence (2009)

Real-Time Control of a Remote Virtual Tutor Using Minimal Pen-Gestures

Yonca Haciahmetoglu and Francis Quek

Center for Human-Computer Interaction - Virginia Tech, USA
{yoncah, quek}@vt.edu

Abstract. We present a distance tutoring system that allows a tutor to provide instruction via an animated avatar. The system captures pen-gestures of real tutor, generates 3D behaviors automatically, and animates a virtual tutor on the remote side in near real-time. The uniqueness of the system comes from the pen-gesture interface. We have done a study to test this interface. The system can effectively recognize and animate different types of gestures. Gesturing on the tablet and gesturing on the board were then compared. The results show that users can easily adopt to tablet, and pen-gesture on the tablet naturally. They were able to use pen-tablet interface effectively after a short instructional period.

Keywords: distance tutoring, virtual tutor, pen-based interaction, gesture generation, embodied interaction.

1 Introduction

In our distance tutoring system, our animated tutor serves as a remote surrogate for a human tutor. The virtual tutor is controlled via a minimal pen-based interface and constructs coherent psycholinguistically-correct 3D gesture and gaze in conjunction with speech at the student site. In the highly spatial and contextually rich interaction between tutor and student, a key to facilitating learning is to provide a sense of situatedness between the tutor and student. This situatedness is both temporal and spatial. Spatial situatedness is essential for deictic references into both the interlocutors space and any artifact of interaction. This is especially critical in disciplines of study such as engineering and science where space, diagrams and graphical representations are of crucial importance. This situatedness is not fully met in video conferencing.

The temporal relationship between gesture and gaze entities with their lexical affiliates in speech provide information on the intensive focus of the tutorial discourse. We believe that these are important in assisting the student to grasp the substance of the tutorial. If the student had access only to a disembodied flying cursor, she will lose track of it the moment her attention is not fixed on the cursor. She will then have to search for the deictic point, expanding valuable cognitive and attentional resources. Having access to the virtual tutors body will help in directing the students attention and cueing her to the spatial presentation of the material. Our system extracts elements of communicative intent through a pen-based interface and constructs the desired fully embodied behavior automatically.

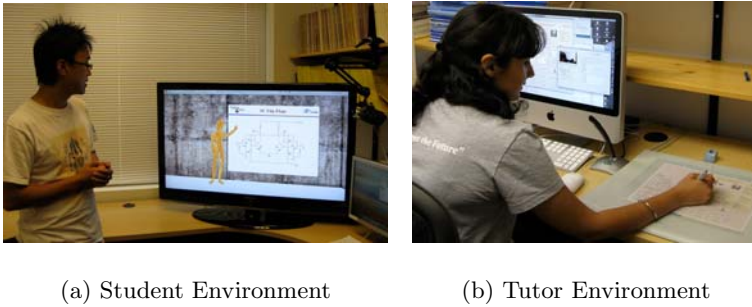


Fig. 1. (a) Student Environment consists of a TV and a camera to track the student. (b) In Tutor Environment, the human tutor interacts with the student at a remote site via a pressure-sensitive tablet and a microphone.

Current interfaces for controlling avatar permit only a small range of behavior. Lee et al. [5] showed 3 different ways of control, however the concentration is on motion generation, not on communicative behavior. Barrientos and Canny used pen gestures to control an avatar [1]. A letter is translated into a body gesture. Behavior is produced via a symbolic mapping, i.e. user has to think about what the avatar has to do, and translate/map this into a symbol, and perform the symbol. We believe that this interferes with speech and gesture synchrony. Our approach differs from canned text-to-speech avatars [3] that use a variety of markup languages [4] in that we facilitate vivid interaction using the tutors own voice. It is different than animated agents [2] since it is human-driven.

2 System Design and Experiment

The system consists of two environments (Figure 1). Pen-gestures are transmitted to student environment and converted into full-bodied gestures in four phases. 1) We analyzed commonly used gestures while tutoring., 2) We generated behavior library., 3) We designed a unique pen-gesture interface to capture the intent of the real tutor., 4) We generate the motion and animate virtual tutor. A within subject study comparing gesturing while using the tablet and gesturing while tutoring on the board was conducted to compare the amount of gestures they performed, to determine how much of the natural gestures are transferred to the pen-tablet and how much of the pen-gestures are recognized correctly. Six volunteers (3 Female, 3 Male) participated.

Results. Participants rated their preference of the pen-tablet and its ease of use. How comfortable did they feel using the pen-tablet. They rated 4.33 (1: not at all comfortable, 5: very comfortable). The other ratings were: Easy to use (mean=4.83), easy to learn (mean=5), was able to explain my topic effectively using tablet (mean=4.5), no difficulty in talking while using the tablet (mean=4.67). They also compared with the board. They rated: talking while using the tablet was as easy as talking while talking on the board (mean=4.17; 1: strongly disagree, 5: strongly agree). Those results confirmed that they all find

the tablet easy to use. We believe that having a tablet and the tutoring material in front of them encouraged them to use it more than just talking or showing on the board remotely. Our pen-tablet interaction is as effective as the natural tutoring. Users find it easy to use.

Discussion. Pen-gesture interfaces have been previously used to control avatars. The limitations due to inability to naturally map the letters to the gestures shown in a previous work [1] were successfully overcome with the our more natural mapping which also preserves timing. The results show that users can easily adopt to tablet, and pen-gesture on the tablet naturally. They were able to use pen-tablet interface effectively after a short instructional period. Interacting through pen-gestures caused no significant reduction in the amount of gestures they performed. The number of participants were low but the results were consistent among them and made it clear that people can gesture on the tablet. In fact they do gesture on the tablet more than on the board. This suggests that the pointing gestures on the board are transferable to the tablet. All the real tutor does is to talk to the pen-tablet and gesture with pen. The system transfers natural pen-gestures into fully embodied and timely synchronized behavior. Using such a system would enable virtual tutors to be used more frequently. We hope that it will inspire others to design gestural interfaces where pen-gestures are used without extra complex mappings.

3 Conclusion and Future Work

Generating believable behavior for the real-time controlled virtual tutor remains to be a big challenge and requires understanding of behavior generation in humans. In this paper, we have demonstrated that it is possible to generate fully embodied behavior to drive a virtual tutor via a minimal pen-based interface. After validating the ease of use of pen-gesture interface, we will assess the student side of the system in our next study. We will look at the effects of using interactive virtual tutor on students' learning. We want to measure how embodied virtual tutor affects the ability to comprehend the information presented.

References

1. Barrientos, F.A., Canny, J.F.: Cursive: Controlling expressive avatar gesture using pen gesture. In: Collaborative Virtual Environments, Germany, pp. 113–119 (2002)
2. Bickmore, T., Cassell, J.: Social dialogue with embodied conversational agents. In: van Kuppevelt, J., Dybkjaer, L., Bernsen, N. (eds.) *Advances in Natural, Multimodal Dialogue Systems*. Kluwer Academic, New York (2005)
3. Jung, Y., Behr, J.: Extending h-anim and x3d for advanced animation control. In: *Web3D 2008*, pp. 57–65. ACM, New York (2008)
4. Kshirsagar, S., Magnenat-Thalmann, N., Guye-Vuillème, A., Thalmann, D., Kamyab, K., Mamdani, E.: Avatar markup language. In: Müller, W.S.S. (ed.) *Eighth Eurographics Workshop on Virtual Environments*, Barcelona, Spain, pp. 169–177 (2002)
5. Lee, J., Chai, J., Reitsma, P.S.A., Hodgins, J.K., Pollard, N.S.: Interactive control of avatars animated with human motion data. *ACM Trans. Graph.* 21(3), 491–500 (2002)

Theoretical Model for Interplay between Some Learning Situations and Brainwaves

Alicia Heraz and Claude Frasson

HERON Lab, Computer Science Department, University of Montréal,
CP6128 succ. Centre Ville, Montréal, QC, H3T-1J4, Canada
{herazali, frasson}@iro.umontreal.ca

Abstract. There is interplay between brainwaves and learning. To describe and understand part of this complex interaction, this paper proffers a new learner model called the LBD Model (or Learning and Brainwaves Dominances Model). Twenty-three participants were recruited to validate this Model. Results show distinct instances of the LBD Model regarding three situations of learning: positive learning, unconscious learning and unlearning.

Keywords: Intelligent Tutoring System, Brainwaves, Learner Model, Learning Situations.

1 Introduction

The Learner Model is one of the most important components within an Intelligent Tutoring System (ITS). To get information about the learner, data collection methods have evolved from self report [1] to facial expression analysis [6], posture's, gestures' and voice's interpretations [2] to biofeedback measurements [3], [4], [5]. Recent approaches combine different kinds of information channels to increase the prediction of the emotional and cognitive learner states. In the field of biofeedback measurements, many tracks remain unexplored especially those related to EEG.

2 Defining the LBD Model

We call Learning and Brain Dominances (LBD) Model the Learner Model that represents the relation between some Learning Situations (PL: Positive Learning, CL: Unconscious Learning and UL: Unlearning) and the Brainwaves' Dominances (Delta, Theta, Alpha, Beta1, Beta2 and Beta3). It is represented in Figure 1. Each ring represents the dominance order and contains the percentage of dominance of each wave during a learning session. The external ring shows the distribution of the dominances' percentage between the six waves as a first dominant wave and the internal ring shows the distribution of the dominances' percentage between the six waves as a last dominant wave during a learning session. The symbols of the three situations of learning PL (Square), CL (Disc) and UL (Triangle) can be associated to some parts of the six rings to describe the illustrate the relation between the variables BDO and LSC.

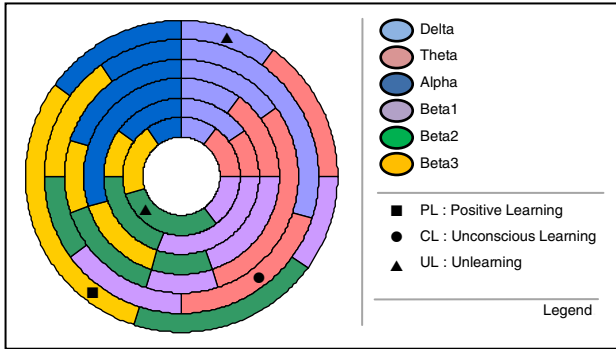


Fig. 1. The LBD Model

3 Experiment and Preliminary Results

To validate the model, we conducted an experiment, collected data and analyzed the instances obtained of the LBD Model. We recruited 23 participants all from university of Montreal. To measure the learners’ brainwaves, we use Pendant EEG, a portable wireless EEG. Pendant EEG sends the electrical signals to the computer via an infra-red connection. Light and easy to carry, it is not cumbersome and can easily be forgotten within a few minutes. The learner wearing Pendant EEG is completely free

Table 1. Percentages of the Brainwaves’ Dominances

Wave	LSC	Percentages of Dominance (%) (Dom ₁ , Dom ₂ , Dom ₃ , Dom ₄ , Dom ₅ , Dom ₆)
Delta	■ (PL)	(07.27, 09.63, 15.38, 20.56 , 24.15 , 30.98)
	● (CL)	(09.02, 15.58, 15.16, 18.76, 21.06, 27.84)
	▲ (UL)	(09.01, 11.12, 15.78, 20.65 , 24.62 , 28.18)
Theta	■ (PL)	(14.86, 18.56, 20.74 , 16.55, 19.63, 15.43)
	● (CL)	(15.75, 15.94, 20.60, 20.27 , 18.44, 14.93)
	▲ (UL)	(17.22, 17.81, 19.05 , 18.76, 17.28, 15.64)
Alpha	■ (PL)	(15.27, 16.54, 18.30, 18.12, 18.26, 14.83)
	● (CL)	(20.04, 19.19, 15.60, 15.09, 15.13, 18.01)
	▲ (UL)	(18.25, 19.03, 16.63, 15.64, 17.65, 14.86)
Wave	LSC	Percentages of Dominance (%) (Dom ₁ , Dom ₂ , Dom ₃ , Dom ₄ , Dom ₅ , Dom ₆)
Beta1	■ (PL)	(08.00, 10.27, 12.18, 19.00, 20.56, 26.74)
	● (CL)	(07.58, 09.70, 11.86, 17.08, 22.15 , 28.35)
	▲ (UL)	(05.83, 11.58, 14.22, 16.17, 22.55, 27.44)
Beta2	■ (PL)	(29.15 , 22.26, 16.00, 13.50, 09.34, 05.97)
	● (CL)	(25.77 , 18.51, 22.61 , 12.25, 12.27, 03.58)
	▲ (UL)	(22.91, 20.34 , 17.34, 16.46, 09.23, 08.75)
Beta3	■ (PL)	(25.45, 22.73 , 17.40, 12.27, 08.06, 06.05)
	● (CL)	(21.84, 21.07 , 14.18, 16.55, 10.95, 07.29)
	▲ (UL)	(26.78 , 20.12, 16.98, 12.32, 08.67, 05.13)

of his movements. One day before the experiment, participants were asked to read a set of 7 articles carefully. They were only allowed to read the articles one time. On the day of the experiment, we recorded the brainwaves of each participant when they were answering 35 questions related to the 7 texts they read before. When asked a question, participants indicate if they knew the answer or not. In both cases, the answer could be true or false. Thus we can calculate the variable LSC. The test duration's varies from 15 to 20 minutes for each participant. Table 1 gives the distribution of all the percentages of dominance for each brainwave at each level and according to each of the 3 learning states: PL, CL and UL.

We observe that Alpha Brainwave is more frequent in the first and the second levels for the state CL than in the 2 other states. Also Theta brainwaves and Delta brainwaves are more dominant respectively in the third and the second levels for the state CL than in the 2 other states. In addition, Beta2 Brainwaves are more frequent in the state CL for the first three levels than in the others two other states.

4 Discussion

The LBD Model for the PL State is characterized by the fact that Beta2 Brainwave dominates at the first level. The LBD Model for the CL State is characterized by the fact that Beta2 Brainwave dominates at the first and the third level and Beta1 dominates at the two last levels. The LBD Model for the UL State is characterized by the fact that Beta3 dominates at the first level. These results show that we could enrich the learner model within an ITS by the LBD Model in order to better recognize the student learning states and adapt a pedagogical strategies.

Acknowledgments. We would like to acknowledge the FQRSC and the CRSNG for funding this work.

References

1. Anderson, J.R.: Tailoring Assessment to Study Student Learning Styles. American Association for Higher Education (2001)
2. D'Mello, S.K., Graesser, A.C.: Automatic Detection of Learner's Affect from Gross Body Language. *Applied Artificial Intelligence* 23, 123–150 (2009)
3. Heraz, A., Daouda, T., Frasson, C.: Decision Tree for Tracking Learner's Emotional State predicted from his electrical brain activity. In: ICITS 2008, Montréal, Canada (2008)
4. Heraz, A., Razaki, R., Frasson, C.: Using machine learning to predict learner emotional state from brainwaves. In: ICALT 2007, Niigata, Japan (2007)
5. Heraz, A., Frasson, C.: Predicting the Three Major Dimensions of the Learner's Emotions from Brainwaves. *International Journal of Computer Science* (2007)
6. Nkambou, R., Héritier, V.: Facial expression analysis for emotion recognition in ITS. In: ITS 2004 workshop on Emotional Intelligence Proceedings (2004)

Cultural Adaptation of Pedagogical Resources within Intelligent Tutorial Systems

Franck Hervé Mpondo Eboa, François Courtemanche, and Esma Aïmeur

Department of Computer Science and Operations Research
University of Montreal, Quebec, Canada
{mpondoef,courtemf,aimeur}@iro.umontreal.ca

Abstract. Intelligent Tutoring Systems (ITS) are increasingly used for distance learning around the world. However, most systems present the learning content regardless of the learner’s cultural background. This paper presents a resource personalization technique for cultural adaptation within ITSs. The approach is based on a collaborative filtering technique using an implicit cultural profile, which is automatically updated using the learner’s interactions with the system.

Keywords: Cultural adaptation, collaborative filtering, user interaction.

1 Introduction

Despite many great successes, most intelligent tutoring systems for distance learning have an important limitation restricting worldwide scale use: the lack of pedagogical adaptation to the learner’s *socio-cultural context* [3]. Specifically, the learning content is displayed to different learners regardless of their cultural environment. However, several researches [1, 2, 4] have highlighted the fact that our mental programming – how we act, think, learn and interpret – is conditioned by our social circle, the countries in which we grew up, etc. For instance, the word “football” has different meanings in North America (American football) than in Europe or South America (soccer). As the creation of homogenous cultural groups for cultural classification of learners requires an important and tedious survey process [3], we propose a cultural adaptation approach that bypasses this difficulty using a two-step technique: a) A minimal amount of information about the learner is recovered to initialize the adaptation process b) A *collaborative filtering* technique is used to adapt pedagogical resources using the learner’s cultural profile, which is dynamically updated by his/her interactions with the system.

2 Adaptation Technique

In our ITS, each *problem* consists of a web page presenting a given topic. The domain knowledge included in a problem is presented using a series of *concepts*. In a problem a concept can be presented to the learner using different materials (images, texts or videos). We name *pedagogical resource* a particular material used to present a

concept. For example, in a problem about nutrition the concept of “Carbohydrate-rich foods” can be presented as an image of *maple syrup* (Canada), *bok choy* (China), *potatoes* (US) or *cassava* (Africa).

2.1 Knowledge Based Adaptation

Each pedagogical resource is tagged by a domain expert (arrow 1, Fig.1) in order to establish its relevance according to different nationalities. When logging for the first time, the learner provides his/her nationality in order to initialize a temporary cultural profile used to choose pedagogical resources at the beginning, effectively reducing the *cold start* effect (arrow 2, Fig.1).

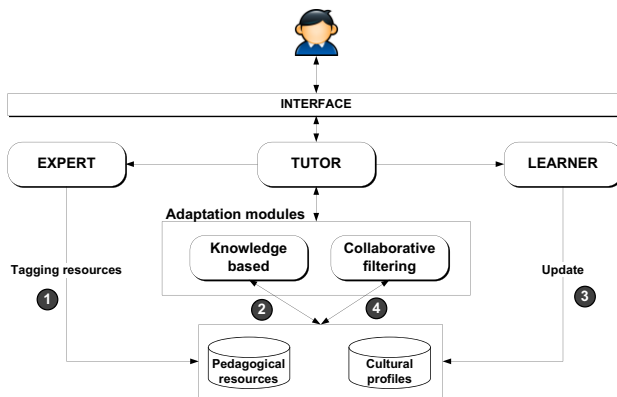


Fig. 1. System architecture

2.2 Collaborative Filtering Based Adaptation

A cultural background may depend on several factors including [4]: *the environment in which one spent most of adolescence, his/her hobbies, religion and degree of worship, countries of residence, etc.* It is therefore unwise to rely solely on nationality to determine a learner’s cultural profile. The *Collaborative Filtering* (CF) adaptation module (Fig.1), based on CF technique, allows to implicitly infer the learner’s cultural preferences over the pedagogical resources. The cultural profile used by the CF module contains the learner’s appreciation (numeric value from 0 to 5) of pedagogical resources (Fig. 2).

During a learning session the learner can change directly in the problem interface a pedagogical resource that is not culturally suited, from a set of equivalent resources for the same concept (e.g.: changing a maple syrup image for a cassava image). The learner’s cultural profile is then dynamically updated (arrow 3, Fig.1) incrementing by one the newly selected resource’s scoring ($R_4 = 2 + 1 = 3$). As learners sharing the same cultural background understand concepts using the same frame of reference they will have similar resource scoring. To integrate the pedagogical resources in problems for a given learner, CF algorithm (in CF module) selects pedagogical resources that were appreciated by learners with the same or close cultural background (arrow 4,

Fig.1). A cultural similarity coefficient is computed between a target learner and all other learners using the *Pearson correlation* defined by the following equation:

$$\text{Sim}_{(t,u)} = \frac{\sum_{i=1}^m (s_{t,i} - r_t) \times (s_{u,i} - s_u)}{\sqrt{\sum_{i=1}^m (s_{t,i} - r_t)^2 \times \sum_{i=1}^m (s_{u,i} - s_u)^2}} \quad (1)$$

Where $s_{t,i}$ is the target learner’s score for the resource i and $s_{u,i}$ stands for another learner’s score for the same resource i . The cultural similarity coefficient is computed over the m resources in the domain knowledge.

		Pedagogical resources				
		R ₁	R ₂	R ₃	R ₄	
Concepts	C ₁	1	3	1	2	
	C ₂	0	0	3	1	1
	C ₃	2	1	4		
⋮						

Example
 C₁ = “Carbohydrate-rich foods”
 R₁ = “Maple syrup”
 R₂ = “Bok choy”
 R₃ = “Potatoes”
 R₄ = “Cassava”

Fig. 2. Implicit cultural profile

3 Conclusion and Further Work

This paper presents an extension of the learner module within ITSs in order to dynamically adapt pedagogical resources to the learner’s cultural profile. The adaptation technique is based on collaborative filtering and allows inferring cultural preferences which the system is unable to retrieve explicitly from the learner. The implicit cultural profile is automatically updated using the learner’s interactions with the system throughout learning sessions. The update process is therefore transparent and requires less additional work for the learner than approaches based on an explicit cultural profile. An experimental validation is currently in progress.

References

1. Blanchard, E., Lajoie, P.: Learner-Concerned AIED Systems: Affective Implications when Promoting Cultural Awareness. In: 2nd International Workshop on Cultural-Aware Tutoring System (CATS), Brighton, pp. 13–23 (2009)
2. Hofstede, G.: Culture’s Consequences: Comparing Values, Behaviors and Organizations across Nations (2003)
3. Savard, I., Bourdeau, J., Paquette, G.: Cultural Variables in the Building of Pedagogical Scenarios: the Need for Tools to Help Instructional Designers. In: 1st International Workshop on Cultural-Aware Tutoring System (CATS), Montreal, pp. 83–92 (2008)
4. Reinecke, K., Schenkel, S., Bernstein, A.: Modeling a User’s Culture. In: The Handbook of Research in Culturally-Aware Information Technology: Perspectives and Models. IGI Global (2009)

An Interactive Learning Environment for Problem-Changing Exercise

Tsukasa Hirashima, Sho Yamamoto, and Hiromi Waki

Graduate School of Engineering, Hiroshima University
tsukasa@isl.hiroshima-u.ac.jp

Abstract. In this paper, an interactive learning environment for problem-changing exercise where a learner is required to solve and change the problem has been described. Activity to make a new problem from the original one and to compare their solutions is promising to promote a learner to be aware of the relation between the problems. For knowledge-rich problems, for examples word problems in arithmetic, mathematics or physics, this awareness is very important to master the use of solution methods. In order to realize such exercise in physics, we have already developed a prototype of computer-based learning environment that allows a learner to change a problem and can also diagnose the problem change and give feedback for the learner.

Keywords: Problem-Posing, Problem-Changing, Knowledge-Rich Problem.

1 Introduction

Word problems in arithmetic, mathematics or physics are often called as knowledge-rich problem that requires a problem solver to use rich domain-specific knowledge and to carefully interpret rich semantic structure of the problem [1]. In such domains, problem solving exercise is an indispensable learning activity to master the use of the domain-specific knowledge. In the exercise, it is important for learners to solve not only more problems but also various kinds of problems with difference semantic structures. Moreover, in order to realize effective learning during the problem solving exercise, it is important for the learners to be aware of the difference between problems. It is well-known that poor problem solvers are often unaware of the semantic structure of the problems from the viewpoint of problem solving. Several researchers have already suggested that problem-changing by learners where a learner poses a new problem by changing the existing problem, is a promising method to promote them to be aware of the difference between problems [2, 3]. However, one of the most difficult issues to effectively realize such learning activity is the way to give feedback for the learner's problem changes. In order to give useful feedback, it is necessary to assess the problem change that is composed of the original problems, the new problems and the differences. If a learner carries out this learning by him/herself, the learner is required not only change and solve the problems but also to assess his/her problem change. It is often too difficult for the learners to complete these tasks. Although a teacher can able to assess the problem change and give feedback based on

the assessment, taking care of some learners at a time is somewhat hard because the learners are usually allowed to change problem in different ways. Mutual assessment by learners is a solution of this issue but to complete these tasks is not easy for learners, especially for the beginners. We have investigated the function of automatic assessment of learner's problem change in order to make "problem-changing exercise" as a more common and useful learning method. We call the framework of the automatic assessment as "agent-assessment", because the above-mentioned first assessment is often called as "self-assessment", the second as "teacher-assessment" and the last as "peer-assessment".

We have paid special attention for learning by problem posing and have already developed interactive learning environments for "solution-based problem-posing exercise with agent-assessment" in arithmetical word problems [4]. In this study, we have investigated "problem-change" and developed an interactive learning environment for learning by problem change. In this paper, the framework of problem-changing exercise has been described by comparing problem-solving and solution-based problem-posing exercises. Then, a prototype of learning environment for the problem-changing exercise has also been explained.

2 Learning Environment for Problem-Changing Exercise

2.1 Framework of Problem-Change Exercise

In this subsection, the framework of problem-changing exercise has been described by comparing problem-solving exercise and solution-based problem-posing exercise. In Figure 1, models of the three kinds of exercises are shown. In problem-solving exercise to master a solution method, a learner is required to solve several problems that can be solved by the same solution method. Through the problem solving

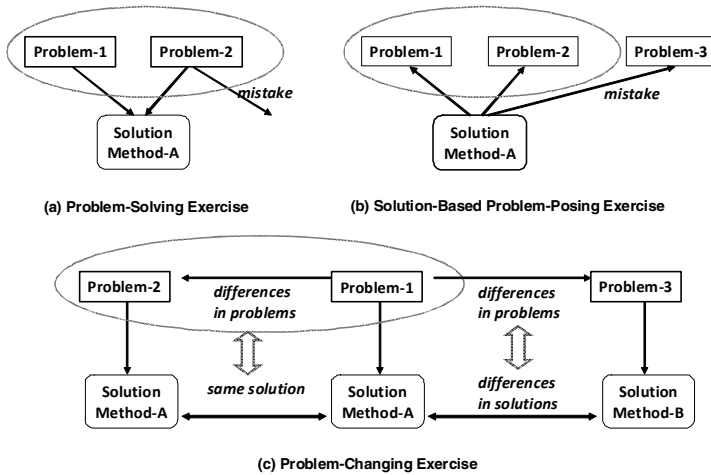


Fig. 1. Models of three types of exercises

exercise, it is expected that the learner can understand the way to use the solution method. In solution-based problem-posing, a learner is required to pose problems that can be solved by the same solution method. The both exercises include no direct activity to promote awareness for the differences among problems or solution methods. In problem-changing exercise, a learner has been provided with a problem and is required to solve it. The learner is, then, required to make a new problem by changing the provided problem. Problem-1 in Figure 1(c) corresponds to the original one and Problem-2 or Problem-3 corresponds to the generated one. The learner is also required to solve the generated problem. It is the next target of the problem-change. Because a learner makes the differences in problems by him/herself, the differences are well-known to the learner. The awareness of the differences in solution methods is also expected because he/she solves the original problem just before the problem change and solves the new problem just after the problem change.

2.2 A Prototype System

The interface of the learning environment for problem-changing exercise is shown in Figures 2. The interface is composed of status change area, situation change area, and solution description area. In situation change area, a learner can able to change the configuration of physical situation by changing the physical objects and their positions. These changes effect to the set of attributes. In the status change area, a learner can able to change the status of attributes; given, unknown and required. In the solution description area, a learner describes the solution of the problem. A learner can is able to complete these tasks by drag & drop or menu selection. After completing the problem change, the two problems and their solution methods are shown to promote to think of their relations.

Thought preliminary evaluation of the learning environment, the problem-changing in the learning environment promoted the subjects attended the experiment to be aware of the relations between problems from the viewpoint of solution methods.

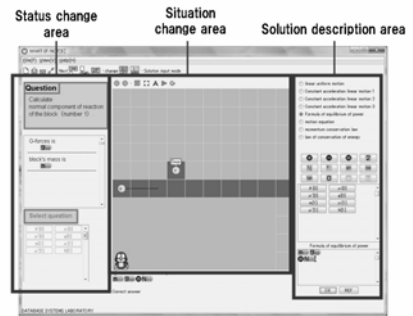


Fig. 2. Learning Environment for Problem-Changing Exercise

References

1. VanLehn, K.: Problem solving and cognitive skill acquisition. In: Posner, M. (ed.) *Foundations of cognitive science*, Erlbaum, Mahwah (1989)
2. Polya, G.: *How to Solve It: A New Aspect of Mathematical Method*. Princeton University Press, Princeton (1957)
3. Brown, S.I., Walter, M.I.: *Problem Posing: Reflections and Applications*. Lawrence Erlbaum Associates, Mahwah (1993)
4. Hirashima, T., Yokoyama, T., Okamoto, M., Takeuchi, A.: Learning by Problem-Posing as Sentence-Integration and Experimental Use. In: *AIED 2007*, pp. 254–261 (2007)

Towards Intelligent Tutoring with Erroneous Examples: A Taxonomy of Decimal Misconceptions

Seiji Isotani¹, Bruce M. McLaren¹, and Max Altman²

¹ Human-Computer Interaction Institute, Carnegie Mellon University, PA, USA
{sisotani, bmclaren}@cs.cmu.edu

² Gila Ridge High School, AZ, USA
max.altman@vanderbilt.edu

Abstract. In the mathematics domain of decimals, students have common and persistent misconceptions. These misconceptions have been identified, studied, and published by many researchers, spanning over 80 years of time. However, no paper discusses and brings together *all* of the identified misconceptions. This paper presents an initial taxonomy of decimal misconceptions, summarizing the results of past work. We also discuss the potential use and benefits of such a taxonomy in supporting the development of intelligent tutors that use erroneous examples as a learning tool for middle-school math students.

Keywords: Decimals, misconceptions, intelligent tutors, erroneous examples.

1 Introduction

The understanding and correct use of decimals is a foundational topic necessary for understanding more advanced mathematical topics. However, past research has indicated that learning decimals is very difficult, leaving students with a variety of misconceptions that often persist into adulthood. Through an extensive math education literature review, covering over 40 published papers and extending as far back as 1928 [e.g., 1,2,4,5,7,8], we found that most past work addresses either a single misconception or a small set of related misconceptions. In other words, the knowledge about the problems students, and even adults, have with decimals, and the means by which these misconceptions can be addressed, is spread over a variety of published papers. There is no single blueprint or guide to how students struggle with decimals and how educational technology can be used to overcome these struggles.

As a step toward addressing this issue, this paper presents a preliminary and partial taxonomy of decimal misconceptions derived from our literature review. In addition, we discuss our intended use of the taxonomy to support the development of an intelligent tutor and erroneous examples to address students' decimal misconceptions.

2 Taxonomy of Decimal Misconceptions

The benefits of having a taxonomy of decimal misconceptions are, on one hand, to provide an overview of the common problems that students have while working with decimals and, on the other hand, to gain insight into selecting appropriate and

effective instructional strategies to help ameliorate misconceptions. Decimals treated as whole numbers or fractions [8] and incorrect beliefs such as “multiplication makes bigger” and “division makes smaller” [2,4] are examples of common misconceptions.

Our literature survey shows that there is no single paper that presents decimal misconceptions in a comprehensive manner. The closest to comprehensive coverage is a paper by Stacey et al [7]. Our work extends the efforts of Stacey et al. by including other research used to construct a taxonomy that covers the most common misconceptions found in previous research.

Much of the past work focuses on how prior knowledge of other areas of math, such as fractions, whole numbers and negative numbers, can interfere with understanding decimals. Figure 1 shows a section of the decimal misconception taxonomy related to prior knowledge. Individual misconceptions have been given short, mnemonic names for easy remembrance (e.g., Decimals misconstrued as NEGative numbers is “Negz”). The complete taxonomy also includes misconceptions related to operations (e.g., multiplication, as in “multiplication makes bigger”). The taxonomy illustrates that students often have misconceptions based on pre-existing knowledge and prior learning. For example, researchers have empirically shown that some students believe that shorter decimals are larger because of a confusion with prior learning of fractions (e.g. $0.2 > 0.25$ because $1/2 > 1/25$). Other students believe that longer decimals are larger ($0.25 > 0.7$) because they confuse decimals with prior learning of whole numbers (e.g., $0.25 > 0.7$ because $25 > 7$).

By organizing and relating misconceptions we provide a unique resource that can be useful both for teachers and for ITS developers to prepare appropriate educational resources to support learning.

Currently we are working on the development of an intelligent tutor that will use this taxonomy as part of its user model, one that associates students’ mistakes with the corresponding misconception(s). Accordingly, we can then adapt the feedback (hints, next problems, concrete examples, etc.) to help students better understand the concept of decimals and address the particular misconceptions that led them to internalize the incorrect behavior/thinking.

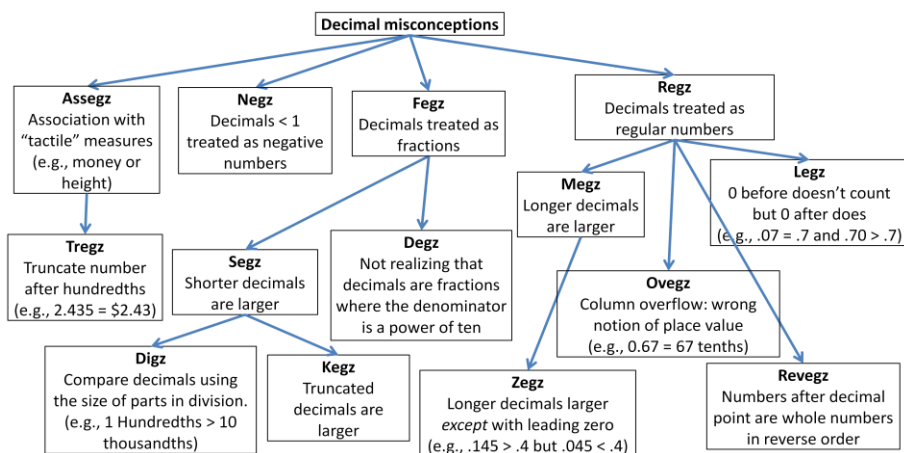


Fig. 1. Part of the misconception taxonomy related to students’ previous knowledge

Another key use of our taxonomy is the creation of an erroneous example-based tutor. An erroneous example (ErrEx) is a step-by-step problem solution in which one or more of the steps are incorrect. Past work has shown that presenting students with errors is valuable, as long as students are not pure novices in the problem domain [3,9]. Through the use of ErrEx within an intelligent tutor we will give students an opportunity to (a) compare correct solutions with incorrect ones; (b) help them understand why a given solution is wrong; and finally, (c) guide them to self explain incorrect solutions. Such an approach we believe will encourage critical thinking and motivate reflection and inquiry. This three-step process with erroneous examples is also consistent with Stellan Ohlsson's theory on learning from performance errors [6].

The erroneous examples-based tutor briefly discussed in this paper will eventually be freely available through MathTutor (webmathtutor.org), a website to help middle school children learn math with intelligent tutors. A description of our project (AdaptErrEx) is found at www.cs.cmu.edu/~bmclaren/projects/AdaptErrEx/.

3 Conclusions

We have discussed an initiative to create a taxonomy of decimal misconceptions. The effort has focused on gathering the findings from math education literature, spanning over 40 published papers, into a single representation. Currently, we are using this taxonomy to drive the creation of an intelligent tutor that will identify student misconceptions and support learning. We also emphasize the use of ErrEx, rarely used by math teachers but potentially quite helpful to student learning. We believe that ErrEx presented to students in an intelligent and adaptive fashion can provide the opportunity to find and reflect upon errors in a way that will lead to deeper and more robust learning. Our future work will investigate the benefits of an ErrEx-based tutor in supporting the learning of decimals.

References

1. Brueckner, L.J.: Analysis of Difficulties in Decimals. *Elementary School Journal* 29, 32–41 (1928)
2. Graeber, A., Tirosh, D.: Multiplication and division involving decimals: Preservice elementary teachers' performance and beliefs. *Journal of Mathematics Behavior* 7, 263–280 (1988)
3. Grosse, C.S., Renkl, A.: Finding and fixing errors in worked examples: Can this foster learning outcomes? *Learning and Instruction* 17(6), 612–634 (2007)
4. Hiebert, J.: *Mathematical, Cognitive, and Instructional Analyses of Decimal Fractions*, ch. 5, pp. 283–322. Lawrence Erlbaum, New Jersey (1992)
5. Irwin, K.C.: Using everyday knowledge of decimals to enhance understanding. *Journal for Research in Mathematics Education* 32(4), 399–420 (2001)
6. Ohlsson, S.: Learning from performance errors. *Psychological Review* 103(2), 241–262 (1996)
7. Stacey, K., Helme, S., Steinle, V.: Confusions between decimals, fractions and negative numbers. In: *25th Conference of the International Group for the Psychology of Mathematics Education*, vol. 4, pp. 217–224 (2001)
8. Resnick, L.B., Neshor, P., Leonard, F., Magone, M., Omanson, S., Peled, I.: Conceptual bases of arithmetic errors: The case of decimal fractions. *Journal for Research in Mathematics Education* 20(1), 8–27 (1989)
9. Tsovaltzi, D., Melis, E., McLaren, B.M., Dietrich, M., Gogvadze, G., Meyer, A.-K.: Erroneous examples: A preliminary investigation into learning benefits. In: Cress, U., Dimitrova, V., Specht, M. (eds.) *EC-TEL 2009. LNCS*, vol. 5794, pp. 688–693. Springer, Heidelberg (2009)

The Efficacy of iSTART Extended Practice: Low Ability Students Catch Up

G. Tanner Jackson¹, Chutima Boonthum², and Danielle S. McNamara¹

¹ Department of Psychology, University of Memphis,
38152 Memphis, TN

{gtjacksn, dsmcnamr}@memphis.edu

² Department of Computer Science, Hampton University,
23668 Hampton, VA

{chutima.boonthum}@hamptonu.edu

Abstract. iSTART is an Intelligent Tutoring System designed to improve students' reading comprehension skills. iSTART was the main component in a long term experiment (across a full academic year) with 389 students who completed a pretest, interacted with iSTART for 6 months, and then completed a posttest. A new extended practice module was implemented, which provided students with repeated practice across a variety of texts. Analyses found improvement in performance for all students, and indicate that students' initial self-explanation abilities may differ, but these abilities improve and converge as a function of practice.

Keywords: Intelligent Tutoring Systems, reading comprehension, long-term learning.

1 iSTART

Interactive Strategy Training for Active Reading and Thinking (iSTART) is a web-based tutoring system designed to improve students' reading comprehension by teaching self-explanation strategies [1], [2]. iSTART utilizes pedagogical agents to introduce students to self-explanation and reading strategies that improve reading comprehension. iSTART consists of three modules that implement the pedagogical principle of modeling-scaffolding-fading: introduction to the strategies, demonstration of the strategies, and interactive practice with the strategies.

The current investigation focuses on the extended practice module for iSTART. Students complete the 2-hour initial training (introduction, demonstration, and brief practice), and followed by a long-term practice module that includes a large repertoire of texts. An animated agent, Merlin, provides feedback on their self-explanations and prompts them to generate self-explanations using their newly acquired repertoire of strategies. To guide feedback to the students, iSTART evaluates each of their self-explanations using an automated NLP algorithm, coding each explanation as a 0, 1, 2, or 3. Self-explanations receive a "0" if they are too short or irrelevant, "1" if they refer only to the target sentence (sentence-based), "2" if they tie in a specific part of

the previous text (text-based), or “3” if they incorporate relevant outside information or an overall theme (global-based). The function[3] and accuracy[4] of the iSTART algorithm has been assessed and performance was found to be comparable to humans.

2 Experiment and Results

The current experiment was conducted over an academic year and included 389 students who participated in all three phases: pretest, iSTART, and posttest. Near the beginning of an academic year, students completed a set of pretest questions, including an assessment of initial self-explanation ability. After the pretest, students interacted with iSTART on a weekly basis over the period of an academic year. Near the end of the academic year, students answered a set of posttest knowledge measures, including an assessment of self-explanation ability.

A repeated-measures ANOVA confirmed that the quality of students’ self-explanations significantly improved from pretest to posttest, $F(1, 562) = 6084.70$, $p < .001$. However, students interacted with texts at their own pace and consequently experienced a different number of total texts and generated a different number of self-explanations. Thus, learning curves on self-explanation quality were calculated for each student and the slopes of those curves were used to investigate the overall learning trend for extended practice. A one-sample t-test confirmed that the average learning curve (slope=.53) was significantly above zero, $t(357) = 3.050$, $p < .01$, thus indicating a positive relation between self-explanation quality and the number of texts completed. Additionally, a regression analysis revealed that when averaged across students, the number of texts self-explained during extended practice significantly predicts the average self-explanation quality, $F(1, 39) = 106.05$, $p < .001$, $R^2 = .731$.

Participants were separated into two groups (median split), based on their ability to self-explain at pretest, and the extended practice texts were split into two groups: the first ten texts completed for each student as compared to any texts completed after the first ten for each student. A 2x2 ANOVA on self-explanation quality revealed a significant interaction between prior self-explanation ability and the two text groups, $F(1, 3708) = 4.413$, $p < .05$ (see Table 1). It was found that, for the first ten texts, students with an initial high ability produced significantly better quality self-explanations than did students with an initial low ability, $F(1, 2479) = 18.81$, $p < .001$. It was also found that after the first 10 texts, all students had improved, and low ability students produced self-explanations comparable to the high ability students, $F(1, 1229) = 0.24$, $p > .05$. For both groups of students, the self-explanations for the first 10 texts within extended practice were significantly lower in quality than the self-explanations for all subsequent texts after the first ten, $F(1, 3710) = 102.95$, $p < .001$.

Table 1. Mean (SE) self-explanation score within extended practice

	“First 10” Texts	All texts after “first 10”
High self-explanation ability	1.91 _a (.019)	2.09 _c (.025)
Low self-explanation ability	1.80 _b (.018)	2.07 _c (.027)

Note: means with a different subscript are significantly different ($p < .05$)

3 Discussion

These results demonstrate that iSTART improves students' self-explanation abilities. Analyzing each participant's progress across time and across texts provides a detailed learning trajectory for each student, and demonstrates that the pedagogy employed by iSTART improves students' performance. The learning trajectory analyses demonstrate a significant improvement during training, while the repeated-measures analysis indicates that students retain these cognitive benefits outside of training.

An implication for all ITS researchers is that this data demonstrates the importance of long-term evaluation. At the end of the basic iSTART training, the initial prior ability levels remained pervasive. Because this study provided extended practice over an entire year, it provides a rich dataset that illustrates the importance of extended practice.

Current research efforts are focused on improving this long-term interaction through the addition of game-based elements. A new version of iSTART has been developed that incorporates a point based economy with purchasable upgrades, environment changes, personalizable avatars, and educational mini-games. These game-based elements will complement the ITS by rendering the reading strategy tutoring not only effective, but enjoyable.

Acknowledgements

This research was supported in part by the National Science Foundation (IIS-0735682) and the Institute for Educational Sciences (IES-R305G040046). 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 NSF or IES.

References

1. O'Reilly, T., Best, R., McNamara, D.S.: Self-Explanation Reading Training: Effects for Low-Knowledge Readers. In: Forbus, K., Genter, D., Regier, T. (eds.) *Proceedings of the Twenty-Sixth Annual Meeting of the Cognitive Science Society*, pp. 1053–1058. Erlbaum, Mahwah (2004)
2. Magliano, J.P., Todaro, S., Millis, K.K., Wiemer-Hastings, K., Kim, H.J., McNamara, D.S.: Changes in Reading Strategies as a Function of Reading Training: A Comparison of Live and Computerized Training. *Journal of Educational Computing Research* 32, 185–208 (2004)
3. McNamara, D.S., Boonthum, C., Levinstein, I.B., Millis, K.: Evaluating Self-Explanations in iSTART: Comparing Word-Based and LSA Algorithms. In: Landauer, T., McNamara, D.S., Dennis, S., Kintsch, W. (eds.) *Handbook of Latent Semantic Analysis*, pp. 227–241. Erlbaum, Mahwah (2007)
4. Jackson, G.T., Guess, R.H., McNamara, D.S.: Assessing cognitively complex strategy use in an untrained domain. In: Taatgen, N.A., van Rijn, H., Schomaker, L., Nerbonne, J. (eds.) *Proceedings of the Thirty First Annual Meeting of the Cognitive Science Society*, pp. 2164–2169. Cognitive Science Society, Amsterdam (2009)

Expecting the Unexpected: Warehousing and Analyzing Data from ITS Field Use

W. Lewis Johnson, Naveen Ashish, Stephen Bodnar, and Alicia Sagae

Alelo Inc, 12910 Culver Bl., Suite J, Los Angeles, CA 90066 USA
{ljohnson,nashish,sbodnar,asagae}@Alelo.com

Abstract. One should expect the unexpected when deploying intelligent tutoring systems. This paper describes a case study in collecting, warehousing, and analyzing field usage data from two language and culture learning environments, to understand what happened when they were deployed. A data warehousing system, *Hoahu*, was used to process the raw data and transform it into a relational database to facilitate queries and analysis. The system also supported data annotation by subject matter experts to facilitate comparison of automated assessments against human raters. Errors and inconsistencies in the data were identified and corrected. The resulting data warehouse has proven valuable for understanding the trajectory of learning over extended periods of time and analyzing the strengths and weaknesses of complex interactive subsystems such as spoken dialog systems.

Keywords: empirical studies, educational data mining, language learning, dialog systems.

1 Introduction

It is often difficult to predict how intelligent tutoring systems (ITSs) perform in the field. Patterns of learner performance may emerge over time that were not apparent in short-duration tests common to formative evaluations. The performance of the system depends in part on the learning trajectory of each learner. Such issues arise quite commonly with the interactive learning environments that Alelo develops, which learners employ for tens or even hundreds of hours, and which incorporate animated characters that engage in many spoken conversations with learners.

This paper presents a case study in which data from field use of two intelligent tutoring systems were collected, warehoused, and analyzed. Prior to the release of the latest versions of our Iraqi Arabic and Sub-Saharan French courses, Naval personnel at several sites around the United States volunteered to take the courses in self-study mode, in their spare time. After the trainees completed their training, we retrieved logs and speech recordings from the training sessions. We then used a data warehousing system, named *Hoahu* to process and organize the data in a database for analysis and query purposes. (*Hoahu* means “to collect” in Hawaiian.)

2 Hoahu Data Warehouse

A schematic overview of Hoahu is provided in Fig. 1. One can look at Hoahu as a pipeline that takes raw log data and recordings and transforms them to a form amenable for high-end analysis. The pipeline works as follows. Data in the logs are first sent through *Kapaa*, the anonymizer module in Hoahu, which creates an anonymized data image that we then elaborate on. The data are then processed by *Ono*, a module that identifies and extracts objects of interest.

Ono creates a relational representation of these objects, and identifies and stores relationships between objects, specifically containment relationships. Analysts can then use this database for analysis of log data – at present we are using structured queries (SQL) over the database.

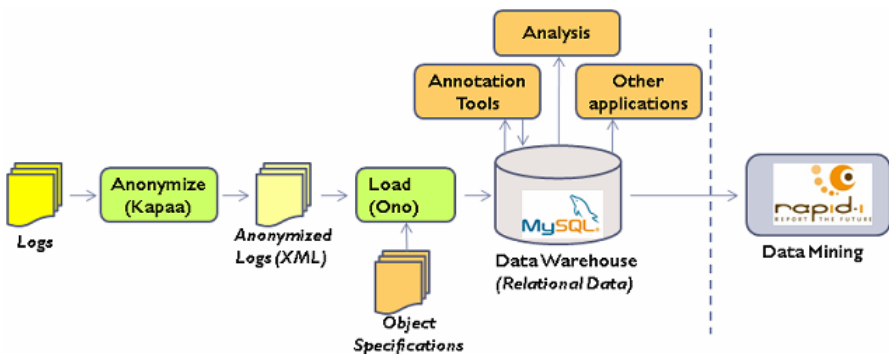


Fig. 1. Hoahu Architecture

Mostow & Beck [1] present a good summary of research activities that can be supported with appropriate system logging ("why to log"), including reporting, mining, and browsing historical data, in addition to tutoring itself. They also present a set of recommendations for *what* and *how* to log. Hoahu adapts these recommendations to a large-scale application context, improving on prior work where logs were analyzed using ad-hoc tools [2]. Other systems that are consistent with the Mostow & Beck model include DataShop [3] and the ASSISTment Builder [4].

3 Results

The following analysis focuses on trainees who completed at least 4 hours of training (8 out of 25 Arabic trainees and 5 out of 20 French trainees). We were particularly interested in studying *dialog breakdowns*, or what Jordan, et. al, [5] call *mis-hearings* and *misunderstandings*. We define a dialog breakdown as any dialog turn where one speaker (e.g., the learner) says something and the other speaker (e.g., an agent) responds in a way that suggests that it did not understand. A certain number of dialog breakdowns is anticipated, just as breakdowns occur in real life when language learners interact with native speakers. However persistent dialog breakdowns are likely to lead to learner frustration.

We chose $N \geq 4$ as the threshold for the number of utterance attempts at which point the dialog breakdown was considered unacceptably severe. The breakdown rates for the two languages were very similar, 6.5% for Arabic vs. 7.1% for French. They were substantially lower than the rate in an earlier pilot study with intermediate French speakers (18.6%). When focusing on the exercises in common between the two data sets, the rate was also lower (7.9% vs. 18.6%). The mean number of utterance attempts per dialog turn was 1.73 in the Arabic data and 1.64 in the French data, vs. 2.2 in the pilot data. Reviews of the speech recordings of the different groups revealed that the speech of the field trial learners was very different from that of the pilot test learners in terms of complexity and pronunciation accuracy. Many of the beginning learners' utterances are very badly pronounced, or even unintelligible. This illustrates the importance of using authentic field data to assess system performance.

We are currently having human raters annotate samples of the learner data, to judge the intelligibility of the speech and the accuracy of the system's interpretation of the speech. In contrast with typical speech recognition applications, our goal is *not* to achieve the highest possible speech recognition rates, but rather to correctly recognize intelligible speech and to reject and diagnose errors in errorful speech.

Meaningful response rates, when the agents understood and responded to the learners' speech, were 58% for Arabic and 59% for French. When learners did not rely on hints, the rates were 51% for Arabic and 58% for French. The speech recognition acceptance rates were 66% for Arabic and 57% for French. Learners must repeatedly attempt utterances until they get a meaningful response, which tends to multiply the number of recognition failures and non-meaningful responses. The gap between recognition and response rates can result from issues with the dialog model.

Acknowledgments. This work was generously supported by the Office of Naval Research under the ISLET project.

References

1. Jack Mostow, J., Beck, J.: What, How, and Why should Tutors Log? In: Proceedings of EDM 2009, pp. 269–278 (2009)
2. Johnson, W.L., Wu, S.: Assessing aptitude for learning with a serious game for foreign language and culture. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 520–529. Springer, Heidelberg (2008)
3. Koedinger, K., Cunningham, K., Skogsholm, A., Leber, B.: An open repository and analysis tools for fine-grained, longitudinal learner data. In: Proceedings of EDM 2008, pp. 157–166 (2008)
4. Razzaq, L., Patvarczki, J., Almeida, S.F., Vartak, M., Feng, M., Heffernan, N.T., Koedinger, K.: The ASSISTment builder: Supporting the Life-cycle of ITS Content Creation. IEEE Transactions on Learning Technologies, Special Issue on Real-World Applications of Intelligent Tutoring Systems 2(2), 157–166 (2009)
5. Jordan, P., Litman, D., Lipschultz, M., Drummond, J.: Evidence of Misunderstandings in Tutorial Dialogue and their Impact on Learning. In: Artificial Intelligence in Education, pp. 125–132. IOS Press, Amsterdam (2009)

Developing an Intelligent Tutoring System Using Natural Language for Knowledge Representation

Sung-Young Jung¹ and Kurt VanLehn²

¹ Intelligent Systems Program, University of Pittsburgh, Pennsylvania, USA
syjung5@asu.edu

² School of Computing Informatics and Decision Systems Engineering,
Arizona State University, Arizona, USA
Kurt.Vanlehn@asu.edu

Abstract. Authoring the domain knowledge of an intelligent tutoring system (ITS) is a well-known problem, and an often-mentioned approach is to use authors who are domain experts. Unfortunately, this approach requires that potential authors learn to write and debug knowledge written in a formal knowledge representation language. If authors were able to use natural language to represent knowledge it would allow them to add and update knowledge far more easily. In this paper, the design of such an authoring system, ‘Natural-K’ is presented. Natural-K is an authoring system in which domain authors including non-programmers are able to add problem statements and background knowledge such as commonsense, in natural language.

Keywords: Authoring System for ITS, Knowledge acquisition, Knowledge representation using natural language, Natural language understanding.

1 Introduction

Intelligent tutoring systems (ITS) require authors to add a several types of knowledge including problem statements, principles, commonsense knowledge, etc. A difficulty in developing such systems is in how to acquire such knowledge from authors who are usually not computer programmers. When authors have added knowledge, they must be able to follow problem solving detect errors, explore fixes and finally updating the knowledge. But this has not been possible so far without help of a knowledge engineer because knowledge has been represented in formal language.

For the ITS community, where tutoring systems often have a finite set of problems and authors often merely want to add more problems, the ability to represent problem statements in natural language is particularly appealing. ISSAC [1], MECHO [3], and pSAT [4] receive a problem statement written in natural language text and then transform into formal language to solve the given problem. Although these projects demonstrated that the problem statements read by students could also be read by the tutor, this capability was limited because the background knowledge required for translating the problem statements was encoded in formal language or computer program code. Only computer programmers were able to add such knowledge.

Several projects have explored the acquisition of inferential domain knowledge from natural language. The basic idea was to transform natural language into formal representations such as logic forms [7] or semantic networks [8], and the systems perform inferences to accomplish a problem solving goal. Domain knowledge includes a great deal of commonsense knowledge. Several universal commonsense knowledge bases are being developed (Opencyc [2]). However, it is unlikely that the complete universal set of commonsense knowledge can be built soon.

The main idea presented in this paper is to have the system represent knowledge in natural language. Thus, authors including non-programmers can add knowledge, detect bugs, and fix them without help of a knowledge engineer.

2 A Problem Statement and Principles

The specific project is to replace the knowledge representation language of Pyrenees with natural language, then use Natural-K to re-author the Pyrenees knowledge base. Pyrenees is an intelligent tutoring system for equation-based problem solving [5] with two main types of domain knowledge: principles and problem statements (Fig. 1). The principle in the figure was modified using a natural language definition for each variable (ex: “the direction of vector_ at time t_”).

(a)	Problem skateboarder: “A skateboarder rolls at 1.3 m/s up an inclined plane angled at 143 degrees. What is her vertical velocity?”
(b)	Pa_equation (along(projection(offaxis, Vector, T), Axis)), V_x=V*Trig) :- match (Axis, 'xy_ component', [xy_/XY]), match(V_x, 'xy_ component of vector_ at time t_', [xy_/XY, vector_/Vector, t_/T]), match(V, 'the magnitude of vector_ at time t_', [vector_/Vector, t_/T]), match(V_dir, 'the direction of vector_ at time t_', [vector_/Vector, t_/T]).

Fig. 1. A problem example. (a) A problem statement. (b) A relevant principle.

Often an author wants to add new domain problems using existing principles. Then, he needs to supply (1) a problem statement, and (2) commonsense knowledge (or inference rules), always in natural language to the system.

3 Mapping to Standard Natural Language

The initial problem statement will be translated into precise natural language until the resulting language exactly matches the variables in a principle, which are also written in natural language as Fig. 1-(b); the principle then applies and produces an equation. The precise natural language that appears in the principles is decided upon by the knowledge engineer and is called *standard natural language*. Many different non-standard phrases all get converted into the same standard sentence (for example, ‘A skateboarder rolls at 1.3 m/s’ and ‘The speed of the skateboarder is 1.3 m/s’ into ‘The magnitude of the velocity of a skateboarder is 1.3 m/s’).

Mapping non-standard natural language to standard natural language is done by inference rules. Rules are entered by authors as two pieces of natural language. The left side of the rule corresponds to a condition, and the right side of the rule corresponds to new knowledge produced.

'A skateboarder rolls at 1.3 m/s up a plane'

implies *'the magnitude of the velocity of the skateboarder is 1.3 m/s'.*

The strings entered by authors are represented inside the authoring system as dependency graphs after parsing [7]. The parser does syntactic normalization for passive and active sentences. The translation process starts with a set of dependency graphs that represent the problem statement. Inference rules run in a forward chaining manner, augmenting the set with more dependency graphs until no new one can be added. Then, the variables of the principles are matched against the set of dependency graphs and produce equations.

As the amount of added rules increases, the system can automatically produce generalized rules. For instance, after entering the rule above and this new one:

'A sled glides at 0.2 mph up a hill'

implies *'The magnitude of the velocity of the sled is 0.2 mph'.*

the authoring system would produce a generalized rule with *semantically constrained variables* (ex: *object_*, *moves_*, etc). The generalization is driven by WordNet's ontology [6]. If the system finds a hypernym (a superclass word), such as "object" which is the least common ancestor of "skateboarder" and "sled", then the system produces a generalized rule with the semantically constrained variable, *object_*. In this way, the system generalizes the existing rule so that it will subsume both the old rule and the newly entered one.

'An object_ moves_ at num_ unit_ up a hill_'

implies *'the magnitude of the velocity of the object_ is num_ unit_'.*

4 Discussion and Further Works

An important issue in using natural language is that there can be a large number of different expressions that have the same meaning. This is called *the paraphrasing problem* [9]. An important thing that should be noted is that it is not a serious issue in an authoring system for ITS. Authoring tasks in ITS is task-specific. It is enough for an author to add only the knowledge required to solve a given problem. Thus, the required set of knowledge to recognize the paraphrases is fixed and determined by the given problem statement. In other words, the author doesn't have to make the system recognize all paraphrases not seen in the given sentence. For this reason, the issue of paraphrases is not a serious problem except the case of fully automatic text processing without help of human authors.

So far, the authoring system succeeded in adding 15 physics problem statements correctly; 98 inference rules were added for them. It showed that the average number of rules per a problem was around seven ($=98/15$) which is small enough for an author to add. The remaining work is to implement rule generalization, integrate with Pyrenee, and perform empirical evaluation of the system.

References

1. Novak Jr., G.S.: Representations of Knowledge in a Program for Solving Physics Problems. In: IJCAI 1977, Cambridge, MA, pp. 286–291 (1977)
2. Opencyc.org, OpenCyc Tutorial, <http://www.cyc.com/cyc/opencyc/overview>
3. Bundy, A., et al.: Solving mechanics problems using meta-level inference. In: IJCAI 1979, pp. 1017–1027 (1979)
4. Ritter, S., et al.: Authoring Content in the PAT Algebra Tutor. *Journal of Interactive Media in Education* (9), 1–30 (1998)
5. VanLehn, K., et al.: Implicit versus explicit learning of strategies in a non-procedural cognitive skill. In: *The Intern'l Conf on Intelligent Tutoring Systems*, pp. 521–530 (2004)
6. Harabagiu, S.M., Miller, G.A., Moldovan, D.I.: WordNet 2 - A Morphologically and Semantically Enhanced Resource. In: *SIGLEX 1999* (1999)
7. Jurafsky, D., Martin, J.H.: *Speech and Language Processing*. Prentice-Hall, Englewood Cliffs (2000)
8. Sowa, J.F.: Current Issues in Semantic Networks. In: *Principles of Semantic Networks: Explorations in the Representation of Knowledge*. Morgan Kaufmann, San Francisco (1990)
9. Iftene, A.: *Textual Entailment*. PhD Thesis, Computer Science, University of Iasi, Iași, Romania (2009)

A Network Analysis of Student Groups in Threaded Discussions

Jeon-Hyung Kang, Jihie Kim, and Erin Shaw

University of Southern California Information Sciences Institute
4676 Admiralty Way, Marina del Rey, CA, U.S.A
{jeonhyuk, jihie, shaw}@isi.edu

Abstract. As online discussion boards become a popular medium for collaborative problem solving, we would like to understand patterns of group interactions that lead to collaborative learning and better performance. In this paper, we present an approach for assessing collaboration in online discussion, by profiling student-group participation. We use a modularity function to compute optimal discussion group partitions and then examine usage patterns with respect to high-versus low-participating students, and high- versus low-performing students as measured by grades. We apply the profiling technique to a discussion board of an undergraduate computer science course with three semesters of discussion data, comprising 142 users and 1620 messages. Several patterns are identified, and in particular, we show that high achievers tend to act as ‘bridges’, engaging in more diverse discussions with a wider group of peers.

Keywords: Student online discussions, group detection in discussions.

1 Introduction

Online discussion boards play an important role in distance education and web enhanced courses. Studies have shown online discussion to be a promising strategy for promoting collaborative problem solving and discovery-oriented activities, however, student participation can vary highly; some students post only one or two messages during the whole semester, while others participate more often and interact with many other students. Some students communicate with only a limited number of peers, while others interact with a wider group of students and participate in more varied discussion topics. It is difficult to understand the many different types of group interactions that occur in online discussions and even more difficult to understand how they affect collaborative learning.

We would like to identify and understand patterns of group interactions that lead to collaborative learning and better performance. The patterns might be used to develop pedagogical strategies for promoting more desirable interactions and increasing student learning. Most of the existing computational work on qualitative discussion analysis has focused on analyzing dialogue patterns in individual discussion threads (Ravi & Kim 2007; Feng, Kim & Shaw, 2006, McLaren et al., 2008) or analyzing the impact of tutors in student discussions (Light et al., 2000; Shaw 2005). Although

these results provide good hints about student behavior within online discussions, they have not yet yielded insightful information about collaborative learning, such as how groups form during online interactions over multiple discussion threads, and how group interactions affect student learning.

To help profile discussion participation, we define three new terms: *high-participating students*, *active group participants (AGPs)* and *bridge students*. High-participating students are defined as those who participate in many discussion threads and AGP are students who participated in many discussion threads in the same group. We also define bridge students as those who participated in multiple discussion threads across several different groups.

2 Modeling Discussion Group

A discussion group is modeled based on information about its students and the discussion threads in which they participate. The relationship can be represented as a directed graph, where nodes are either users or threads, and edges connect users who participate in threads. The graph is then partitioned to detect optimal communities of discussants. This is done by first representing the graph as a discussion (student-thread) matrix, and then finding a partitioning that maximizes the strength of the connections within a group, called the modularity (Girvan and Newman, 2002; Ghosh and Lerman, 2008). The modularity measures how good the given partition is by comparing the difference between the number of edges that lie within groups in the given partition and the same quantity when the edges are placed randomly and the vertices have the same degree. Once an ideal community is found, we study patterns between and within groups. To set the threshold, we analyzed the degree of participation by individual students in our largest dataset, in which 50 discussants participated in 100 discussion threads. We applied the power-law to set the threshold. In the dataset for this work, the top 20% students participate in at least five discussion threads and 80% of the messages were written by top 20% of the students. We labeled the top 20% students high-participating, and the rest low-participating. We applied the same threshold for AGPs. AGPs participated in at least five different discussion threads in any given group. Since our model produces 4 groups in spring, 3 groups in summer, and 7 groups in fall 2008 semester, we used the smallest number to define bridge students. Bridge students participated in more than three threads that belong to more than three different groups.

3 Preliminary Results

Each class discussion forum we studied corresponded to a unique class project. 'Project 2' discussion communities were modeled because the forum had the highest number of participants. We then compared the corresponding 'Project 2' grades of *high- and low-participating students*, *AGPs*, and *bridge students*. The results are shown in Table 1. To evaluate the differences in grades between different pairs of groups we applied the *t*-test. There was no significant difference in the grades of high- and low-participating students; however, AGPs received lower average grades than

Table 1. Student grades for Project 2 in three different semesters

Groups	Spring 2008	Summer 2008	Fall 2008	Year 2008
Bridge Students	37.3	-	39.3	38.4
Non-Bridge Students	33.4	34.2	34.6	34.3
High Participating Students	28.3	37.0	35.8	34.8
Low Participating Students	36.1	22.2	23.8	34.8
Active Group Participants	24.0	37.6	36.3	34.4
Non-AGP	36.1	33.4	34.9	34.9
All Students	34.3	34.5	35.1	34.8

non-AGP, high- and low-participating students. This is consistent with our earlier findings that high-participating students may be help seekers and not necessarily high performers. A full understanding will require an in-depth investigation of thread features, taking into account the technical quality of messages, dialogue patterns, technical terms used, and discussion topics.

Interestingly, we found that high-performing students, however, tend to act as bridge students across several different community groups, engaging in more diverse discussions with a wider group of peers. Specifically, they participated in multiple discussion threads across several different groups. A *t*-test for the comparison between bridge student and non-bridge student grades shows that the difference is significant at a 99.9% level with $t(56) = 4.39$, $p < 0.001$.

Acknowledgments. This work was supported by National Science Foundation CCLI Phase II (#0618859) and CISE IIS (#0917328) grants.

References

1. Feng, D., Kim, J., Shaw, E., Hovy, E.: Towards Modeling Threaded Discussion using Induced Ontology Knowledge. In: National Conference on Artificial Intelligence (2006)
2. McLaren, B., Scheuer, O., De Laat, M., Hever, R., De Groot, R., Rose, C.: Using Machine Learning Techniques to Analyze and Support Mediation of Student E-Discussions. In: AIED (2007)
3. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. *Natl. Acad. Sci. USA* 99, 7821–7826 (2002)
4. Ghosh, R., Lerman, K.: The structure of heterogeneous networks. In: 1st IEEE Social Computing Conference (2009)
5. Ravi, S., Kim, J.: Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers (2007)

A New Framework of Metacognition with Abstraction/Instantiation Operations

Michiko Kayashima¹ and Riichiro Mizoguchi²

¹ Tamagawa University, 6-1-1 Tamagawagakuen, Machida, Tokyo, 194-8610 Japan
kayasima@lit.tamagawa.ac.jp

² I.S.I.R., Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka, 567-0047 Japan
miz@ei.sanken.osaka-u.ac.jp

Abstract. While there is acknowledgement of the importance of metacognition in education, some researchers indicate that the domain of metacognition lacks coherence. In order to overcome this issue, it is necessary that each researcher explains his own approach by using his own or other people's framework of metacognition. We propose a new framework for metacognition which explains what types of metacognition-driven learning occur so to enable regulation of cognitive activities. With this framework, it becomes possible not only to identify what types of metacognitive activity a computer system supports but also to propose new functions that support the various types of metacognitive activities.

Keywords: metacognition, framework, abstraction operations, instantiation operations, metacognition-driven learning.

1 Introduction

Recently, research on metacognition has increased, however one issue that has been raised about the current situation is the difficulty researchers face in understanding each other, so to share more easily their respective achievements. A conceivable solution to this problem is that each researcher explains his own research by using his or other's framework of metacognition. To this purpose, a number of models and conceptual frameworks of metacognition and self-regulated learning have already been proposed. We also proposed our framework [4]. Unfortunately, however, our previous framework was not successful in explaining what types of metacognitive activity cause learning to enable regulation of cognitive activities. With the extended framework presented in this paper, it becomes possible to analyze existing training systems and identify what types of metacognitive activity the systems support with the same vocabulary; also, it is possible to propose new functions to support the various types of metacognitive activities.

2 What Types of Metacognitive Activities Occur in Metacognition-Driven Learning?

We tackle metacognition-driven learning, an advanced type of learning that is qualitatively different from repetitive use of metacognitive strategies. Collins et al. claim

that two kinds of abstractions are needed for reflection: “*The first concerns how to structure the problem solving audit trail, the second concerns choosing the right “grain size” of events that are to be stored on the audit trail*” [2, p.14]. In our interpretation both of them are metacognitive activities: the former is a *perceptual* abstraction and the latter is a *verbal* abstraction. By verbal abstraction, we mean the operation by which one can recognize the relation between a specific instance of a problem solving activity and its class-level verbal expression; we call this operation the *abstraction operation*. Class-level verbal expressions can be also used for controlling problem solving processes such as backward reasoning or forward reasoning; for these reasons Collins et al. call them *metacognitive strategies*. Similarly, we also consider verbal expression as a type of metacognitive activity, but we characterize it as an operation that generates an instance from a class; we call this an *instantiation operation*. In addition, in order to regulate our cognitive activities we require class-level modification of verbal expressions; we call this the *modification operation*.

Metacognition-driven learning can be defined as the process by which we learn through class-level modification of verbal expressions. This entails that a learner must see his own problem solving strategies as modifiable strategies grounded on his particular experience. Moreover, learners must be aware of the fact that by modifying problem solving strategies they can control their problem solving process.

Note here that learning scenarios that focus on the acquisition of single specific strategies do not bring about metacognition driven learning, because the latter is a meta-level learning that involves the modification of some preexisting problem solving strategy.

Here, we present an extended version of our previous framework [4]. We added an *abstraction operation*, a *modification operation*, and an *instantiation operation* to the previous metacognitive activities. Furthermore, we added a description of what metacognition-driven learning achieves, such as class deletion (unlearning), addition, modification and clarification of class attributes.

3 Analysis of Training Systems Based on the Extended Framework

We analyzed Betty's Brain system [1] and EBS [3].

Betty's Brain system aims at supporting the acquisition of complex scientific knowledge and learning self-regulation skills by asking a learner to teach a computer agent named Betty; the learner is helped in this process by Davis, another computer agent who plays the role of mentor. The trigger for observing use of self-regulated learning strategies is Betty's response or Davis's advice. With Betty's response, the learner reflects on whether he has been using strategies effectively on Betty. Davis' advice is an explanation of how, when, and why to use each learning strategy. Davis' advice promotes training of the instantiation operation in the strategies, but does not lead to the abstraction operation of strategy application or metacognition-driven learning. Thus, an analysis based on our extended framework suggests the addition of new functions that aim at supporting the abstraction operation. For example, suppose that with Betty's explanation the learner noticed an error in the cause-effect relationship in a concept map. We propose that in such a situation the learner should be asked

"Explain the reason why Betty misunderstood the cause–effect relationship". By doing so the learner is prompted to reflect on the reasons that caused Betty not to learn the cause–effect relationship (reasons that mirror the learner's own knowledge acquisition); as a result, the learner might understand the cause of the error.

Our second system, EBS, addresses the learning of Newtonian physics in particular by helping students unlearn erroneous knowledge. For example, learners might be asked to define what forces act on an object by drawing them on the screen. If they commit a mistake (e.g. by drawing a force that doesn't comply with Newton's theory) EBS exhibits some unusual behavior (e.g. by burying the objects in the ground). Upon seeing such strange behavior, learners realize that they have made a serious error and carefully reconsider the situation presented to them. In other words, EBS's strange behavior leads a learner to unlearn his erroneous principles and to recognize the class-level correct principles. Evaluation experiments successfully proved this point [3]. However, according to our framework EBS does not provide explicit support for a learner to learn *the abstraction operation*. We propose a new function that would overcome this limitation could be exemplified as follows. Imagine that a learner is shown a stationary object on the desk and asked "Do forces act on this stationary object?". The learner would answer "no force is acting on it". Next, a small ball of mass m that falls freely is shown. If asked the same question once more, a learner would answer " mg exerts on the ball downward". Then, the learner's two answers are presented together so that he is asked to explain why he answered differently. By doing so, the learner would notice that mg exerts on the stationary object downward as well. Such a thought would suggest to him to realize that gravity acts on stationary objects, thus leading him to the unlearning of the previous erroneous knowledge.

4 Conclusion

We proposed the *abstraction operation*, *modification operation*, and *instantiation operation* as metacognitive activities and extended our previous framework.

References

1. Biswas, G., Roscoe, R., Jeong, H., Brian Sulcer, B.: Promoting Self-Regulated Learning Skills in Agent-Based Learning Environments. In: Proceedings of ICCE 2009, pp. 67–74 (2009)
2. Collins, A., Brown, J.S.: The Computer as a Tool for Learning through Reflection. In: Mandl, H., Lesgold, A. (eds.) Learning Issues for Intelligent Tutoring Systems, pp. 1–18. Springer, New York (1988)
3. Hirashima, T., Imai, I., Horiguchi, T., Toumoto, T.: Error-Based Simulation to Promote Awareness of Errors in Elementary Mechanics and Its Evaluation. In: Proceedings of AIED 2009, pp. 409–416 (2009)
4. Kayashima, M., Inaba, A., Mizoguchi, R.: What Do You Mean by to Help Learning of Metacognition? In: Proceedings of AIED 2005, pp. 346–353 (2005)

Expansion of the xPST Framework to Enable Non-programmers to Create Intelligent Tutoring Systems in 3D Game Environments

Sateesh Kumar Kodavali¹, Stephen Gilbert¹, and Stephen B. Blessing²

¹ Virtual Reality Applications Center, Iowa State University

² University of Tampa

Abstract. Our previous work has demonstrated that the Extensible Problem Specific Tutor (xPST) framework lowers the bar for non-programmers to author model tracing intelligent tutoring systems (ITSs) on top of existing software and websites. In this work we extend xPST to enable authoring of tutors in 3D games. This process differs substantially from authoring tutors for traditional GUI software in terms of the inherent domain complexity involved, different types of feedback required and interactions generated by various entities apart from the student. A tutor for a village evacuation task has been constructed in order to demonstrate the capabilities of using the extended xPST system to create a game-based tutor.

Keywords: Intelligent Tutoring System, xPST, 3D Games, Authoring, Cognitive Tutor.

1 Background: xPST Authoring System

We developed the Extensible Problem-Specific Tutor (xPST) [1] in order to create ITS-based software training within the software itself. The xPST architecture is an instantiation of the architecture of plug-in tutor agents described in [2]. The xPST file, which contains information that allows for instruction akin to a model-tracing tutor, describes the objects within the learning domain and rules that determine which feedback the student will receive. A Listener plugin or module eavesdrops on user actions in the third party software and sends them to the xPST Tutoring Engine, which checks them with the xPST file. Feedback is mapped back to the client UI control and displayed appropriately. We have confirmed that the xPST approach can be used to develop real tutors rapidly; our most extensive effort is described in [3], in which a tutor taught university faculty how to use a complex web-based homework authoring tool. We have also built a web-based xPST editor to author xPST files [4].

2 Extensions to xPST

We have added the following extensions to the xPST architecture to facilitate easy authoring of ITSs in 3D games.

2.1 Actions by Non-player Objects

Events can be triggered by non-player objects in 3D games. These events are modeled as hypothetical events by a Player-class object which is not the user. For example *Avatar1:request-answer* is the step corresponding to requesting an answer from the Avatar1 entity, where *Avatar1* is the unique id of the entity and is the recipient of the associated action *request-answer*. This approach is useful in tutoring on generic actions associated with any entity in the game, such as Tanker1:explode, Enemy1:attack. This is a more generic way of handling events compared to the previous xPST architecture in which the unique ID attribute always corresponds to the Player class object and was hence ignored while writing in the file. This is because the previous xPST architecture could support tutoring only on events generated by the Player.

2.2 Proactive Hints or Prompts

Since the state space of a 3D game is quite complex with interactions between various entities, and since it is sometimes not obvious what the current game state is, it is sometimes useful for the learner to have direct feedback when the current step is completed or to receive some reminders about the next step. So we have included a new type of feedback in xPST, “*OnComplete*”, supplementing the potential *Hints* and *JITs* for each step. This feedback is proactively provided to the student on completion of that particular step.

2.3 Communication Events

In 3D games the student may be required to be able to communicate with other player entities. We have extended xPST to support tutoring on communication events by using a special step *starttalk*, to initiate the communication with other entities. The student will be able to choose the entity with which to communicate and the message to communicate. This approach facilitates tutoring on the protocol of communication and the message that is being communicated. If the student is supposed to choose *Evacuate* command for the task and if he chooses a different command, say, the *Fire* command, a *JIT* can be fired saying “*You used Fire command on this occupant. That’s not something you need to do right now.*”. Future research will evaluate the effectiveness of this framework within military scenarios.

2.4 Location Events

Location events facilitate tutoring on the navigational aspects of the player’s performance. Unlike the traditional GUI software or websites, almost every task in a 3D game requires the player to move within the virtual environment. The author can use the *entityid-enter* step to tutor on when the player enters a particular designated location in the game.

3 Evacuate Demo Task

We have developed a demonstration task called *Evacuate* to show that the extended xPST Framework can be used to create ITSs in 3D games. The task teaches the learner how to evacuate the civilians from all the buildings in the scenario. The xPST file for this task contains three major steps which illustrate the game-enabling extensions of xPST. The location-based *buildingid-enter* step is completed when the player enters the building with id *buildingid*. The *buildingid-evacuate* step is completed when the player sends the evacuate command to the civilian in the building with id *buildingid*. The *starttalk* step is completed when the player initiates communication with the civilian. All these steps are provided with appropriate *Hint*, *JIT* and *OnComplete* feedback to guide the student in successfully completing the task.

4 Conclusions

We have discussed the xPST framework which allows for fast creation of model-tracing tutor for a specific problem. We have also described the extensions that were required for the xPST framework to enable it to be able to tutor in 3D games. Finally, we have discussed a demonstration task showing that the extended xPST framework can be used to tutor in 3D games. In the future, we would like to evaluate the programmability of this system by conducting a study where we examine how novice xPST authors with little programming experience can learn to create to these kinds of tutors.

Acknowledgments. We thank Steven Ourada as the senior architect of xPST. This work was supported in part by the National Science Foundation under OII-0548754 and by the Air Force Office of Scientific Research.

References

1. Blessing, S., Gilbert, S., Blankenship, L., Sanghvi, B.: From SDK to xPST: A New Way to Overlay a Tutor on Existing Software. In: Proceedings of the Twenty-Second International FLAIRS Conference (2009)
2. Ritter, S., Koedinger, K.: An architecture for plug-in tutor agents. *Journal of AIED* 7(3-4), 315–347 (1992)
3. Roselli, R.J., Gilbert, S., Howard, L., Blessing, S.B., Raut, A., Pandian, P.: Integration of an Intelligent Tutoring System with a Web-based Authoring System to Develop Online Homework Assignments with Formative Feedback. In: American Society for Engineering Education Conference (2008)
4. Gilbert, S., Blessing, S.B., Kodavali, S.: The Extensible Problem-Specific Tutor (xPST): Evaluation of an API for Tutoring on Existing Interfaces. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education (2009)

A Computational Model of Accelerated Future Learning through Feature Recognition

Nan Li, William W. Cohen, and Kenneth R. Koedinger

School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh PA 15213 USA
{nli1,wcohen,koedinger}@cs.cmu.edu

Abstract. Accelerated future learning, in which learning proceeds more effectively and more rapidly because of prior learning, is considered to be one of the most interesting measures of robust learning. A growing body of studies have demonstrated that some instructional treatments lead to accelerated future learning. However, little study has focused on understanding the learning mechanisms that yield accelerated future learning. In this paper, we present a computational model that demonstrates accelerated future learning through the use of machine learning techniques for feature recognition. In order to understand the behavior of the proposed model, we conducted a controlled simulation study with four alternative versions of the model to investigate how both better prior knowledge learning and better learning strategies might independently yield accelerated future learning. We measured the learning outcomes of the models by rate of learning and the fit to the pattern of errors made by real students. We found out that both stronger prior knowledge and a better learning strategy can speed up the learning process. Some model variations generate human-like error patterns, but others learn to avoid errors more quickly than students.

Keywords: accelerated future learning, learner modeling.

1 Motivation and Algorithm

Perhaps one of the most interesting measures of robust learning is accelerated future learning. A growing number of studies have experimentally demonstrated that some instructional treatments lead to accelerated future learning. These treatments (and associated studies) include inventing for future learning [1], self-explanation [2], and feature prerequisite drill [3]. While results are starting to accumulate, we have little by way of precise understanding of the learning mechanisms that yield these results. A computational model of accelerated future learning that fits student learning data would be a significant achievement in theoretical integration within the learning sciences, and reveal insights on improving current education technologies.

Previous work [4] showed that one of the key factors that differentiates experts and novices is that experts view the world in terms of deep functional features, while novices see in terms of shallow perceptual features. In this paper, we propose a novel approach to modeling accelerated future learning through the use of

machine learning techniques to acquire deep features. We assume that the input of the system is a set of feature recognition records. Each record consists of an original problem (e.g. an expression, $-3x$), and the feature recognized from the problem (e.g. the coefficient in the problem, -3 in $-3x$). The objective of this work is to construct a computational model to learn feature recognition.

After careful examination of the problem, we find out that the feature recognition problem closely connects to the probabilistic context free grammar (PCFG) induction problem, where knowledge is represented by grammar rules, and the learning process is similar to grammar induction. Therefore, we extended a grammar induction algorithm proposed by Li et al. [5], since it acquires PCFG from observation sequences without any prior structural knowledge. Details about this learning algorithm are described in [5].

To support feature learning, after acquiring the grammar with Li et al.'s algorithm, our system finds the intermediate symbol that corresponds to the feature most frequently in the parse trees of the training examples, and identifies it as the target feature. To understand how prior knowledge and learning strategy could affect learning outcomes, we extended the learning algorithm in two directions. First, we designed a transfer learning mechanism that biases the probabilities of rules in future tasks toward the probabilities associated with previous tasks. The learner records the number of times each grammar rule appeared in a parse tree from previous tasks, and updates the rule probability in a new task by adding the previous applied rule frequency to the training problems. Second, we extended our learning mechanism to making use of a "semantic non-terminal constraint" embedded in training data during learning. More specifically, the learner forces all the feature subsequences to correspond to one non-terminal symbol.

2 Empirical Study

We carried out a controlled simulation study in algebra to test 1) whether stronger prior knowledge and better learning strategies could yield accelerated future learning, 2) if so, how prior knowledge and learning strategies affect the learning outcome. There were 2-by-2 (4) alternative versions of the proposed learning model in the study: L00, the original learner without transfer learning and the non-terminal constraint; L01, the learner with the non-terminal constraint but without transfer learning; L10, the learner with transfer learning but without the non-terminal constraint; L11, the learner with both transfer learning and the non-terminal constraint.

We designed three curricula in the study. Three tasks were used across the three curricula with increasing complexities. The three curricula are 1) task one, then task two; 2) task two, then task three; 3) task one, then task two, then task three. In all but the last task, each learner was given 10 training problems. For the last task, each learner was given one to five training records. Under each training condition, both systems were tested on 100 expressions in the same form of the training data in the last task. For each testing record, we compared the feature recognized by the oracle schemas with that recognized by the acquired schemas, and evaluated the correctness of output.

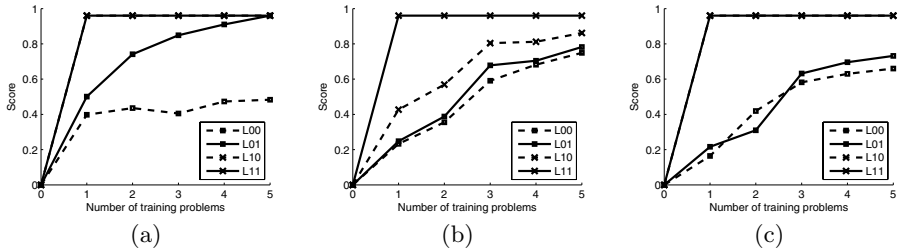


Fig. 1. Learning curves for four learners in curriculum (a) from task one to task two. (b) from task two to task three. (c) from task one to task two to task three.

We also compared the errors made by the learning system with common errors made by real students in curriculum one. We inspected the common errors made by real students in task two from a study of 71 high school students used Carnegie Learning Algebra I Tutor, and noticed that mishandling of negative coefficients (e.g., marking 3 instead of -3 as the coefficient of $-3x$) is the most common error. The learning system was asked to recognize the coefficients of the 100 given expressions, and was evaluated based on the match of the errors made by the learner and the most common error made by real students.

As shown in Figure 1(a), the result suggests that with transfer learning, learners are able to acquire knowledge quicker than those without transfer learning. Comparing the base learner, L00, and the learner with non-terminal constraint, L01, we can see that a better learning strategy yields a steeper learning curve. We can also see that in all three curricula, the transfer learner, L10, always outperforms the learner with semantic non-terminal constraint, L01. Similar results were also observed with curriculum two and curriculum three. This suggests that prior knowledge is more effective in accelerating future learning than better learning strategies. In the error matching study, we see that after being trained with one to five problems, L00 generated the most common error in testing. Besides that all other incorrect answers are due to the incapability of identifying a coefficient from the problem.

References

- Bransford, J.D., Schwartz, D.L.: Rethinking transfer: A simple proposal with multiple implications. *Review of Research in Education* 24, 61–100 (1999)
- Hausmann, R.G., VanLehn, K.: Explaining self-explaining: A contrast between content and generation. *Artificial intelligence in education: Building technology rich learning contexts that work* 158, 417–424 (2007)
- Pavlik Jr., P., Bolster, T., Wu, S.M., Koedinger, K., Macwhinney, B.: Using optimally selected drill practice to train basic facts. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008. LNCS*, vol. 5091, pp. 593–602. Springer, Heidelberg (2008)
- Chi, M.T.H., Feltovich, P.J., Glaser, R.: Categorization and representation of physics problems by experts and novices. *Cognitive Science* 5(2), 121–152 (1981)
- Li, N., Kambhampati, S., Yoon, S.: Learning probabilistic hierarchical task networks to capture user preferences. In: *Proceedings of the 21th International Joint Conference on Artificial Intelligence*, Pasadena, CA, USA (2009)

Automated and Flexible Comparison of Course Sequencing Algorithms in the LS-Lab Framework

Carla Limongelli¹, Filippo Sciarrone³, Marco Temperini², and Giulia Vaste¹

¹ DIA-Department of Computer Science and Automation
Roma Tre University - Via della Vasca Navale, 79 00146 Rome, Italy
{limongel,vaste}@dia.uniroma3.it

² DIS-Department of Computer and System Sciences
Sapienza University of Rome - Via Ariosto, 25 00185 Rome, Italy
mart@dis.uniroma1.it

³ Open Informatica srl
E-learning Division - Via dei Castelli Romani, 12/A 00040 Pomezia, Italy
f.sciarrone@openinformatica.org

Abstract. Curriculum Sequencing is one of the most interesting challenges in learning environments, such as Intelligent Tutoring Systems and e-learning. The goal is to automatically produce personalized sequences of didactic materials or activities, on the basis of each individual student's model. In this paper we present the extension of the LS-LAB framework, supporting an automated and flexible comparison of the outputs coming from a variety of Curriculum Sequencing algorithms over the same student models. The main aim of LS-LAB is to provide researchers or teachers with a ready-to-use and possibly extensible environment, supporting a reasonably low-cost experimentation of several sequencing algorithms. The system accepts a student model as input, together with the selection of the algorithms to be used and a given learning material; then the algorithms are applied, the resulting courses are shown to the user, and some metrics computed over the selected characteristics are presented, for the user's appraisal.

Keywords: adaptive e-learning, learning object sequencing.

1 Introduction

Curriculum sequencing is one of the key components in classic Intelligent Tutoring System (ITS) [46]. Each solution to this problem has its own strength and weakness: different teachers could prefer different sequencing approaches, but there is not a didactic framework to help them in selecting the right sequencing algorithm. The question: "what is the best sequencing algorithm to use in a particular learning environment?" is a hard question because of the number of variables that could affect this choice. Here we propose a framework for comparing and testing different Sequencing algorithms to reason about

them in a self-contained and homogeneous environment. We extend the approach presented in [3] where the system LS-LAB is sketched. In LS-LAB different sequencing algorithms, belonging to different adaptive educational environments, are involved. These algorithms, through suitable software interfaces, e.g. parsers, run in the same environment, taking in input the same educational material, the same student model, the same goal for the student's course. The different generated courses are presented to the teacher, highlighted by a set of measures that could suggest and support her evaluation.

2 The LS-Lab System

The design of LS-LAB and its functional schema have been presented in [3]. Here we give a synthetic description of the system. Once an algorithm has been added to the system, the *GUI* allows to perform *experiments*. An experiment consists in selecting i) an algorithm (or more, if available), ii) a learning domain, i.e. a set of learning materials tagged with prerequisites and acquired knowledge, iii) a target knowledge, iv) a student model, v) the metrics to be used, and then activating the selected algorithms, so to produce, accordingly, comparable learning sequences for the student (model). The algorithms run on the same input, suitably adapted for each of them. The algorithms presently integrated into LS-LAB are used in the LS-PLAN system [2], in the KBS-Hyperbook system [1], and in the IWT system [5].

Two basic attitudes could be considered for the teacher's assessment of a Learning Objects Sequence (*LOS*). In a *subjective comparison* attitude, the teacher is left to judge the suitability of the sequence. We concentrate instead on a more *objective comparison* attitude: we operate through the following three metrics in order to measure certain characteristics and qualities of the *LOS* and offer the results to support teacher's *LOS* evaluation.

Overall_Effort metrics \mathcal{M}_E : One possible way to measure a *LOS* is by computing the cognitive effort implied by the *LOs* in the sequence. We have defined the *effort* as a value associated to a *LO*, that might represent the time expected to study the *LO*, or the complexity of such contents. The metrics \mathcal{M}_E compares *LOSes* basing on the overall effort required by their respective set of *LOs*.

Overall_Acquired_Knowledge metrics \mathcal{M}_{AK} : This metrics allows to compare *LOSes* by measuring how redundantly a *LOS* does actually cover the gap between the student's starting knowledge with respect to the topics to be learnt and the target knowledge of the course. It is the set of pieces of knowledge acquired studying the *LO* of the *LOS*. Of course a "more direct course" is not necessarily "simpler" in terms of \mathcal{M}_E .

Overall_p-effort metrics \mathcal{M}_{p-eff} : It measures the "cognitive distance" between a *LO* of the sequence, and its prerequisites, by measuring "how recently" the prerequisites for studying a given *LO* have been acquired. The more the prerequisites have been recently acquired, the less the \mathcal{M}_{p-eff} .

3 A First Experiment

We used the system for comparing *LOSes* produced (they are below) by the three available algorithms, for a given student model in the *Recursion* domain.

KBS	LS-PLAN	IWT
$\mathcal{M}_E = 13$ (effort)	$\mathcal{M}_E = 16$ (effort)	$\mathcal{M}_E = 13$ (effort)
$\mathcal{M}_{p-eff} = 2.25$ (distance)	$\mathcal{M}_{p-eff} = 3.00$ (distance)	$\mathcal{M}_{p-eff} = 1.75$ (distance)
id1:Unit description	id1:Unit description	id1:Unit description
id2:Recursive programs	id2:Recursive programs	id2:Recursive programs
id3:Rec.Funct. intro	id4:Rec.Funct. intro	id9:Rec. r/t stack examples
id5:Rec.Funct. StrgReverse	id5:Rec.Funct. StrgReverse	id3:Rec.Funct. intro
id6:Rec.Funct. examples	id6:Rec.Funct. examples	id5:Rec.Funct. StrgReverse
id9:Rec. r/t stack examples	id9:Rec. r/t stack examples	id6:Rec.Funct. examples
id10:Recursion exercises	id14:Recursive list	id10:Recursion exercises
	id10:Recursion exercises	

Note that LS-PLAN has one additional *LO* (id 14) and, consequently a bigger effort; all the three sequences present the same *LOs*, proposed in different order, and all the learning paths are logically consistent with the prerequisite relations in the learning domain; id3 and id4 are two alternative *LOs*, i.e. they have same prerequisites and acquired knowledge, but they have different learning styles: IWT and LS-PLAN have a different methodology for selecting alternative *LOs*, consequently they choose id3 and id4 respectively. On these bases the teacher has some elements for judging and comparing the behavior of the algorithms.

References

1. Henze, N., Nejd, W.: Adaptation in open corpus hypermedia. *International Journal of Artificial Intelligence in Education* 12(4), 325–350 (2001)
2. Limongelli, C., Sciarone, F., Temperini, M., Vaste, G.: Adaptive Learning with the LS-Plan System: a Field Evaluation. *IEEE Trans. on Learning Technologies* 2(3), 203–215 (2009)
3. Limongelli, C., Sciarone, F., Vaste, G.: LS-Lab: A framework for comparing curriculum sequencing algorithms. In: *Proc. of 9th Int. Conf. of intelligent Systems, Design and Application, ISDA 2009* (2009)
4. McArthur, D., Stasz, C., Hotta, J., Peter, O., Burdorf, C.: Skill-oriented task sequencing in an intelligent tutor for basic algebra. *Instr. Science* 4(17), 281–307 (1988)
5. Sangineto, E., Capuano, N., Gaeta, M., Micarelli, A.: Adaptive course generation through learning styles representation. *Universal Access in the Information Society* 7(1), 1–23 (2008)
6. Stern, M.K., Woolf, B.P.: Curriculum sequencing in a web-based tutor. In: Goettl, B.P., Half, H.M., Redfield, C.L., Shute, V.J. (eds.) *ITS 1998. LNCS*, vol. 1452, pp. 584–593. Springer, Heidelberg (1998)

Correcting Scientific Knowledge in a General-Purpose Ontology

Michael Lipschultz and Diane Litman

Department of Computer Science, University of Pittsburgh,
Pittsburgh PA 15260, USA
{lipschultz,litman}@cs.pitt.edu

Abstract. General-purpose ontologies (e.g. WordNet) are convenient, but they are not always scientifically valid. We draw on techniques from semantic class learning to improve the scientific validity of WordNet’s physics forces hyponym (IS-A) hierarchy for use in an intelligent tutoring system. We demonstrate the promise of a web-based approach which gathers web statistics used to relabel the forces as scientifically valid or scientifically invalid. Our results greatly improve the F1 for predicting scientific invalidity, with small improvements in F1 for predicting scientific validity and in overall accuracy compared to the WordNet baseline.

Keywords: Ontology, Semantic Web, Natural Language.

1 Introduction

An ontology is a formal definition of terms and the relationships between them [1]. An existing general-purpose natural language ontology called WordNet [2] has been successfully used in various tutoring systems [3,4]. We are interested in augmenting a dialog-based physics tutoring system with an ontology of physics terms so it can identify partially correct student responses and possibly offer different remediations based on the level of incorrectness (e.g. too vague or too specific). Prior work suggests that tutoring systems that detect partially correct responses and remediate differently may improve learning [5].

By using WordNet, we would not need to construct our own physics ontology. However, general-purpose ontologies, such as WordNet, contain mistakes in scientific domains [6], causing some researchers to construct their own domain-specific ontology [7]. We describe a method for automatically correcting an existing general-purpose ontology. Our scientific domain is physics forces and the general-purpose ontology is WordNet. The method identifies scientifically invalid terms contained within WordNet’s hierarchy for physics forces by using information on the web, thus improving the scientific correctness of the ontology. Other work on correcting WordNet begins by specifying formal properties that an ontology should have, then considers any violations of the properties an error to fix in the existing ontology [8]. However, this method requires human effort to correct while our method does not.

2 Method

We work with the hyponym (IS-A) hierarchy for the physics meaning of “force”. An expert tagged each of the hyponyms for scientific correctness. Of the 75 unique terms in the hyponym hierarchy, 29 were considered scientifically invalid (called *invalid* later) and 46 were considered scientifically valid (*valid*). This original WordNet is our baseline.

To improve WordNet, we need to classify whether an existing hyponym of “force” is *invalid* or *valid*. Our construction of the classifier is similar to work in semantic class learning [9], where the goal is to learn new terms for a particular topic in a domain. Learning new terms requires a pattern template containing a wildcard where the term-to-learn will go [10] and a corpus of text to search through. The corpus is then searched for instances of the pattern template and for each match, the term that replaces the wildcard is extracted. A corpus can be either domain-specific [11] or the entire web [9]; the first can be more reliable, but the second requires less effort to create and may be larger.

In our case, we already have terms, but wish to determine whether or not they are scientifically valid. Our corpus is all .edu websites to focus the search on sites we believe will use physics terminology correctly. In this paper, we found that simply searching for the term within quotes (i.e. $\langle \textit{“term”} \rangle$) to be the pattern that performed best. We then use Google to search our corpus and count the number of results returned.

3 Results

We construct 13 classifiers from the data collected as described in the previous section. These classifiers differ only in their threshold, ranging from 0 to 5,000,000. We chose the highest threshold to be larger than the highest result count. Those forces having counts above the threshold for the classifier are labeled as *valid* and those below the threshold are relabeled *invalid*.

For relabeling forces as *invalid*, the classifier outperforms the baseline. The baseline does not predict any scientifically invalid forces, so precision, recall, and F1 are all 0. All classifiers with thresholds at least 1,000 outperform the baseline. Recall and F1 increase as the threshold increases, reaching maxima of 1.00 and 0.5577 respectively. Precision plateaus at around 0.70 between thresholds 1,000 and 10,000, before dropping. So, while high thresholds are best for recall and F1, if precision is of greatest importance, then a threshold of 10,000 is best.

For overall accuracy and relabeling forces as *valid*, lower thresholds provide the greatest performance. Recall drops as the threshold increases. The greatest precision (0.6615) and F1 (0.7748) occur at threshold 10,000 (recall is 0.9348). Accuracy is around baseline, with a peak (66.67%) at threshold 10,000. So, while we are also able to improve over our baseline for labeling scientific validity and overall accuracy, the improvement is not as large.

4 Conclusions and Future Work

In this paper, we modified a method from semantic class learning to correct the scientific validity of WordNet's hyponym hierarchy of physics forces. We saw that the simple pattern <“*term*”> was able to label scientifically invalid forces. A threshold of 10,000 provides the best F1 for labeling scientific validity and for overall accuracy, while providing a good F1 for scientific invalidity. However, higher thresholds improve F1 for scientific invalidity.

In future work, we plan to further improve our method by exploring other algorithms, patterns, corpora, and ontologies. We also plan on addressing the incompleteness of scientific knowledge in general-purpose ontologies. Finally, we want to incorporate the corrected ontology into our tutoring system to automatically detect partially correct responses. For example, if a student answered a question with “a force” (when the correct answer was “force of gravity”), the system could respond with “You're close. Which force is acting on the keys?” instead of its current response “I disagree with you. The force of gravity is acting on the keys.”.

Acknowledgements. The authors thank Art Ward, Pam Jordan, Wenting Xiong, and Joanna Drummond for their input. We also thank Guangtian Zhu and Chandralekha Singh for their help in creating the gold standard.

References

1. Hendler, J.: Agents and the semantic web. *IEEE Intelligent systems* 16(2) (2001)
2. Fellbaum, C., et al.: *WordNet: An electronic lexical database*. MIT Press, Cambridge (1998)
3. Brown, J., Frishkoff, G., Eskenazi, M.: Automatic question generation for vocabulary assessment. In: *Proc. of the Conference on HLT and EMNLP*. Association for Computational Linguistics (2005)
4. Ward, A., Litman, D.: *Semantic Cohesion and Learning*. In: *Proc. 9th ITS* (2009)
5. Jordan, P., Litman, D., Lipschultz, M., Drummond, J.: Evidence of Misunderstandings in Tutorial Dialogue and their Impact on Learning. In: *Proc. of AIED* (2009)
6. McCrae, J., Collier, N.: Synonym set extraction from the biomedical literature by lexical pattern discovery. *BMC bioinformatics* 9(1), 159 (2008)
7. Christopher, B., Simon, J., Joanne, L., David, S., Robert, S., Ziqi, Z.: Issues in learning an ontology from text. *BMC Bioinformatics* 10 (2009)
8. Gangemi, A., Guarino, N., Oltramari, A.: Conceptual analysis of lexical taxonomies: The case of WordNet top-level. In: *Proc. Intl. Conf. on Formal Ontology in Information Systems*. ACM, New York (2001)
9. Kozareva, Z., Riloff, E., Hovy, E.: Semantic class learning from the web with hyponym pattern linkage graphs. In: *Proc. ACL 2008, HLT* (2008)
10. Hearst, M.: Automatic acquisition of hyponyms from large text corpora. In: *Proc. of ACL* (1992)
11. Cimiano, P., Pivk, A., Schmidt-Thieme, L., Staab, S.: Learning taxonomic relations from heterogeneous sources of evidence. In: *Ontology Learning from Text: Methods, evaluation and applications*, pp. 59–73 (2005)

Learning to Argue Using Computers – A View from Teachers, Researchers, and System Developers

Frank Loll¹, Oliver Scheuer², Bruce M. McLaren², and Niels Pinkwart¹

¹ Clausthal University of Technology, Department of Informatics, Germany
{frank.loll,niels.pinkwart}@tu-clausthal.de

² German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany
{oliver.scheuer,bmclaren}@dfki.de

Abstract. The ability to argue is essential in many aspects of life, but traditional face-to-face tutoring approaches do not scale up well. A solution for this dilemma may be computer-supported argumentation (CSA). The evaluation of CSA approaches in different domains has led to mixed results. To gain insights into the challenges and future prospects of CSA we conducted a survey among teachers, researchers, and system developers. Our investigation points to optimism regarding the potential success and importance of CSA.

Keywords: Argumentation, Survey, CSCL.

1 Introduction

The ability to argue is essential in many aspects of life. However, people often struggle to engage in reasoned arguments [1], making the acquisition of argumentation skills an important educational goal. Classical face-to-face teaching methods are limited due to teacher time and availability. To remedy these limitations, computer-based argumentation systems have been developed. In a detailed review of 50 argumentation systems and methods [2], we recently surveyed the existing approaches. Here, we want to extend this review with a view behind the scenes.

2 A Web-Based Survey: Description and Evaluation

Table 1 contains a set of research questions (RQ) that we are interested in. In this paper, we provide (at least partial) answers to these research questions based on a web-based survey conducted with argumentation researchers, teachers and system developers. Participation was voluntary. The participants were informed about the purpose of the survey and the use of the data. As motivation, we offered the anonymized results of the survey and raffled an iPod among all participants. As part of the survey, participants were asked for their background, including their experience with research, teaching and designing/developing of argumentation systems (on a 5pt Likert scale) and their primary domain of expertise. Participants were then asked (among others) the questions listed in table 2. Typically, two items were designed to measure the same thing in order to test for consistency. We refer to these pairs as “question” and “control question”. Table 2 presents one of each pair.

Table 1. Research questions

No.	Question (Abbreviation)
RQ1	Are visual argument representations helpful for learning and/or understanding argumentation?
RQ2	Can computer-supported / computer-mediated argumentation replace face-to-face argumentation?
RQ3	Does the formality of a domain influence the type of collaboration that is appropriate?
RQ4	Do argumentation researchers, teachers and system developers differ in their views on the suitability of collaboration for argument learning?
RQ5	Is it possible to develop automated analysis features that can effectively analyze arguments and are there any domain-specific differences?
RQ6	When is tutorial feedback most effectively provided?

Table 2. Survey questions

No.	Question
Q1	Which answer best describes the type of argumentation that is taught and/or used in your primary domain of interest? (<i>Scale: 1 = Informal, ..., 5 = Formal, 6 = I do not know</i>)
Q2	In my primary domain of interest, it is important that people learn argumentation through discussions, rather than on their own (e.g., from a book, by sketching arguments on paper, etc).
Q3	In my primary domain weaknesses and errors in arguments can be identified by general and recurring patterns.
Q4	It is possible to assess the quality of an argument just by analyzing general patterns in the argument.
Q5	Arguments shown in graphical fashion are likely to be helpful in learning or understanding argumentation.
Q6	Computer systems have the potential to support people in conducting useful, valid arguments over the Internet, perhaps even improving upon standard, face-to-face discussion.
<i>Scale for Q2-Q6: 1 = Strongly disagree ... 5 = Strongly agree, 6 = I do not know</i>	
Q7	Feedback on errors and problems when engaging in argumentation learning is most effectively provided... (<i>Scale: 1 = Immediately after the error or problem occurs, 2 = After the argument is over, 3 = Only when the participants explicitly ask for feedback, 4 = ... (Other), 6 = I do not know</i>)

In total, we received 97 responses. Among them there were 2 participants who self-reported two domains of expertise; thus we counted their responses for both. We excluded all participants with a self-reported experience score below 3 in all domains. To answer the RQs, we calculated means, standard deviations, Spearman-Rho correlations and an ANOVA after filtering “I do not know” responses. For the ANOVA between different argumentation domains, we created an “others” group combining all domains containing only a few, i.e. three or less, participants.

To answer RQ1, we used the results of question Q5 (Correlation with control question: $\rho=.519$ with $p=.000$ and $n=97$). The mean results was $m=4.24$ ($sd=.63$, $n=97$). In particular, there was only one score below 3 in Q5. Thus, there is a strong agreement that visual representations of arguments are indeed helpful for understanding and reflecting upon, and hence learning. There were no significant differences between argumentation domains on this question. Concerning RQ2, our respondents consider computers useful to support argumentation (Q6: $m=3.98$, $sd=.94$, $n=92$), but not to the extent that face-to-face argumentation could be entirely *replaced*, as suggested by the more strongly formulated control question ($m=3.28$, $sd=1.18$, $n=93$). Again, there were no domain-specific differences on this question. Concerning RQ3, formality

(Q1) did not correlate with either a preference for an individual or a collaborative argumentation learning approach when considering all domains. However, in the domain “Education” a preference for individual learning was found when argumentation is more formal ($\rho=.63$, $p=.007$; $n=17$). Concerning RQ4, experience with research, teaching and development did not correlate with the participants’ views on individual or collaborative argumentation. Regarding RQ5, our experts showed a tendency to believe in the existence of general recurring patterns that indicate errors and weaknesses in their domain of interest ($m=3.66$, $sd=.78$, $n=87$), a tendency that increases with the amount of teaching experience ($n=85$, $\rho=0.23$, $p=.038$), but independent from the concrete domain. RQ6 deals with the question when to react on students’ errors and misconceptions. Here, most experts think that feedback is best provided immediately following the error or problem that occurs ($n=34$). However, nearly the same number of people stated that it is most effectively provided after the argument is over ($n=28$). A considerable number of experts ($n=13$) proposed other approaches that mostly depend on the situation. Only few participants ($n=3$) preferred feedback on request.

In conclusion, there is considerable agreement among the experts that argumentation systems are able to facilitate learning via argument visualization techniques. Nevertheless, the questions whether individual or collaborative argumentation is more beneficial for learning could not be clearly answered. Another open issue is the future and application potential of computer-based analysis and feedback on argumentation, especially in less structured argumentation domains.

Acknowledgments

This work was supported by the German Research Foundation (DFG) under the grant LASAD – Learning to Argue: Generalized Support Across Domains. We would like to thank all respondents and beta testers of the questionnaire.

References

1. Kuhn, D.: *The Skills of Argument*. Cambridge University Press, Cambridge (1991)
2. Scheuer, O., Loll, F., Pinkwart, N., McLaren, B.M.: Computer-Supported Argumentation: A Review of the State of the Art. *International Journal of Computer-Supported Collaborative Learning* 5, 43–102 (2010)

How to Take into Account Different Problem Solving Modalities for Doing a Diagnosis?

Experiment and Results

Sandra Michelet, Vanda Luengo, Jean-Michel Adam, and Nadine Madran

Laboratory of Informatics, Grenoble (LIG)
961 rue de la Houille Blanche 38402 Grenoble, France
{Sandra.Michelet, Vanda.Luengo, Jean-Michel.Adam,
Nadine.Mandran}@imag.fr

Abstract. We are interested in cognitive diagnosis systems able to understand the learners' work in environments involving them in various problem solving modalities. We are designing a diagnosis model taking into account various factors. In this paper we are interested in the problem solving modality factor and we present an experiment for analyzing the impact of it on the diagnosis.

Keywords: Adapted diagnosis, problem solving modality.

1 Introduction

Several platforms (ActiveMath [2] and SCY [1]) propose various problem solving modalities associating different tools connected to a learning activity. Indeed, this variety of proposed tools modify the diagnosis: information about the learner's activity comes from different sources and need to be combined. We thus research to characterize factors to be taken into account for the diagnosis in this type of environment. In this paper we are interested in the problem solving modality factor.

Moreover, in a previous work, we showed that for some learners, there was a contradiction between different modalities for the same problem [3]. These results motivated us to associate a believe degree to these modalities.

To evaluate the impact of this modality factor, we did an experiment. In this paper we present various results from this experiment.

DiagElec: a model of Diagnosis in Electricity. In our environment the learner solves the problems either by completing a *MCQ* (M), expressing a *formulation* in natural language (F), and/or working with *microworld* TPElec¹ (T). After some semi automatic treatments of the activity trails (we don't analyze directly the formulations given in natural language by the learner), we obtain a set of prolog facts. We use these facts for producing a cognitive diagnosis [4]. In our system we distinguish three diagnosed elements: knowledge (k_i), skill (s_j) and error (e_l). DiagElec integrates also a

¹ TPElec. <http://tpelec.imag.fr/>

concept of uncertainty by associating a degree of belief between 1 (lowest belief) to 4 (highest belief). A diagnosed modality vector for the learner L1 when s/he was working with the TPElec and the formulation tools, solving the P2 problem, could be: $V_{L1-P2-TPElecForm} = \{(k11,2), (e7,3), (s6,4)\}$. We want to know the impact of the problem solving modalities on the diagnosis: do they help for detecting the diagnosis elements? Do they have the same importance for the diagnosis?

2 Impact of Modalities: Experiment, Analysis and Results

The experiment was done in two steps. The first one, involving 60 learners, was the collection of the learners’ activity trails. The proposed problems were built on the model of scientific reasoning. The second one was realized with teachers: they were asked to diagnose the learners’ productions. For this experiment, we collaborated with 3 Physics teachers involved in our research project by the French National Institute of Educational Research (INRP).

Because we have 3 different tools we have compared the 7 combinations of these tools and, done a separate diagnosis for each combination. For feasibility reasons, we could not ask each teacher to do 7 diagnoses for all the 60 learners. We select, with a protocol link to our research hypothesis, 18 learners and we showed their productions to the teachers according to the 7 contexts. Each teacher had to identify the learner’s knowledge, skills and errors.

Diagnosis vector. For each element detected by at least one expert, we build a vector of belief: $V_Belief_{L-P-C-D}$. It consisting of the degrees of belief of experts E_1, E_2, E_3, E_4 , for the learner L solving the problem P, with the modality context C and the detected element D. We could obtain the following belief vectors: $V_Belief_{L1-P2-TPElec-e7} = \{(3,2,4,3)\}$ or $V_Belief_{L1-P2-TPElec-s21} = \{(0,1,0,0)\}$. The expert E_4 is DiagElec the three others are the teachers. We have got 18306 belief vectors.

Human Convergence. The human convergence is established using the following distance formula: $d = \sqrt{d1^2 + d2^2 + d3^2} / 3$ ($d1, d2$ and $d3$ are the differences between the teachers’ degrees of belief). There is *total human convergence* if $d = 0$, *partial convergence* if $0 < d \leq 0.5$ and *human divergence* otherwise.

Problem solving modalities	Number	Total Human Convergence		Partial Human Convergence		Human Divergence	
		Number	%	Number	%	Number	%
M	118	118	100,0%	0	0,0%	0	0,0%
F	1963	1105	56,3%	450	22,9%	408	20,8%
T	1599	1102	68,9%	414	25,9%	83	5,2%
M+F	3750	2918	77,8%	438	11,7%	394	10,5%
M+T	2913	2324	79,8%	463	15,9%	126	4,3%
F+T	3095	1907	61,6%	803	25,9%	385	12,4%
M+F+T	4868	3730	76,6%	767	15,8%	371	7,6%
TOTAL	18306	13204	72,13%	3335	18,22%	1767	9,65%

Fig. 1. Human convergence according to problem solving modalities

We observed (Figure 1) that the teachers mainly agree (72.13%). However, this agreement seems to be linked to the problem solving modality. Indeed, if we only consider the *formulation (F)*, we observe that the rate of divergence between teachers is the highest, as well as the rate of partial convergence.

Problem solving modality and detection of knowledge, skills and errors. In a second analysis level we kept only the vectors corresponding to a total human convergence. We focused on the number of knowledge elements, skills and errors detected by the teachers, according to the different modalities. If we observe the left graphic of Figure 2 we can say that taking into account several tools enriches the diagnosis; this conclusion was predictable. We can also observe that the modalities are linked to the type of the detected elements: the skills are more easily detected from the action of the learner on the TPElec microworld, while the knowledge is better detected in the learner’s formulations.

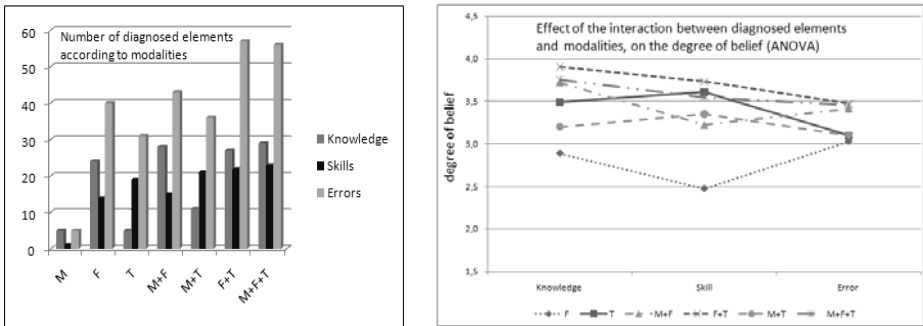


Fig. 2. Number of diagnosed elements (left) and ANOVA (right)

Problem solving modalities and degree of belief of the diagnosed elements. We applied an ANOVA analysis (Figure 2 right) in order to study the effect of the interaction between diagnosed elements (knowledge, skill, error) and problem solving modalities, on the degree of belief given by the human experts. The ANOVA shows a significant effect of the modalities ($F=294,6$, $df(5,13068)$, $p<0.001$), the diagnosed elements ($F=174,7$, $df(2,13068)$, $p<0.001$), and their interaction on the degree of belief ($F=38,8$, $DF(10,13068)$, $P<0.001$).

Conclusion. Our experimentation shows that taking the problem solving modality into account is significant and necessary in some ways. Moreover, in the case of contradictions intra-learners, we can identify which tool is more important for the diagnosis, in order to obtain a better adaptation [5]. We need to calibrate more finely the degree of belief given by DiagElec. For that purpose, we are planning to do an experiment with more teachers to do the calibration by a data mining process.

References

1. d'Ham, C., Marzin, P., Wajeman, C.: SCY-Science Created by You: an online environment for inquired-based and design-based learning. In: First Workshop on the S-Team European Project (2009)
2. Goduaez, G., Melis, E.: Combining Evaluative and Generative Diagnosis in ActiveMath. In: Artificial Intelligence in Education, AIED 2009 (2009)
3. Michelet, S., Adam, J.M., Luengo, V.: Adaptive learning scenarios for detection of misconceptions about electricity and remediation. *International Journal of Emerging Technologies in Learning* 2(1) (2007)
4. VanLehn, K.: Student modelling. In: Polson, Richardson (eds.) *Foundations of Intelligent Tutoring Systems*, Lawrence Erlbaum Associates Ltd., Hove (1988)
5. Keenoy, K., Levene, M., De Freitas, S.: Personalised Trails: How machine can learn to adapt their behaviour to suit individual learners. In: *Technology Enhanced Learning, Special on Trails in Education*, vol. 1, pp. 33–58. Sense publishers (2007)

Behavior Effect of Hint Selection Penalties and Availability in an Intelligent Tutoring System

Pedro J. Muñoz-Merino, Carlos Delgado Kloos, and Mario Muñoz-Organero

Carlos III University of Madrid, Department of Telematics Engineering,
Avda de la Universidad, 30, E-28911 Leganés (Madrid), Spain
{pedmume, munozm, cdk}@it.uc3m.es

Abstract. This paper presents empirical results about the behavior effect of two different hinting strategies applied on exercises within an ITS: having some penalty on the scoring for viewing hints or not having any effect on the scoring; and hints directly available or only available as a result to an incorrect attempt. We analyze the students' behavior differences when these hinting techniques changed, taking into account the type and difficulty of the presented exercises.

Keywords: assessment, hints, empirical evaluation, student behavior.

1 Introduction

Different student behaviours within hinting systems have been studied. Paper [1] shows a model identifying the ideal student behavior. The article [2] shows different useful parameters based on student actions. The different student behaviors have an effect on their learning gains [1], and the different student actions have a relationship with their final scores on the tests [2].

This paper focuses on the analysis of the student behavior regarding the variation of two hinting strategies, taking into account the type of problem and their difficulty. These hinting techniques imply some effort or cost for obtaining help. Whether the provision of help to students is beneficial or not, is not a trivial question [3].

Some existing hinting systems (e.g. SIETTE [4]) can adapt hint contents. The results of this paper can be used for example for the adaptation of the commented hinting techniques without changing the hint contents.

2 Preparation of the Experiment

The experiment took place in two editions of a computer architecture laboratory course. The data was taken from the hinting module [5] of XTutor during two sessions. Students interacted with a set of exercises with hints, changing the hinting techniques to compare. Exercises were multiple Choice (MC, only one option is correct), Multiple Response (MR, the correct solution involves the selection of several options) and Fill In the Blank (FIB). Four exercises (S1, M1, I1, F1) changed the strategy of having penalties for viewing hints or not. Another four exercises (S2, M2,

I2, F2) changed the strategy of having penalties for viewing hints, or not but with a maximum limit of hints to select. Finally, four exercises (S4, M4, I4, F4) changed the strategy of hint availability. Each student interacted only with one of these hinting techniques for a specific problem, and this is selected randomly at the beginning but in a way that there is a balance in the number of total presented hinting techniques.

3 Results and Analysis

Tables 1 and 2 show the statistics for the 12 exercises, comparing the hinting techniques regarding penalties for selecting hints (table 1) and hint availability (table 2). The tables distinguish the type of problem (MC, MR or FIB), their difficulty (in a scale from 0 to 4), the total number of interactions with each one of the techniques compared, the number of times a user selected at least one hint for each one of the techniques compared, and the probability of requesting at least one hint (P) provided in a confidence interval for each one of the hinting techniques (applying the binomial non-parametric test with a 95% probability). If there is not an intersection of the intervals of the two techniques compared for a specific exercise, then there is a statistically significant difference between both techniques for this specific exercise. For each specific problem, samples are independent. We are planning to analyze more hinting techniques and integrate the results with a logistic regression model taking into account the random effect of the same student answering different exercises.

Table 1. Statistics for penalties or not for selecting hints, for the eight exercises

Problem	Type	Diff.	Total Interact. with Penalties	Total Interact. without Penalties	Selected Hints with Penalties	Selected Hints without Penalties	P (%) Hint with Penalties	P (%) Hint without Penalties
S1	MC	1	48	46	1	3	[0, 10]	[1, 16]
M1	FIB	1	43	40	27	37	[49, 75]	[82, 98]
I1	FIB	3	43	43	35	35	[69, 91]	[69, 91]
F1	MC	3	42	47	7	13	[8, 29]	[17, 40]
S2	MR	4	47	45	30	42	[51, 75]	[84, 98]
M2	MR	3	42	42	25	34	[45, 73]	[68, 90]
I2	MR	2	44	41	19	32	[31, 57]	[65, 88]
F2	MR	1	37	36	11	11	[18, 45]	[18, 45]

Table 2. Statistics for hints directly available or not for the four exercises

Problem	Type	Diff.	Total Interact. no available	Total Interact. directly available	Selected Hints no available	Selected Hints directly available	P (%) Hint no available	P (%) Hint directly available
S4	FIB	3	39	46	23	34	[45, 72]	[61, 84]
M4	FIB	4	36	40	27	37	[60, 86]	[82, 98]
I4	FIB	3	28	41	19	29	[51, 82]	[57, 82]
F4	FIB	1	30	37	6	22	[9, 36]	[44, 73]

From table 1, there is a significant difference between the number of hints selected, comparing the cases of having penalties or not, when the amount of selected hints is not too high nor too low. As MC problems make the number of requested hints to be low then none of the cases resulted in a significant difference. As FIB problems make the number of requested hints to be high, then if the problem is difficult the amount of requested hints will be high, and there will not be significant difference; if it is easy or medium, then the number of requested hints will be medium and there will be significant difference. Finally, the MR problems have also certain effect that makes students to do a lot of attempts until they reach the correct answer, but they need much more attempts than for MC problems. So if the problem is easy then there will be few requested hints and the difference will not be significant, but if the problem is medium or difficult then the amount of requested hints will not be too high nor too low, so this can make the difference significant. A similar effect happens with the hinting technique of having hints directly available or not. In this case, it was only tested with FIB problems. A similar analysis can be performed from table 2.

4 Conclusions

There were several statistically significant differences in the number of hint requests between the hinting technique of having penalties for selecting hints or not, depending on the type and difficulty of the problem. In a similar way, there was a statistically significant difference in the number of hint requests comparing hints directly available or not, depending on the problem difficulty. The probabilities for selecting hints were provided for each case, as confidence intervals.

Acknowledgments. Work partially funded by the Learn3 project TIN2008-05163/TSI within the Spanish “Plan Nacional de I+D+I”, and the Madrid regional community project eMadrid S2009/TIC-1650.

References

1. Alevén, V., McLaren, B., Roll, I., Koedinger, K.R.: Toward Meta-cognitive Tutoring: A Model of Help Seeking with a Cognitive Tutor. *Int. J. Artif. Intell.* 16, 101–128 (2006)
2. Feng, M., Heffernan, N.T., Koedinger, K.R.: Predicting state test scores better with intelligent tutoring systems: developing metrics to measure assistance required. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) *ITS 2006*. LNCS, vol. 4053, pp. 31–40. Springer, Heidelberg (2006)
3. Beck, J.E., Chang, K., Mostow, J., Corbett, A.: Does Help Help? Introducing the Bayesian Evaluation. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 540–550. Springer, Heidelberg (2008)
4. Conejo, R., Guzmán, E., Pérez de-la Cruz, J.L., Millán, E.: An Empirical Study About Calibration of Adaptive Hints in Web-Based Adaptive Testing Environments. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) *AH 2006*. LNCS, vol. 4018, pp. 71–80. Springer, Heidelberg (2006)
5. Muñoz-Merino, P.J., Delgado Kloos, C.: A software player for providing hints in problem-based learning according to a new specification. *Computer Applications in Engineering Education* 17, 272–284 (2009)

DesignWebs: A Tool for Automatic Construction of Interactive Conceptual Maps from Document Collections

Sharad V. Oberoi, Dong Nguyen, Gahgene Gweon,
Susan Finger, and Carolyn Penstein Rosé

Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA, 15213
{svo, dongn, gkg, sfinger, cp3a}@andrew.cmu.edu

Abstract. Prior work supports the pedagogical value of conceptual maps for offering students an overview of a topic as well as the connections between sub-topics. In this poster we describe a system that uses automated topic modeling technology to map the topics and sub-topics in a collection of documents. An interactive graphical representation allows users to explore this topic analysis, using it as an interface for browsing a collection of documents. We present a small user study evaluating the usability of the interactive map.

Keywords: conceptual maps, graphical representations, language processing.

1 Introduction

This poster presents a framework that supports knowledge management for project teams. An important part of many group projects is becoming aware of the state-of-the-art in areas relevant to the design or development goal. As supporters of the learning process, instructors can raise student awareness of relevant knowledge and support student integration of that knowledge. However, it may still occur with their limited experience that students miss important connections between concepts. We describe a working system that constructs conceptual maps automatically from document collections, such as from the articles found through Google Scholar. These conceptual maps are referred to as DesignWebs [1]. DesignWebs are also a navigation aid since the nodes in the map link directly to the source documents.

2 Creating DesignWebs

Providing concept maps to students has been suggested as a metacognitive tool to enhance their learning in the sciences [2,3]. Topic modeling approaches identify high-level topics present in a document collection. One such method is Latent Dirichlet Allocation (LDA)[4], which is a generative process that models each document as a mixture of topics, and models each topic as a multinomial distribution over words. To construct a DesignWeb, the instructor needs to gather a collection of relevant documents and start the automatic preprocessing script. The result is a DesignWeb not

only showing the main topics within a given set of documents, but also allowing users to zoom-in to view sub-topics and the terms relevant to these topics (See Figure 1). Users can also browse the relevant documents.

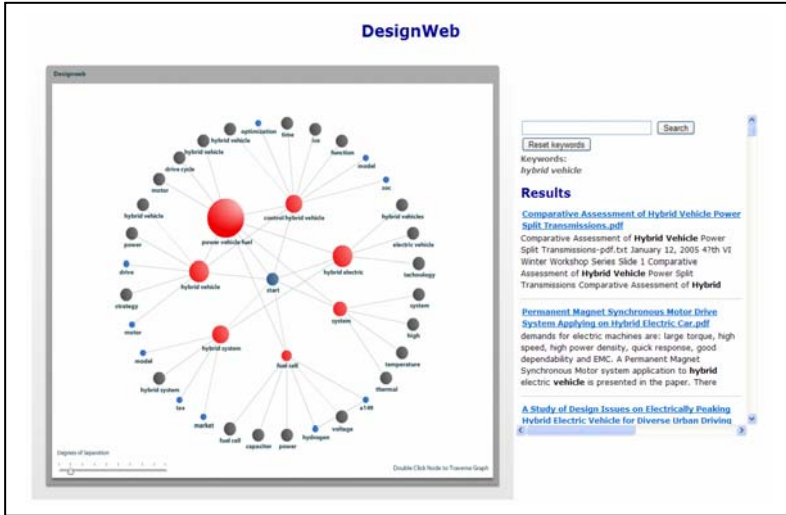


Fig. 1. Screenshot of the DesignWeb interface

The main steps of our technical approach are:

1. We first apply LDA to model the topics in a document collection. After the extraction of the topic model, a hierarchy for every topic is induced using hierarchical clustering.
2. Term lists associated with topics were compressed by collapsing terms that had similar topic distributions and could be reduced to the same stem after removing any morphological endings. In order to include common phrases in the vocabulary list, we extract the documents most strongly associated to each topic and then extract the collocations from this set. Clusters are labeled by counting unigrams and bigrams in the set of most relevant documents of the cluster and choosing the label the one that has the highest count.
3. Connections between topics are computing by first representing each topic as a vector of probabilities that represent the topic’s associated word distribution. Cosine similarity is then computed for each pair of topics, and if it is higher than a threshold (0.2), the topics are considered to be linked.
4. Each node is associated with a topic in the topic model.

Clicking on a node in the Design Web issues a query that retrieves the documents most strongly associated with the topic. This query is treated as though it contains all the terms most strongly associated with the topic so that when the documents are retrieved, and snippets are displayed for each, a snippet will be selected that contains a high concentration of those terms. Additional terms can be added to this query in

order to influence the displayed snippets. We use the Lingpipe framework¹ for extracting the LDA model, collocations and applying the clustering. Lucene² is used for the document retrieval functionality.

3 Informal Evaluation and Current Work

We conducted a small user study to evaluate how well students are able to use Design Webs to explore a document collection. The scenario for our user study is a class of engineering graduate students about to embark on a project involving an analysis of hybrid cars. To simulate a typical a literature review task at the start of such a class, we downloaded 250 articles from Google Scholar and automatically generated a DesignWeb from these documents. The students were given a demonstration of the system using a DesignWeb created from a different corpus and allowed to acquaint themselves with it. Then, they were asked to identify alternative battery types and alternative energy sources for hybrid cars from the DesignWeb, along with the pros and cons of each. The students had to cite the sources and not use any prior knowledge. The students' answers for the alternative battery types and energy sources varied between 3 and 7 choices each, with an average of 5 choices per student. The number of sources cited varied between 3 and 10 (average: 5.4), while the total number of advantages and disadvantages cited was between 5 and 10 (average: 7.4).

DesignWebs provide a robust and automatic method to organize, navigate and synthesize the documents referenced by students during a project. These are expected to support learning tasks by providing a bird's eye-view that is otherwise not possible due to information scattered in research literature that is needed for the task.

This work was supported by NSF Grants EEC-0935127 and EEC-064848.

References

1. Oberoi, S.V., Finger, S.: DesignWebs: An Interactive Organizational Memory Assimilation and Navigation Tool. In: 17th International Conference on Engineering Design, Stanford, CA, August 24-27 (2009)
2. Horton, P.B., McConney, A.A., Gallo, M., Woods, A.L., Senn, G.J., Hamelin, D.: An Investigation of the Effectiveness of Concept Mapping as an Instructional Tool. *Science Education* 44, 95–111 (1993)
3. Lawless, C., Smee, P., O'Shea, T.: Using Concept Mapping and Concept Mapping in Business and Public Administration, and in Education: An Overview. *Educational Research* 40(2), 219–235 (1998)
4. Blei, D.M., Griffiths, T.L., Jordan, M.I., Tenenbaum, J.B.: Hierarchical Topic Models and the Nested Chinese Restaurant Process. In: *Advances in Neural Information Processing Systems*, p. 2003 (2004)

¹ Lingpipe: <http://alias-i.com/lingpipe/>

² Lucene: <http://lucene.apache.org/>

Extraction of Concept Maps from Textbooks for Domain Modeling

Andrew M. Olney*

University of Memphis, Memphis TN 38152, USA
aolney@memphis.edu
<http://iis.memphis.edu>

Abstract. Previous research using concept maps as domain models in intelligent tutoring systems has demonstrated their power and flexibility. However, these concept maps must still be authored by a domain expert, creating a development bottleneck. We present a new, streamlined methodology for automatically extracting concept maps from textbooks using term extraction, semantic parsing, and relation classification.

Keywords: concept map, domain model, semantic parsing.

1 Introduction

In an intelligent tutoring system (ITS), the representation of subject matter knowledge is often referred to as a domain model. The domain model is an integral part of an ITS and is typically strongly connected both the model of the student's knowledge (student model) and the model of how to teach the subject matter (pedagogical/expert model). As exemplified by CIRCSIM-Tutor and Betty's Brain, concept maps can be used as both domain models and overlay student models as well as to interpret student utterances, generate explanations, and perform qualitative reasoning [1,2]. However, in both CIRCSIM-Tutor and Betty's Brain, expert concept maps need to be authored. In this paper we outline a new approach to concept map extraction from textbooks. Our approach is significantly streamlined relative to previous approaches [3,4]. In what follows we define our concept map representation and then describe our approach to extracting concept maps from a textbook.

2 Definitions

Our concept map definition is a blend of previous work in the psychology and education literatures [5,6,7]. We adopt a formulation of concept maps largely

* The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A080594 to the University of Memphis. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education.

consistent with the SemNet formulation [7]. This leads to maps with one layer of links radiating out of a core concept. In our representation, only key terms can be the start of a triple (equivalently the center of a map). End nodes can contain key terms, other words, or complete propositions. In addition, our representation uses a restricted set of labeled edges. As noted by Fisher [7], a small set of edges account for a large percentage of relationships. Moreover, having a prescribed set of edges facilitates linkages between the map representation and question asking/answering [5,6].

While generalized term extraction procedures exist [8], such methods tend to be less relevant in a pedagogical context where key terms are often already provided, whether in glossaries [9], textbook indices [10], study guides, etc. To develop our key terms, we used the glossary and index from a textbook as well as the keywords in a test-prep study guide, yielding approximately 3,000 terms. Thus we can skip the keyword extraction step of previous work on concept map extraction [3,4].

We obtained general relations from previous work [5,6,7,11] but augmented these with relations specific to our biology domain through manual analysis. We manually analyzed and clustered 4371 biology triples available on the Internet¹ to a set of 30 relations, including *after*, *contrast*, *enable*, *has-consequence*, *lack*, *produce*, *before*, *convert*, *example*, *has-part*, *location*, *purpose*, *combine*, *definition*, *extent*, *has-property*, *manner*, *reciprocal*, *connect*, *direction*, *follow*, *implies*, *not*, *require*, *contain*, *during*, *function*, *isa*, *possibility*, and *same-as*. Previous approaches to concept map extraction do not appear to use a fixed set of relations [3,4], making them less suited to question asking/answering techniques [5,6].

3 Extraction

Given a textbook as input, the LTH SRL Parser [11] outputs a dependency parse annotated with semantic roles derived from Propbank and Nombank, i.e. part of speech, lemma, head, and relation to the head, verbal predicates, nominal predicates, and associated arguments. Use of a semantic parser significantly streamlines the extraction of concept maps, which previous approaches have addressed using syntactic parsing with regular expressions, or word/relation extraction through phrase identification [3,4].

For each syntactic or semantic relation found by the parser, we require that the start node be a key term. Several relations are handled purely syntactically, including *is-a* via “be,” *has-property* via adjectives, and *location* via prepositions. Relations from Propbank and Nombank require examination of several features in order to determine the relationship between the arguments, including the lexical form, the gloss for the roset of the predicate, the label given to the argument, and the gloss given to the argument. These features are input to a manually designed decision tree, which inspects the features by priority and assigns a relation.

¹ <http://www.biologylessons.sdsu.edu>

Using this approach, we extracted 28,994 relations from a thousand page textbook. These relations were distributed around 1,886 key terms out of approximately 3,000. The mean number of relations per term is 15.4, but the variation is quite high (min 1, max 552, sd 31.7). The five most connected key terms are *animals*, *cell*, *species*, *genes*, and *blood*. We extracted 27 relations of 30, excluding *lack*, *requires*, and *same-as*. The top five relations extracted were *has-property* *has-consequence*, *isa*, *manner*, and *location*, making up roughly 80% of the total relations. The relations *has-property*, *is-a*, and *has-part* make up 52% of the total, which is consistent with reported human concept maps for biology domains [7]. Our future work will expand our evaluation of the raw maps as well as their application to question asking/answering and domain models.

References

1. Evens, M., Brandle, S., Chang, R., Freedman, R., Glass, M., Lee, Y., Shim, L., Woo, C., Zhang, Y., Zhou, Y., Michael, J., Rovick, A.: CIRCSIM-Tutor: An intelligent tutoring system using natural language dialogue. In: Proceedings of the 12th Midwest AI and Cognitive Science Conference (MAICS 2001), Oxford, OH, pp. 16–23 (2001)
2. Leelawong, K., Biswas, G.: Designing learning by teaching agents: The Betty's Brain system. *Int. J. Artif. Intell. Ed.* 18(3), 181–208 (2008)
3. Zouaq, A., Nkambou, R.: Evaluating the generation of domain ontologies in the knowledge puzzle project. *IEEE Trans. on Knowl. and Data Eng.* 21(11), 1559–1572 (2009)
4. Valerio, A., Leake, D.B.: Associating documents to concept maps in context. In: Canas, A.J., Reiska, P., Ahlberg, M., Novak, J.D. (eds.) Proceedings of the Third International Conference on Concept Mapping (2008)
5. Graesser, A.C., Franklin, S.P.: Quest: A cognitive model of question answering. *Discourse Processes* 13, 279–303 (1990)
6. Gordon, S., Schmierer, K., Gill, R.: Conceptual graph analysis: Knowledge acquisition for instructional system design. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 35(3), 459–481 (1993)
7. Fisher, K., Wandersee, J., Moody, D.: Mapping biology knowledge. Kluwer Academic Pub., Dordrecht (2000)
8. Medelyan, O., Frank, E., Witten, I.H.: Human-competitive tagging using automatic keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, August 2009, pp. 1318–1327. Association for Computational Linguistics (2009)
9. Navigli, R., Velardi, P.: From glossaries to ontologies: Extracting semantic structure from textual definitions. In: Proceeding of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge, Amsterdam, The Netherlands, pp. 71–87. IOS Press, Amsterdam (2008)
10. Larrañaga, M., Rueda, U., Elorriaga, J.A., Lasa, A.A.: Acquisition of the domain structure from document indexes using heuristic reasoning. *Intelligent Tutoring Systems*, 175–186 (2004)
11. Johansson, R., Nugues, P.: Dependency-based syntactic-semantic analysis with PropBank and NomBank. In: CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning, Morristown, pp. 183–187. Association for Computational Linguistics (2008)

Levels of Interaction (LoI): A Model for Scaffolding Learner Engagement in an Immersive Environment

David Panzoli¹, Adam Qureshi¹, Ian Dunwell¹, Panagiotis Petridis¹ Sara de Freitas¹,
and Genaro Rebolledo-Mendez²

¹ The Serious Games Institute, Coventry University Technology Park, Cheetah Road,
Coventry, CV1 2TL, West Midlands, UK

DPanzoli@cad.coventry.ac.uk, s.defreitas@coventry.ac.uk

² Facultad de Estadística e Informática, Universidad Veracruzana,
91500, Veracruz, Mexico
GRebolledo@uv.mx

Abstract. In this paper we present a theoretical framework describing an original method for the design of intelligent tutoring environments, building upon the notion of shared attention. This framework is defined as the *Levels of Interaction* (LoI) approach. It is applicable to applications where the learner/player is immersed in a 3D virtual environment, interacting and exchanging knowledge with an adaptive crowd of conversational agents.

Keywords: Virtual environments. Serious Games. Agent technology. Levels of Interaction. Learner engagement. Intelligent tutoring systems.

1 Introduction

To engage and immerse learners, next-generation learning environments must capitalise upon advanced visualisation and interaction paradigms, as user expectations of fidelity are high. This paper incorporates approaches built on a specific entertainment gaming genre: *role-playing games* (RPGs). This genre is particularly relevant to serious applications as it exhibits many educational parallels; long-term and affective involvement of learners, evolution and adaptation of content, and community-building and collaborative aspects.

2 Background

One of the most significant areas of potential for the deployment of an intelligent tutoring system (ITS) is the ability to integrate virtual characters with the environment and learning experience so that an *environment* is developed as opposed to a *system*, where all aspects of the learning experience are interlinked and driven by adaptive and intelligent agents. In the next section, we discuss the RPG genre of entertainment gaming, and its potential for serious game design.

3 RPGs as Large-Scale Interaction-Driven Environments

Freedom of exploration in a large-scale environments and interaction with non-player characters (NPCs) can increase the sense of presence and immersion of the player [1]. The *Rome Reborn* model is an example of a virtual recreation of a real-world environment with sufficient fidelity to serve as a learning environment. The case study considers how history may be taught to young students through interacting with realistic virtual Roman characters. These will exhibit historically plausible behaviours, whilst remaining interactive.

4 Levels of Interaction

The Levels of Interaction (LoI) model is a theoretical framework, where interactions between a human user/player and background characters are simplified to three levels. They are based on aspects of the joint attention (JA) theory [2], which emphasises *intentionality*. JA is defined as an active process involving agents performing intentional actions [3]. The LoI also conform to proxemics [4], in which physical settings constrain social interaction and also spatial proximity. Close proximity is also associated with improved collaboration [5]. The relationship of LoI to the *Rome Reborn* model is detailed in the next section.

4.1 Walking through an Immersive Crowd (LoI Level 1)

The first LoI is a *living background* underpinning the player's experience, enhancing their feeling of immersion within the experience.

4.2 Observing the Virtual Romans Trading and Craft Working (LoI Level 2)

Characters in the second LoI have a more realistic graphic representation and more complex behaviours, as enabled by the use of levels of detail (LoD) and levels of simulation (LoS) (see section 4.4) and a hierarchical cognitive controller.

4.3 Dialogue with Intelligent Roman Characters (LoI Level 3)

A conversational agent at the dialogue level equates to more traditional learning methods. JA has been linked to improved learning [6].

4.4 Boundaries and Transitions

The boundaries are defined with respect to the distance to the player, but are flexible in terms of context. Proximity between the player and the character(s) is controlled by the player in response to verbal and non-verbal communication. This allows increased interaction, characterised by the number of prerequisites (of JA) also increasing, up to the point of actual JA at the dialogue level.

5 Scaffolding Learner Engagement through Levels of Interaction

The LoI framework enables designers to consider the technological and pedagogic challenges to be met at each level in order to develop an effective learning experience within a serious game, based on a situative and experiential model. By consolidating interaction and visual fidelity (LoD) and simulation fidelity (LoS), the framework provides a model for developing interaction techniques.

6 Conclusions and Future Work

The paper proposes the LoI as a model for harmonising the techniques of RPG games with serious games techniques and embedded conversational agents to produce a design criteria for new ITSSs.

References

1. Jennett, C.I., Cox, A.L., Cairns, P., Dhoparee, S., Epps, A., Tijs, T., Walton, A.: Measuring and defining the experience of immersion in games. *International Journal of Human-Computer Studies* 66(9), 641–661 (2008)
2. Tomasello, M., Carpenter, M., Call, J., Behne, T., Moll, H.: Understanding and sharing intentions: The origins of cultural cognition. *Behavioral and Brain Sciences* 28, 675–691 (2005)
3. Kaplan, F., Hefner, V.V.: The challenges of joint attention. *Interaction Studies* 7(2), 135–169 (2006)
4. Hall, E.T.: *The Hidden Dimension: Man's Use of Space in Public and Private*, Garden City, N.Y. (1966)
5. Nova, N.: A Review of How Space Affords Socio-Cognitive Processes during Collaboration. *PsychNology* 3(2), 118–148 (2005)
6. Cleveland, A., Schug, M., Striano, T.: Joint Attention and Object Learning in 5- and 7-Month-Old Infants. *Infant and Child Development* 16, 295–306 (2007)

Tools for Acquiring Data about Student Work in Interactive Learning Environment T-Algebra

Rein Prank and Dmitri Lepp

University of Tartu, Institute of Computer Science, Liivi Str 2,
50409Tartu, Estonia
{rein.prank,dmitri.lepp}@ut.ee

Abstract. T-algebra is an interactive learning environment for elementary algebra. The main program of T-algebra enables visualizing information about a particular student in tables, indicating solution times, the numbers of errors (in 20 categories) and hint usage for each task. Additional software for teachers allows examination of solution results in group views.

Keywords: learning environments, algebra, acquiring data about learners.

1 Introduction

The results of the work in exercise classes can help the teacher to detect a need for additional explanations or even rearrangement of lessons. However, before this, the teacher needs to identify the difficulties (individual and common). If exercises are solved in a computerized environment, the computer can also be used to check solutions and prepare data for didactical decisions.

The current paper describes the teacher tools of T-algebra for acquiring data about student work. T-algebra is an interactive learning environment for elementary algebra: integer expressions; fractions; linear equations and systems; polynomials. Working with T-algebra, the student performs at each step three actions: selects the operation from the menu; marks the operand(s) in expression; enters the result of the operation. After the first two substeps the program checks the selected operation and suitability of marked operands. After input of the result T-algebra verifies its equivalence with operands and checks whether the form of the result corresponds to the operation performed. In case of error T-algebra issues an error message and requires correction of the mistake. A general description of T-algebra is presented in [1] and [2].

Different solution environments are able to extract and visualize different information. PepiGen [3] and ASSISTment [4] are two recent bigger examples in mathematics. They do not only show the results but also try to compute the didactical consequences automatically. However, this requires specific diagnostically oriented task design. Our approach is more lightweight. T-algebra tries to use its detailed solution dialog for creation of useful error messages and classification of errors. This information and some topic-independent statistics are presented to the teacher.

2 Data Tools of T-Algebra

The main program of T-algebra enables to visualize data about the work of one student. The teacher can see the solutions exactly in the form they were created by the student. Beyond that, the View menu offers five additional views: Statistics of solving, Error counters, Error list, Counters of help usage, List of help usage.

The statistics table includes the number and designation of problems solved by the student, the number of errors (total and separate for each task), the number of instances of help and Autosolve usage, and the number of solution steps and time characteristics (begin, end, duration) of each task. Error counters for each task are given for 20 separate categories of errors diagnosed by T-algebra at the three stages of solution step. Help usage is counted by the stages of the solution step where help was requested. Lists of errors and help usage enable to retrieve all error/help situations.

The figure consists of two side-by-side screenshots from the T-algebra software. The left screenshot shows a student's work on the problem $\frac{51}{9} \cdot \frac{1}{3} =$. The student has converted $\frac{51}{9}$ to $10 \frac{240}{27}$ and then to $10 \frac{240}{27}$. The right screenshot shows the 'List of errors' interface with error 13 selected, displaying the error message 'Some fractions not yet reduced'.

Fig. 1. Unfinished solution and List of errors with one selected error

Figure 1 presents two screenshots describing an incomplete solution. The student did the calculations without completing the reduction in the first factor. Then he tried to convert the result into mixed number, but did not choose the largest possible integer part. Before the last step he tried to submit the answer but received an error message. He started the corresponding operation Reduce but evidently was not able to find a common divisor and gave up.

The teacher also needs an overview of the work of the entire class or group of students. Our additional tool for group views creates the following tables:

1. **Table of solved/unsolved tasks.** The table consists of separate rows for each student and columns for each task. The cells contain symbol '+' if the student solved the task, symbol '-' if the student tried to solve the task but failed, and symbol '0' if the student did not try the task.

- Table of step counts.** In this table the rows of each student show the number of steps in each (finished or unfinished) solution and the total number of executed steps. The teacher can use the first two rows to display data from two reference files, e.g., from Autosolve solutions and teacher's own solutions.
- Table of solution times.** The rows of each student show the time used for each task (finished or unfinished) and the total time used for solutions.
- Table of error counts.** The rows of each student show the total number of errors and the number of errors made in each task.
- Tables of error types in tasks.** There are two overview tables for the errors made in tasks. The rows display the categories of errors in the first table and particular error messages in the second table.

Messages \ Tasks	Total	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Total	418	63	38	38	27	11	21	31	5	12	3	90	54	15	7		3	
Sign error	70	13	9	12	10	1	-	9	2	2	-	11	1	-	-	-	-	-
Calculation error	53	4	-	8	4	-	-	8	1	2	-	9	11	1	4	-	1	-
Incorrect member	23	5	-	-	3	-	-	2	1	1	1	6	4	-	-	-	-	-
Incorrect factor	21	3	-	1	5	4	-	3	-	2	-	2	1	-	-	-	-	-
Mark one variable	18	-	-	-	-	-	18	-	-	-	-	-	-	-	-	-	-	-
Mark one equation	16	1	2	1	-	-	-	1	-	1	-	5	5	-	-	-	-	-
Incorrect marking	14	-	1	3	3	-	-	-	-	1	1	1	2	-	1	-	1	-
Incorrect coefficient	13	-	-	-	-	-	-	-	-	-	-	6	6	1	-	-	-	-

Fig. 2. Upper part of the table of error messages

Figure 2 presents the upper part of the table of error messages of a class with 15 students of grade 7. The task file contained 17 complex tasks on linear equations with fractions. The average number of solved tasks was 8. After 30 minutes of work the teacher asked the students to move to the task 11 (with fractions before parentheses).

Acknowledgments. The authors are financed by grants SF0182712s06 and ETF7180 of Estonian Science Foundation.

References

- Issakova, M., Lepp, D., Prank, R.: T-algebra: Adding Input Stage to Rule-Based Interface for Expression Manipulation. *The International Journal for Technology in Mathematics Education* 13, 89–96 (2006)
- Prank, R., Issakova, M., Lepp, D., Tõnisson, E., Vaiksaar, V.: Integrating Rule-based and Input-based Approaches for Better Error Diagnosis in Expression Manipulation Tasks. In: Li, S., Wang, D., Zhang, J.-Z. (eds.) *Symbolic Computation and Education*, pp. 174–191. World Scientific, Singapore (2007)
- Delozanne, E., Grugeon, B., Previt, D., Jacoboni, P.: Supporting teachers when diagnosing their students in algebra. In: *AIED 2003 Online Supplementary Proceedings*, pp. 461–470 (2003), http://www.cs.usyd.edu.au/~aied/vol8/vol8_Delozanne.pdf
- Feng, M., Heffernan, N.T.: Towards Live Informing and Automatic Analyzing of Student Learning: Reporting in ASSISTment System. *Journal of Interactive Learning Research* 18, 207–230 (2007)

Mily's World: A Coordinate Geometry Learning Environment with Game-Like Properties

Dovan Rai, Joseph E. Beck, and Neil T. Heffernan

Computer Science, Worcester Polytechnic Institute
{dovan, josephbeck, nth}@wpi.edu

Abstract. *Mily's World* is a learning environment for coordinate geometry that has game-like properties, that is, elements of games that are engaging such as cover story, graphical representation, and animated feedback. This paper proposes that adding game-like properties to a computer tutor results in more student engagement and interest in the content material. We have taken a measured and minimalist approach to making the original environment more game-like by making a balance between stimulation and overload. We received mixed result in our experiment with sixty six students.

Keywords: educational game, authentic activities, motivation, visualization.

1 Introduction

Motivational benefit of computer games are appealing to education researchers but the games can add costs like taking instructional time away and increasing cognitive load on students. Hence, instead of completely integrating educational content into a game framework, we choose to incorporate into the tutor those features of games that are motivational but do not overly detract from learning. In order to balance between stimulation and overload, we are examining known characteristics of what makes a game enjoyable. We define *game-like properties* as elements of games that are responsible for their engaging nature such as graphics, fantasy, interactive feedback, rewards, etc. One goal is to determine the costs and benefits of these properties, both individually and when used in concert with other game-like properties.

The ASSISTment system is a web-based tutoring program for mathematics. To make ASSISTments more game-like, we created *Mily's World*. This environment has a series of 8th grade (approximately 13-year olds) coordinate geometry problems wrapped in a visual cover story. Students help characters in the cover story solve coordinate geometry problems to move the story forward. Similar to a classic ITS, students will receive tutorial help as they stumble on problems and misconceptions. Mily, a 9-year old girl, is the protagonist who has a puppy and some friends with whom she plays soccer. Students are engaged in many different math-related tasks. For example, they calculate Mily's height and the distance between her and her puppy based on the coordinates of their heads. As they proceed, students help Mily decide the name of the puppy and then help create a doghouse (see Fig 1). When students give the correct answer for slopes, the doghouse wall and roofs are built gradually and

then a new doghouse pops up. The puppy develops a bad habit of chewing socks; so Mily ties him to a post. Students have to help her find the coordinates of a position to place the socks where the puppy cannot reach them. Afterwards, Mily goes out with her friends to play soccer wearing the socks that the students have kept the puppy from chewing. Here, students have to calculate slopes and equations of the path of the ball as Mily and her friends play.

We have used a simple game-environment with familiar activities and minimalistic visual presentation. Our hope is that the game-like properties we added: authentic activities, visual presentation, storyline, immediate animated feedback not only helps to arouse emotional interest but also cognitive interest in the math content [4].

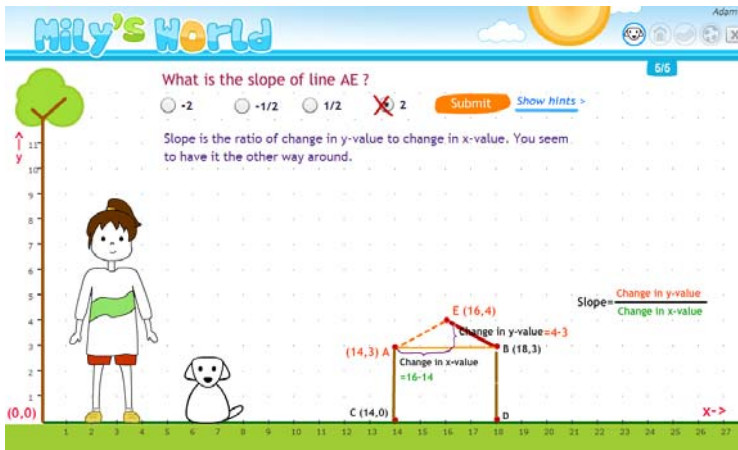


Fig. 1. Screenshot: *Mily's World*, doghouse problem

2 Experiment

Mily's World was assigned as homework to 8th grade students (12-14 year olds) in a school in the suburb of a small city in the Northeastern USA. Sixty six students started the exercise and 58 students completed it. Those students use Assistment in regular basis. There were 16 math questions and 12 survey questions and one open ended feedback question. Since we considered addition of game-like properties as much a cognitive intervention as an emotional one, we wanted to see if this is preferred by students who have preference for real-world problems and using pictures for learning math. We asked them these questions before using the tutor:

Do you find real-world examples helpful for solving math problem?

a) Yes, examples are helpful b) No, they make it more confusing

Do pictures help you learn math?

a) Yes, pictures help me b) I am not sure c) No, pictures don't help me

We later asked the students about their experience with *Mily's World*. On the question whether they like *Mily's World*, 20% said they liked it, another 20% said they did not

like it and 60% said they find it ok. When we made a regression analysis on liking Mily to their students' other survey responses (Table 1), we found that it is dependent on whether they liked story and graphics of Mily (emotional interest) and also on whether they find real world examples helpful or confusing (cognitive aspect). The open responses from students also revealed that some students found the mapping of math content to real-world scenario helpful while other found that confusing.

Table 1. Linear regression analysis, Dependent variable: like_Mily'sWorld (R Square= 0.35)

Variable	Beta (Standard coefficients)	Sig.
Real-world examples helpful/confusing	.31	.007
Pictures helpful/not helpful	.18	.13
Like story and graphics of <i>Mily's World</i>	.36	.003

We also asked students about their preference between *Mily's World* and Assistent. 52% preferred *Mily's World*, 13% preferred Assistent and 35% had no preference. This question was asked in the middle of the exercise instead of the end as we wanted to include the students who do not finish the exercise (we assume them to dislike it and therefore important for our study). So, their preference of *Mily's World* can be a factor of relative difficulty (questions ordered in increasing complexity in *Mily's World*) along with the novelty effect.

3 Conclusions and Future Work

Based on students' open responses, we found that the students generally liked the interactive approach of using pictures and feedback, but felt that the story was not age-appropriate for them. "The story was a bit childish, but it was clever how everything was incorporated. I found everything easy". This is our first iteration of finding the optimal point in the tutor-game space. Our belief that posing a younger character as a kid sister will be appealing was not correct for all students. Hence, as a next iteration, we are working on a new story *Monkey's Revenge* [4] using participatory design method.

References

1. O'Neil, H., Wainess, R., Baker, E.: Classification of learning outcomes: Evidence from the computer games literature. *The Curriculum Journal* 16(4), 455–474 (2005)
2. Gee, J.P.: *What Video Games Have to Teach Us About Learning and Literacy*. Palgrave/Macmillan, New York (2003)
3. McQuiggan, S., Rowe, J., Lee, S., Lester, J.: Story-Based Learning: The Impact of Narrative on Learning Experiences and Outcomes. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) *ITS 2008*. LNCS, vol. 5091, pp. 530–539. Springer, Heidelberg (2008)
4. Rai, D., Beck, J., Heffernan, N.: Coordinate Geometry Learning Environment with Game-like Properties. In: *Young researchers track of Tenth International Conference on Intelligent Tutoring Systems, Pittsburgh, USA* (in press 2010)

An Intelligent Tutoring System Supporting Metacognition and Sharing Learners' Experiences

Triomphe Ramandalahy, Philippe Vidal, and Julien Broisin

Institut de Recherche en Informatique de Toulouse, Université Paul Sabatier,
Route de Narbonne, 31062 Toulouse, France
{ramanda,vidal,broisin}@irit.fr

Abstract. Literature shows that Intelligent Tutoring Systems (ITS) are growing in acceptance and popularity because they increase performances of students, leverage cognitive development, but also significantly reduce time to acquire knowledge and competencies. We present an ITS offering the opportunity of evaluating various metacognitive indicators and able to share this information with other learning tools. Our online tutor is based on an existing ITS authoring tool that we extended to support metacognition and share learners' profiles and activities into a standardized, distributed and open tracking repository.

Keywords: intelligent tutoring, authoring tool, metacognitive experiences, model tracing tutoring, trace-based system.

1 Introduction

Intelligent Tutoring Systems (ITS) have proven their worth in multiple ways in education [1]. Namely, evaluations of these tutors showed significant achievement gain: students could achieve at least the same level of proficiency as conventional instruction in one third of time.

Leading researchers state that metacognition ensures effective learning. For example, if the feeling of difficulty is high and associated with negative affect, the learner quits the task. Even if several researchers have developed systems supporting various metacognitive attributes (self-explanation, gaming the system, self-monitoring or help-seeking), a few of them dig into some of dimensions of metacognitive experiences such as self-representation, self-image, self-evaluation or feeling of confidence.

To tackle the above issues, works presented here relate an ITS providing a generic approach to support any metacognitive attribute. This ITS is based on an existing authoring tool called Cognitive Tutoring Authoring Tools (CTAT) and elaborated by Carnegie Mellon university.

The remainder of the paper is organized as follows. The next section details how CTAT can be extended to integrate metacognitive attributes. Then, we describe the trace-based system able to gather and store data produced by the ITS. Finally we conclude before exposing some future works.

2 MetaCTAT: A Metacognition Aware ITS

Even if the well-known CTAT authoring tool brings numerous functionalities to easily build an ITS, it doesn't support metacognition: students are unable to reflect or monitor their knowledge from the tutoring activities they process.

Thus, we designed a set of classes presenting a high level of abstraction to support an undefined number of metacognitive indicators. This generic approach has been specialized to elaborate a tool dedicated to the curriculum Certificat Informatique et Internet (C2I). Our ITS named MetaCTAT and illustrated by Fig.1 consists of 20 questions endorsed by psychologists.

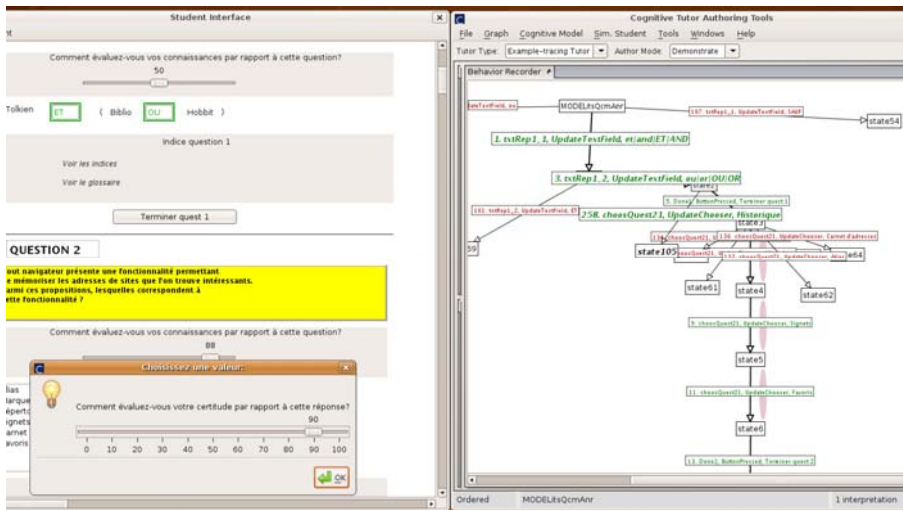


Fig. 1. MetaCTAT and the behavior recorder

This application focuses on metacognitive experiences because they are connected to the cognitive regulatory loop which concerns an ITS. MetaCTAT takes into account 3 indicators: Judgment of Learning, Feeling of Confidence and Feeling of Satisfaction.

Once logged in MetaCTAT, a student completes repeatedly one question from another according to the following process: first, he reads the question and immediately reports his prediction (JOL) through a Likert scale. Then, the student reads the propositions and finally answers to the question. As soon as the last event occurs, the learner has to indicate his level of confidence (FOC). At the end of the test, he reports his Feeling of Satisfaction (FOS). All of his activities are logged into a shared and distributed repository based on the standard Web-Based Enterprise Management (WBEM) specification.

3 The Trace-Based System

The WBEM specification is an ongoing initiative started by the Distributed Management Task Force (DMTF) to manage disparate networks, systems, and applications.

This widely-adopted standard stands on the extensible Common Information Model (CIM) to represent managed entities (systems, networks and applications) in a uniform point of view, and addresses the needs for a scalable solution by adopting a distributed architecture of the management components.

In order to take into account data resulting from MetaCTAT, we defined a set of classes and modeling users, resources and activities reported on [3].

We set up a distributed architecture conform to the WBEM standard comprising three parts: MetaCTAT, which represents the learning context; the tracking framework and an intermediate layer to offer an easy access to the tracking repository.

4 Conclusions and Future Works

The whole framework has just been achieved. Bugs have been identified and corrected, but the relative number of sample is not important enough to suggest a theory on effects of JOL, FOC and FOS regarding help-seeking. As we have a generic model, we will be able to personalize this ITS according to the learner profile. It is possible to have different ITSs from this sole framework.

We are working on integrating the rule engine JESS, on which is based metaCTAT onto a free and open-source ontology editor and knowledge-base framework Protégé through the plug-in JessTab. Our aim is to deliver to students a suitable resource satisfying a set of base of facts queried from a common knowledge base. Metadata resources are recorded into a native XML database management while his learning activities are stored within the WBEM repository.

References

1. Roll, I., Aleven, V., McLaren, B.M., Koedinger, K.: Designing for Metacognition - Applying Cognitive Tutor Principles to the Tutoring of Help Seeking. In: *Metacognition and Learning*, vol. 2, pp. 125–140. Springer, New York (2007)
2. Ramandalahy, T., Vidal, P., Broisin, J.: An abstract modeling of learning environments to ensure tracking of learners. In: *International Conference on Artificial Intelligence in Education*, pp. 650–652. IOS Press, The Netherlands (2009)

Are ILEs Ready for the Classroom? Bringing Teachers into the Feedback Loop

James Segedy, Brian Sulcer, and Gautam Biswas

Vanderbilt University, Nashville TN 37235, USA
{james.segedy,brian.sulcer,gautam.biswas}@vanderbilt.edu

Abstract. This paper proposes a new approach for incorporating intelligent learning environments (ILEs) into K-12 classrooms that tightly integrates interactions between the students, the classroom teacher, and the ILE. The ILE’s ability for continual, fine-grained monitoring and analysis of students’ learning activities supports the teacher’s ability to more effectively guide student learning.

Keywords: intelligent learning environments, classroom integration, behavior analysis.

1 Introduction

In intelligent learning environments (ILEs), students work to accomplish learning tasks while receiving guidance and scaffolding based on their performance and interactions with the computer-based system (e.g., [1]). While ILEs have shown impressive gains in student learning [2,3], integrating them into the classroom presents a new set of interesting and important challenges that revolve around the question: “How should the roles of the teacher and the ILE complement each other to help students learn in a more effective manner?” We believe that the relationship between the ILE and the classroom teacher is an important but often neglected aspect of designing and deploying ILEs. In this paper, we discuss our current vision for incorporating the teacher more fully into the ILE-enhanced classroom.

2 The Student, Teacher, ILE Relationship

In general, the student-teacher relationship, summarized in Figure 1a, is one where the teacher sets learning goals, and the student works to achieve those goals. Along the way, the teacher is likely to monitor student progress and provide necessary guidance and feedback. ILEs change the student-teacher relationship by providing the guidance and feedback, and also by automatically collecting data on students’ interactions with the system. The relationship between the student, teacher, and ILE is summarized in Figure 1b.

The student-teacher and student-teacher-ILE models have advantages and disadvantages. In the student-teacher model, the teacher can more effectively modify pedagogical interactions based on student characteristics and environmental factors outside of the learning interaction. Similarly, in the ILE-enhanced model, the ILE can “observe” all students simultaneously, examine and record how they progress through the learning task, and analyze these observations.

Given these separate and complementary strengths, we propose an extended model of the ILE-enhanced classroom, presented in Figure 2, where the ILE extracts interesting behavior patterns from log traces of student activities, and teachers use their pedagogical expertise to respond to them. Creating ILEs according to this model should allow for more successful integration of ILEs into classrooms.

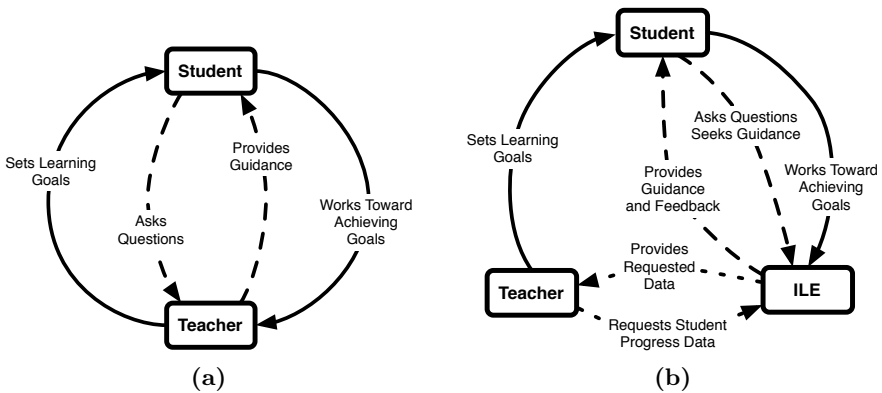


Fig. 1. Models of the student-teacher and student-teacher-ILE relationships

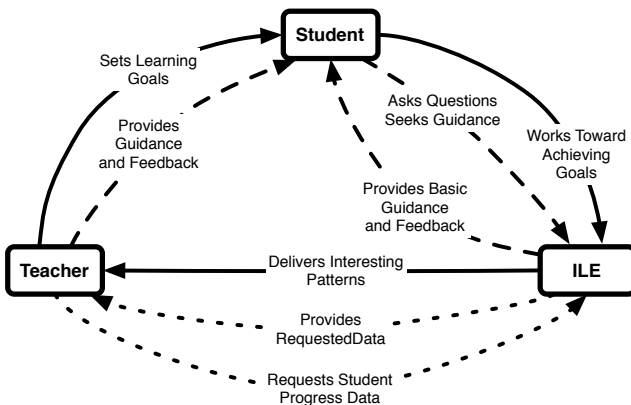


Fig. 2. Our extended model of the student-teacher-ILE relationship

3 Exploring the Extended Model

We explored the usefulness of the extended model by incorporating new tools into Betty's Brain [4], our ILE for teaching middle school students about scientific processes. In Betty's Brain, students are instructed to learn about these processes and then teach them to a virtual agent called Betty. They accomplish this task by reading textual resources and building a causal concept map (a set of entities and their relationships) that describes the information they've learned. When asked, Betty can answer questions and take quizzes. To answer questions, she employs qualitative methods to reason through "chains of relations" in the map. The students' goal in the ILE is to continue learning and teaching Betty until her concept map correctly answers all questions.

As a first step toward realizing the extended model, we provided teachers with information on the number of correct links about their students' concept maps daily. The teachers could track these numbers for individual students or the class. An innovative tool developed produced step-by-step movies of students' concept maps.

As we continue moving toward the extended model, we plan on automating the process by which the ILE delivers relevant information to the teacher. We also plan to expand the type of information reported. For instance, the ILE should also be able to report indicators of disengagement, such as goal-less button clicking. In order to support this near real-time information reporting, we plan on designing a "teacher dashboard" through which the teacher can monitor his/her classroom from a computer, see interesting behavior and performance trends, and "zoom in" on particular students to view more information about them.

4 Conclusions

In this paper, we proposed a model for ILE-enhanced classrooms that utilizes the complementary expertise of ILEs and teachers to optimize the learning experience for students. When implemented properly, it should lead to more successful incorporation of ILEs into classrooms.

References

1. Akhras, F., Self, J.: Beyond intelligent tutoring systems: situations, interactions, processes and affordances. *Instructional Science* 30(1), 1–30 (2002)
2. Koedinger, K., Corbett, A.: Cognitive tutors: Technology bringing learning sciences to the classroom. In: Sawyer, R.K. (ed.) *The Cambridge handbook of the learning sciences*, pp. 61–77. Cambridge University Press, Cambridge (2006)
3. VanLehn, K., Lynch, C., Schulze, K., Shapiro, J., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., Wintersgill, M.: The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence in Education* 15(3), 147–204 (2005)
4. Leelawong, K., Biswas, G.: Designing learning by teaching agents: The Betty's Brain system. *International Journal of Artificial Intelligence in Education* 18(3), 181–208 (2008)

Comparison of a Computer-Based to Hands-On Lesson in Experimental Design*

Stephanie Siler¹, Dana Mowery², Cressida Magaro¹,
Kevin Willows¹, and David Klahr¹

¹ Carnegie Mellon University, Department of Psychology,
5000 Forbes Avenue, 15213, Pittsburgh, PA, USA
{siler, klahr, cmagaro, KevinWillows} @cmu.edu

² Pittsburgh Science and Technology Academy,
107 Thackeray St., 15213, Pittsburgh, PA, USA
dmowery1@pghboe.net

Abstract. In this study, we compared our computer tutor (“TED” for Training in Experimental Design) to a teacher-guided control lesson also targeting experimental design but incorporating hands-on learning. Students in both groups showed significant gains in ability to design unconfounded experiments. TED instruction was significantly more efficient than the control lesson. When the teacher’s ratings of student ability were co-varied, students in the TED condition significantly out-performed control students on both immediate and delayed far transfer assessments taken three weeks after instruction. Students in both groups also reported a preference for physical over virtual materials.

Keywords: computer-based tutor; experimental design; middle-school students.

1 Introduction

An essential component to scientific literacy is an individual’s ability to design and evaluate experiments. Students may struggle to apply this fundamental skill in various contexts, falling into “cookbook” recipes of how to conduct scientific investigations, focusing more on materials than conceptual content. Some have claimed this problem is even worse with computerized instruction: “The conceptual or research goals of the laboratory get lost in the attention for equipment and there is no conceptual learning, nor learning of research or inquiry skills. Computers can glue students’ minds and hands even more strongly to the world of equipment...[1]”. To test such indictments of computer-supported instruction as well as to answer the practical question of how our computer tutor (“TED” for Training in Experimental Design) compares to what we considered to be a “good” lesson on experimental design, we compared TED to

* The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305H060034 to CMU and the National Science Foundation through the Pittsburgh Science of Learning Center, Grant SBE0354420. The opinions expressed are those of the authors and do not represent views of IES or the U.S. Department of Education.

ateacher-guided lesson incorporating more commonly used hands-on lab equipment. We were interested in comparing both learning/transfer and motivational outcomes.

2 Comparison of TED to Control Lesson

2.1 Participants, Design, and Procedure

Participants were 29 8th-graders in two classes at a local magnet school who did not show initial mastery of experimental design skills. Students in the 4th period class were assigned to the Control lesson and all students in the 5th period class to TED instruction. Both groups completed a 6-item computerized story pretest in which they designed and evaluated experiments in three contexts (drinks, cookies, and rockets). Then students participated in the 2-period lesson on consecutive days in their respective condition. Both groups then completed the computerized story posttest¹. They were reassessed three weeks later. Between the posttest and follow-up, all students completed a motivation survey assessing their enjoyment of different parts of the lessons and interest in science.

TED lesson. TED is a computerized tutor that administers instruction to help 4th-8th-graders learn the Control of Variables Strategy (CVS), the strategy of setting up an unconfounded experiment by changing only the variable of interest. The interface for TED includes virtual ramps whose four variables (e.g., slope) can be manipulated by the user. The instruction delivered in TED is based on the “explicit” CVS instruction developed by Klahr and colleagues [3], and involves evaluating experiments and receiving “explicit” feedback and explanations of the rationale for applying CVS.

Control lesson. The Cambridge Physics Outlet (CPO) Science Company offers their Foundations of Physical Science curriculum [2] to schools to support students’ understanding of science through hands-on investigations and basic concept lessons. Investigation 1.2 targets experimental design using ramps. In this lesson, students designed and ran experiments to test a predicted relationship between ramp steepness and car speed. Discrepant results led to a discussion, then conclusion this was due to experimental confounds. Thus, this lesson also addressed the rationale for using CVS.

2.2 Results

Pretest and posttest outcomes. There was no difference in the mean number of unconfounded experiments students designed or corrected on the pretest (Table 1). Though students in the TED condition tended to score higher on the posttest, this difference was not significant, $F(1, 26) = 1.95, p = .17$. On the delayed posttest, though TED students again tended to set up more unconfounded experiments, this difference was also not significant, $F(1, 25) = 1.68, p = .21$. Because standardized reading scores—typically correlated with CVS outcome measures—were not available, the teacher rated each student’s general ability on a 5-point scale. Co-varying these ratings, TED students scored significantly higher on immediate, $F(1, 25) = 7.48, p = .01$, and delayed posttest, $F(1, 24) = 5.34, p = .03$.

¹ Due to space limitations, we will not discuss the results of standardized posttests also taken other than to note there were no significant effects of condition.

Table 1. Mean story test score and efficiency (and standard deviations) by condition and time

Condition	Mean score (maximum of 6)			Efficiency	
	Pretest	Posttest	Delayed	Pre-to-Post	Pre-to-Delayed
Control	1.33 (1.18)	2.53 (2.23)	3.07(2.12)	0.08 (0.13)	0.12 (0.12)
TED	1.15 (1.21)	3.50 (2.10)	3.85 (2.44)	0.21 (0.19)	0.26 (0.20)

Instructional efficiency. On the final day of the intervention, to finish the lesson and posttests, the Control class ran 20 minutes late, causing the TED class to be shortened by 20 minutes. Thus, Control students had an extra 40 minutes to complete instruction and posttests. Students in the Control condition also took significantly longer on the posttest ($p < .01$). Because instructional times were significantly different, we compared instructional efficiency (i.e., pre-to-posttest gain divided by instructional time). As shown in Table 1, TED instruction was significantly more efficient with respect to both pre-to-immediate gains, $F(1, 27) = 4.43, p < .05$, and pre-to-delayed gains, $F(1, 26) = 4.67, p = .04$.

Survey results. On the motivational survey, the primary difference for both groups was a reported preference for working with real over simulated ramps.

3 Conclusions

Students in the TED condition had significantly higher immediate and delayed post-test scores when teacher ratings of student ability were factored out. Furthermore, TED instruction was significantly more efficient, as measured using either immediate or delayed story posttest gains. We believe these results are due to the more focused and repeated *conceptual* CVS instruction given in TED. Regarding students' reported preference for physical over virtual ramps, student enjoyment of virtual ramps may be increased by, for example, allowing them to run experiments, once correctly designed. However, adding such functionality risks diverting attention from instructional aspects more closely tied to meaningful learning.

References

1. van de Berg, E.: Improving Teaching in the Laboratory: Old Problems, New Perspectives. Paper presented at the Annual Conference Samahang Pisika ng Pilipinas, Cebu City (1997)
2. Hsu, T.: Foundations of Physical Science. CPO Science, Cambridge (2002)
3. Chen, Z., Klahr, D.: All Other Things being Equal: Children's Acquisition of the Control of Variables Strategy. *Child Development* 70(5), 1098–1120 (1999)

Toward the Development of an Intelligent Tutoring System for Distributed Team Training through Passive Sensing

Robert A. Sottilare

U.S. Army Simulation & Training Technology Center
12423 Research Parkway, Orlando, Florida, USA 32826
robert.sottilare@us.army.mil

Abstract. The development of intelligent tutoring systems (ITS) capable of supporting training experiences for geographically-distributed team members in shared virtual simulation environment presents considerable challenges. Even human tutors face challenges in developing team cohesion, coordinating roles and assessing contributions. Just as a human tutor might assess collective performance, a team ITS must be capable of passively assessing the trainees' readiness to learn and evaluating their progress toward team objectives. Passive sensing methods offer the opportunity for the ITS to understand the team's cognitive and emotional state without interfering with the learning process. It also helps determine their any interventions needed to optimize performance. This article reviews challenges and hypothesizes functions for computer-based distributed team tutors.

Keywords: team tutoring systems, distributed team training, passive sensing .

1 Introduction

This paper evaluates the challenges related to the management of distributed team training. Specifically, we address the functions and challenges of a distributed team ITS and recommend areas for future research. Functions include the ability to understand the student's readiness to learn (e.g. their emotional state), their individual performance, their interactions with other team members and their contributions to the collective performance of the team. A distributed team tutor must be capable of supporting the training needs of each team member, but also be able to communicate with other individual ITS regarding overall progress toward team goals, individual contributions toward team goals and the formulation of instructional strategies and interventions for the collective team and/or individual members. The tutor's ability to passively/unobtrusively sense behavioral and physiological cues and use those cues to classify individual student's states (e.g. emotions, beliefs, desires or intentions) is critical in determining interventions or instructional strategies (e.g. support, hints, pumps or directions) to provide a focused learning experience.

2 A Distributed Team Tutoring Model

A goal of our research is to develop a team tutor that will ultimately eliminate the need for human tutors for distributed training where the trainee’s interaction and direct access to human tutors is either limited or undesirable. Five design goals were identified for our team tutor model: passive, low-cost, accurate, portable and real-time. Our tutoring model evolved from an individual ITS model [1] and a procedural reasoning system [2] used for virtual characters to create the model below.

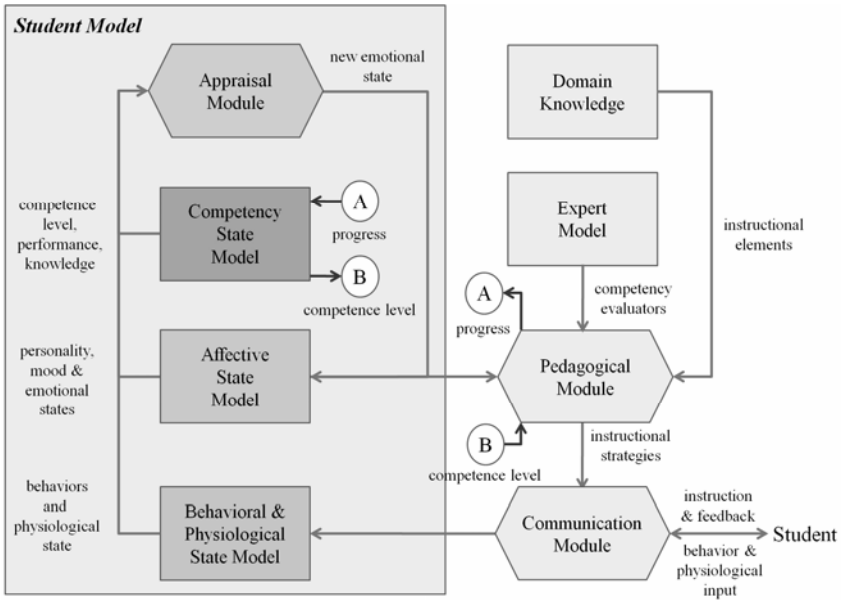


Fig. 1. Enhanced *student model* accounts for competence, affect, physiological and behavioral data to assess new emotional states and predict performance

For our team tutoring model, we identified the information to be exchanged between tutors, the frequency (periodic or event-driven) and triggers for exchanges:

- Team Performance State Models – event-driven changes occur as assigned tasks, progress toward team goals are registered within the individual ITS
- Team Competency State Models – event-driven changes are influenced by the performance of the team over time
- Team Affective State Models – periodic updates of individual affect and the distribution of these states to other ITS to determine team affect
- Team Trust State Models. periodic updates to bi-directional models of trust between team member; based on perceived competency, perceived integrity, perceived benevolence and knowledge of the other team members [3]

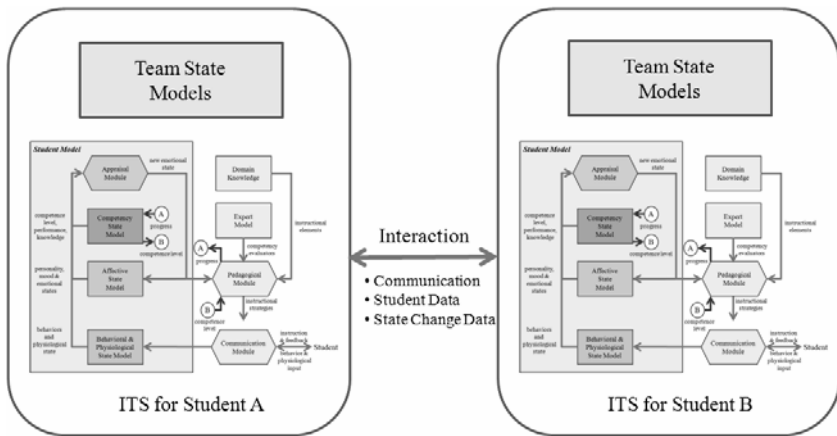


Fig. 2. Notional team tutoring model illustrates interactions between geographically-distributed autonomous ITS

In considering the design of computer-based team tutors, we identified five challenges in developing a team tutor: low cost, passive sensing of student behavioral and physiological data to assess readiness to learn; classification of team trust; selection of interventions based on individual and team states; real-time interaction of individual tutors to support real-time team training; and assessment of individual performance and contributions to team goals.

3 Conclusions and Recommendations for Future Research

Technologies exist to support a distributed team tutoring model, but additional research is needed to meet the challenges set forth in this article. Our goal was to find accurate, low-cost, portable, real-time, passive sensing methods that would be compatible with available technology (e.g. laptops or other mobile computing platforms). We reviewed methods of assessing affect using conversation patterns, student actions and knowledge, and facial changes. Additional research is needed to enhance conversational pattern recognition so it can be used easily on laptops/mobile devices with minimal setup. Finally, a more comprehensive student model developed in real-time through passive methods versus self-report methods is also needed.

References

1. Beck, J., Stern, M., Haugsjaa, E.: Applications of AI in Education. ACM Crossroads (1996)
2. Parunak, H., Bisson, R., Brueckner, S., Matthews, R., Sauter, J.: A Model of Emotions for Situated Agents. In: AAMAS 2006, Hakodate, Hokkaido, Japan, May 8-12 (2006)
3. Hung, Y.T., Dennis, A.R., Robert, L.: Trust in Virtual Teams: Towards an Integrative Model of Trust Formation. In: 37th Hawaii Intl. Conference on System Sciences (2004)

Open Educational Resource Assessments (OPERA)

Tamara Sumner¹, Kirsten Butcher², and Philipp Wetzler¹

¹ Institute of Cognitive Science, University of Colorado

² Department of Educational Psychology, University of Utah

{Sumner, Phillip.Wetzler}@colorado.edu,

Kirsten.Butcher@utah.edu

1 Background and Significance

“*Share, Remix, Reuse — Legally*”, the tagline for creative commons, cogently captures the ethos of peer production. Through the rapid growth of open educational resources (OER), peer production has begun to play a major role in how we teach and learn. OER are teaching and learning resources that reside in the public domain or have been released under licensing schemes that allow their free use or customization by others. They encompass a multiplicity of media types, including lesson plans, animations, videos, scientific data, etc. OER can be created by scientific institutions, by university faculty, by K-12 teachers, or by learners. Here, we focus on K-12 teachers engaging in peer-production for instructional purposes.

Central to OER vision is the assumption that peer-production processes lead to a cycle of continuous improvement. Namely, educators find useful OER on the Web (*reuse*), adapt and/or combine them to better meet their needs (*remix*), and then *share* their new resources with others. However, these skills require two types of knowledge that individuals often lack: content knowledge and metacognitive skills [1]. To reuse OER, teachers need to make difficult, complex, and time-consuming judgments to assess how well OER suit their instructional purposes. Judgments about the quality and appropriateness of OER are influenced by the information present in the resource, structural and presentational aspects of the resource, and the user’s content knowledge about the topic [2]. To adapt or remix OER, teachers need to think strategically about how to leverage the strengths and compensate for the weaknesses of particular resources (e.g., by clarifying learning objectives or adding reflective questions). To realize the transformative promise of OER, there is a critical need for reliable and scalable software tools that can help educators characterize the instructional quality of OER and use them more effectively in their own peer-produced resources.

2 Approach and Motivating Use Case

Our research is investigating how *open educational resource assessments* (OPERA) can increase the effectiveness of educators’ peer production practices. Open educational resource assessments use sophisticated algorithms combining machine learning and natural language processing to automatically analyze OER along dimensions important to teaching and learning, such as “organized around learning goals,” or “effective use of representations.” The goal is not to produce a single thumbs-up-or-down

decision on the overall quality of a resource, but to produce a rich profile characterizing the different strengths and weaknesses of a resource to aid human judgments. We do not hypothesize useful indicators; we have carefully studied the cognitive processes of skilled educators and designed software models to approximate their processes. We have developed: (a) a set of software models capable of assessing OER and approximating expert human judgments for a variety of quality indicators, and (b) a methodology for identifying and operationalizing potentially useful indicators through empirical studies of human decision-making processes.

To illustrate the potential impact of OPERA, consider a teacher creating a webquest to help students understand how changes in the water level of the Colorado River impact predators and prey. The teacher wants to include background readings on environmental adaptation and resources that enable students to use scientific models. Imagine that the search engine and the editing tools are OPERA-enhanced: the OPERA-enhanced search results note that one resource is from “a highly reputable sponsor” and another is “making effective use of representations.” The “reputable sponsor” resource provides an excellent background reading and a powerful instructional video; the “effective representations” will form the backbone of her webquest, since it contains a series of simulations that enable students to explore what-if scenarios for a fictional watershed. As she saves her webquest, the OPERA-enhanced editor notes that the webquest not “organized around learning goals” and lacks clear “instructions” (Figure 1). She revises her webquest to include questions to guide student thinking and instructions on how to use the simulations and animations to explore questions about the relationship between water level and predators/prey.

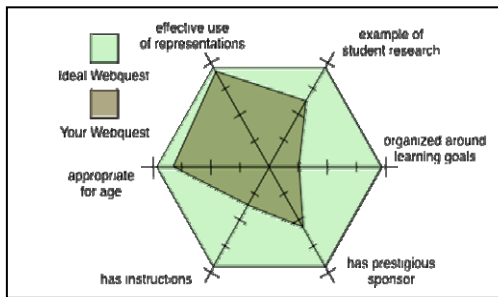


Fig. 1. Mock-up illustrating possible quality profile visualization

3 Results

As a first step towards realizing the OPERA vision, we have demonstrated the feasibility of automatic quality assessments for a single domain: high school Earth science. We conducted a series of experiments to identify an initial set of quality indicators; i.e., a set of criteria useful for assessing the instructional quality of OER for K-12 settings. Indicators were identified and characterized using a set of rich qualitative and quantitative data on expert evaluative processes involving re-analyses of data sets from other projects as well as a lab study with science education experts. We then developed computational quality assessment models, one model for each indicator.

Initial models were created and evaluated following a standard supervised machine learning approach: models were trained using a carefully prepared corpus of annotated examples, then evaluated on previously unseen corpus examples. Details of the experimental and computational methodologies are published elsewhere [3]. Our test bed was 1000, high-school level educational resources drawn from the Digital Library for Earth System Education (www.DLESE.org).

Our machine learning models analyze a resource and determine whether or not a quality indicator is present. Every decision looks at a complete resource – containing multiple web pages, rich media, and PDF files – as a unit. Since machine learning models operate on numerical vectors, we build a vector representation of each resource by extracting a number of numerical and yes/no features. Some features are taken straight from the text (e.g. groups of words in the resource), and some make use of non-textual elements (e.g., HTML structure); other features include the host domain (URL), or the sites linked to the resource. The system analyses these vectors, generated from the training corpus, to learn a statistical model for each indicator. We use a support vector machine approach to machine learning. Training parameters are chosen using cross validation: we repeatedly build a model from one part of our training data and evaluate it on the rest, each time refining the parameters of the algorithm. We then compare the results to a simple baseline: always assume the most common case (e.g., the *has instructions* indicator is present in 39% of resources; if we always assume that a resource has *no instructions*, we'd be correct in 61% of cases).

Table 1. Model Evaluation Performance Results

Indicator	Baseline	Models
Has instructions	61%	78%
Has prestigious sponsor	70%	81%
Indicates age range	79%	87%
Identifies learning goal	72%	81%
Organized for goals	75%	83%

Table 1 shows the performance of the machine learning models relative to this baseline. Good improvements over the baseline were achieved on *has instructions* and *has prestigious sponsor*, and moderate improvements on the *indicates age range*, *organized for goals*, and *identifies learning goals* indicators. These results are very encouraging in that, even using basic features, we classified many indicators well.

References

1. Lin, L., Zabrocky, K.: Calibration of comprehension: Research and implications for education and instruction. *Contemporary Educational Psychology* 23, 345–391 (1998)
2. Sumner, T., Khoo, M., Recker, M., Marlino, M.: Understanding Educator Perceptions of “Quality” in Digital Libraries. In: 3rd ACM/IEEE Joint Conference on Digital Libraries, pp. 269–279. ACM Press, New York (2003)
3. Bethard, S., Wetzler, P., Butcher, K., Martin, J., Sumner, T.: Automatically Characterizing Resource Quality for Educational Digital Libraries. In: 9th ACM/IEEE Joint Conference on Digital Libraries, pp. 221–230. ACM Press, New York (2009)

Annie: A Tutor That Works in Digital Games

James M. Thomas and R. Michael Young

Digital Games Research Center
Department of Computer Science
North Carolina State University, Raleigh, NC USA
jmtthoma5@ncsu.edu, young@csc.ncsu.edu

Abstract. This paper describes Annie, a domain-independent intelligent tutor that can be “plugged-in” to digital games to guide learners using the core mechanics of the game.

Keywords: Intelligent tutor, digital game, exploratory learning, plug-in.

1 Introduction

We are developing a system that embeds intelligent tutors into exploratory game-based learning environments [1,2]. As de Jong noted [3], it is difficult to balance guidance with student exploration and “in such a way that learning is supported effectively, but the inquiry process is not reduced to following cookbook instructions.” Critics charge that combining exploratory learning with games often “sucks the fun out” or “sucks out the learning” [4]. Thoughtful integration of learning and gameplay can result in successful educational games, but often couples game and tutorial logic, limiting scalability and reusability. We have built a system, named “Annie”, that facilitates the construction of a new class of engaging educational experiences through a tight integration of game mechanics and learning, driven by general, reusable tutorial plug-ins.

2 Integrating Game Mechanics with Learning

Annie builds an integrated description of learning goals and game play by extending a well-understood computational model of action, cause, and effect. The same model is used both to describe the current and goal states of student knowledge as well as the current and goal states of the game world. The model extends the language of automated planning, specifically a STRIPS-style description of actions, effects, and objects in a planning domain.

Annie constructs an initial tutorial plan consisting of a plausible partially-ordered sequence of student and system-initiated actions that is designed to bring about a specific goal state for the world, (e.g. the player/student correctly assembles a particular molecule in an educational chemistry game). In addition to achieving a goal state for objects in the world, the tutorial plan ensures a particular state of task knowledge acquisition in the student model (e.g., the student knows that the first step in the synthesis of a particular molecule involves hydrogen atoms bonding with carbon). The tutorial plan

marks out an optimal game play path for the user prior to the start of the session, but it must be continually revised based on student actions.

Each time an action is taken in the world, either by the student or the system, Annie updates its model and considers revising its plans. Following the action, Annie consults an extensive library of general **diagnostic** templates to update its student model. These templates encode domain-independent plan reasoning diagnostics such as cases where a student seems to be ignorant of a precondition of a particular operator. Annie uses the updated student model in consulting a second extensive and domain-independent library containing **remediation** templates that can be used to generate narrative scaffolding. Although these templates target a fairly primitive level of task elements, their hierarchical compositions mirror methods in which learning principles are currently embedded in games. Space limitations prohibit a more complete description of Annie's knowledge representation and design, but one can be found in [5]. This paper describes in more detail how Annie will guide students through a representative learning game.

2.1 Recapitulating Game-Based Learning through Planning

Competitive market pressures have instilled commercial digital games with impressive tutorial capabilities. Games that effectively scaffold player development while delivering engaging content outsell those that require the player to first graduate from obvious “tutorial modes”, or worse yet, read the friendly manual distributed with the game. Gee describes a rich set of learning principles employed by commercial games [6] and Quintana organized a framework of many of the scaffolding techniques used in exploratory learning systems [7]. Gee and Quintana have provide what are essentially catalogues of loosely related concepts culled from a wide array of existing systems. To dynamically generate appropriate game-based guidance based on individual student behavior, Annie requires a more prescriptive model.

Annie's model consists of plan-based building blocks that are assembled into at run-time through a set of templates. These templates emulate many of the concepts described by Gee, Presnky and Quintana. Annie effectively transforms these descriptive models of game-based learning into a system that synthetically generates learning content. Synthetic generation is advantageous because it can happen at run-time, dynamically adapting to the behaviors exhibited by the student.

For example, one of the thirty-six principles articulated in Gee's survey states that “The learner is given explicit information both on-demand and just-in-time, when the learner needs it or just at the point where the information can best be understood and used in practice.” Annie embodies this principle by continuously reviewing the set of possibly successful plans and noting how soon the student needs to know each portion of the tutorial content to make progress through a tutorial plan. Because this process is automatic, it is subject to quantifiable heuristics. For some students or groups of students Annie may want to vary how far in advance help can be provided based on the estimated attention span or projected memory persistence of those students. Analysis of these metrics over many students allows Annie to be easily fine-tuned to improve its performance.

2.2 Example Application: FixIt

Because Annie specializes in task-based learning, it is optimally suited for domains where the key learning challenges involve processes that involve the composition of a sequence of causally-related actions. For our initial evaluation of Annie's effectiveness we have developed a game called "FixIt" that teaches the conceptual domain of computer operating system security. The game features four progressively more difficult missions that require the student to identify and remove increasingly complex forms of computer malware.

The player's first mission is to identify a system program that is consuming too many resources. The player is asked to use the "Information" tool to determine the wayward process, and is then guided to correct the problem with a "Nice" tool that resets the priority of the process. In the third mission, a renegade child process is automatically re-spawned by a trojan hidden inside a trusted system program. If the parent is found and killed, it triggers an automatic system restart. But the attack has also left a back door, so if the user does not close that vulnerability, the parent process will be reinfected and re-spawn the child process. The back door is a hidden file that can only be deleted when not protected by a FILE_IN_USE flag held by the renegade child process. The third mission forces the user to re-evaluate the lessons of the second mission where killing a bad process cured an earlier malware infection. By guiding the student to through similar tasks through in a series of "missions", where the same task is operating on different objects in the world, we are able to better leverage our task-based student model.

3 Conclusion

Annie leverages a plan-based representation of the game world to generate guidance through the core mechanics of the game. Our next task is an empirical evaluation of Annie's teaching effectiveness.

References

1. Bunt, A., Conati, C.: Probabilistic Student Modelling to Improve Exploratory Behaviour. *User Modeling and User-Adapted Interaction* 13(3), 269–309 (2003)
2. Mott, B., Lester, J.: U-director: a decision-theoretic narrative planning architecture for storytelling environments. In: *AAMAS*, pp. 977–984 (2006)
3. de Jong, T.: Technological advances in inquiry learning. *Science* 312(5773), 532–533 (2006)
4. Prensky, M., Bowers, J.C.: Serious games debate. In: *Serious Games Summit* (October 2005)
5. Thomas, J., Young, R.: Using Task-Based Modeling to Generate Scaffolding in Narrative-Guided Exploratory Learning Environments. In: *Proceedings of the 14th International Conference on Artificial Intelligence in Education* (2009)
6. Gee, J.: What video games have to teach us about learning and literacy. *Computers in Entertainment (CIE)* 1(1), 20 (2003)
7. Quintana, C., Reiser, B., Davis, E., Krajcik, J., Fretz, E., Duncan, R., Kyza, E., Edelson, D., Soloway, E.: A Scaffolding Design Framework for Software to Support Science Inquiry. *The Journal of the Learning Sciences* 13(3), 337–386 (2004)

Learning from Erroneous Examples

Dimitra Tsovaltzi, Bruce M. McLaren, Erica Melis, Ann-Kristin Meyer,
Michael Dietrich, and George Gogvadze

DFKI GmbH, German Centre for Artificial Intelligence, Stuhlsatzenhausweg 3
(Building D3 2) D-66123 Saarbrücken Germany
Dimitra.Tsovaltzi@dfki.de

Abstract. We present students with common errors of others in the context of an intelligent tutoring system (ITS). We conducted two studies with students of different curriculum levels to measure the effects of learning through such erroneous examples. We report that erroneous examples with additional support can assist lower curriculum level students develop better meta-cognitive skills.

Keywords: Erroneous examples, fractions misconceptions, adaptive learning.

1 Introduction

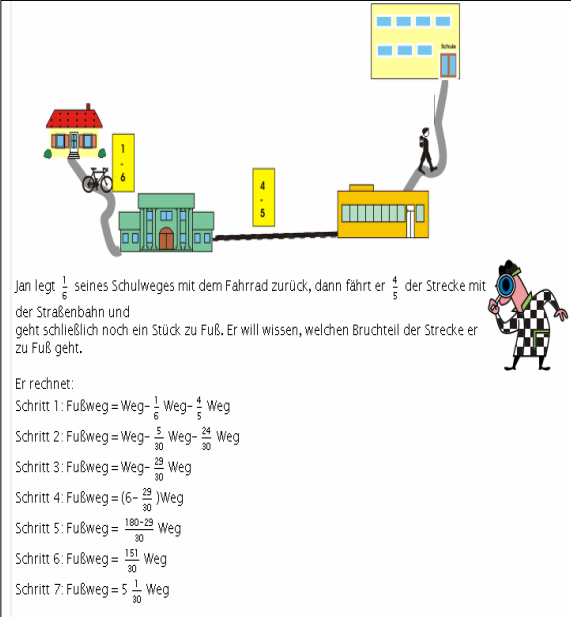
Erroneous examples (ErrEx) that is, worked solutions that include one or more errors that the student is asked to detect, explain, and/or correct have been rarely investigated as a teaching strategy in mathematics. On the contrary, correct worked examples have been the focus of research on mathematics and science problem solving, e.g. (4). Yet, the scarce work on ErrEx in mathematics has provided some evidence that studying errors can promote student learning (1). Grosse and Renkl's (2) empirical studies showed some learning benefits of studying incorrect examples on transfer, but only for high-competent learners.

We investigate ErrEx with and without help and are interested in how students react when they receive help in the context of an ITS. This novel design relies on the intelligent technology developed in *ActiveMath* (3), a web-based learning environment for mathematics. The help supports students who are not accustomed to working with ErrEx and may not have the required skills to analyze, and reflect upon such examples. We aim to find out if and when erroneous examples can foster students problem-solving, concept understanding, and transfer abilities, as well as the meta-cognitive competencies of error detection/ awareness and error correction.

2 Lab Studies with Erroneous Examples in ActiveMath

We conducted lab studies with German 6th, and 7th and 8th-graders. There were three conditions. No-ErrEx (NOEE) was the control and included standard fraction exercises of the form $3/4+5/7$, but no ErrEx in the intervention. ErrEx-Without-Help (EEWOH) included standard exercises, and erroneous examples but with no additional help, whereas ErrEx-With-Help (EEWH) received additional help on the ErrEx.

The design included pretest, familiarization, intervention, and posttest. In the intervention, all groups solved six sequences of three items: two standard exercises and either a third exercise or an ErrEx on fractions. Word problems were also included (Figure 1). 7th and 8th-graders also solved *modeling exercises*, which required them to use fraction operators to represent a word problem. The online ErrEx that were used included two phases: error detection and error correction. Figure 1 displays the first phase. The task is: “Jan rides his bike for 1/6 of his way to school, then drives with the tram 4/5 of the way and finally goes the rest of the way on foot. He



Jan legt $\frac{1}{6}$ seines Schulweges mit dem Fahrrad zurück, dann fährt er $\frac{4}{5}$ der Strecke mit der Straßenbahn und geht schließlich noch ein Stück zu Fuß. Er will wissen, welchen Bruchteil der Strecke er zu Fuß geht.

Er rechnet:

Schritt 1: Fußweg = Weg - $\frac{1}{6}$ Weg - $\frac{4}{5}$ Weg
 Schritt 2: Fußweg = Weg - $\frac{5}{30}$ Weg - $\frac{24}{30}$ Weg
 Schritt 3: Fußweg = Weg - $\frac{29}{30}$ Weg
 Schritt 4: Fußweg = $(6 - \frac{29}{30})$ Weg
 Schritt 5: Fußweg = $\frac{180-29}{30}$ Weg
 Schritt 6: Fußweg = $\frac{151}{30}$ Weg
 Schritt 7: Fußweg = $5 \frac{1}{30}$ Weg

Fig. 1. Phase 1: Error detection

wants to know what fraction of the way he goes on foot.” In this phase, students select the erroneous step. There are three types of unsolicited feedback: (i) Minimal feedback is both flag and verbal. (ii) Error detection and awareness (EAD) feedback, intends to foster meta-cognitive skills (e.g., “The result, way on foot = 5 1/30 cannot be correct. The trip with the bus is already 4/5, so the way on foot must be less than 1/5 of the whole way”.) (iii) Multiple choice questions (MCQs) attempt to help students understand the underlying principles of the task through questions like “Why is the 4th step wrong?” EAD and MCQs were only available to EEW condition. In the second phase, students are prompted to correct the error. All conditions received minimal feedback and the whole correct answer after one wrong attempt. The posttest consisted of similar exercises as the intervention, a transfer exercise (four-fraction addition), and ErrEx with conceptual questions to test students’ error detection skills as well as underlying principles of fractions (e.g. “What mistake did Oliver make?”).

6th-GradeResults. Twenty-three (23) paid volunteers participated in the lab studies, (EEWH=8, EEWOH=7, NOEE=8). The mean of their term-grade in math was 2.04 (best=1, fail=6), so the participants were high prior knowledge students. As a result, there was a ceiling effect in the standard exercises (post-pre-diff, $M = 5.04$). EEWH had the highest score in all ErrEx scores (cf. Table) and the same score with NOEE in the transfer exercise. Additionally, the total score on ErrEx that includes finding and correcting the error and conceptual questions, is significantly higher in EEWH than in NOEE ($t(14)=2.227$; $p = 0.043$). With the term-grade as covariate, there are also significant differences in all ErrEx scores between EEWH and EEWOH.

Condition	Descriptive Statistics 6 th -Grade			Descriptive Statistics 7 th -8 th -Grade		
	EEWH	EEWOH	NOEE	EEWH	EEWOH	NOEE
<i>Type of Score</i>	<i>mean(sd)%</i>	<i>mean(sd)%</i>	<i>mean(sd)%</i>	<i>mean(sd)%</i>	<i>mean(sd)%</i>	<i>mean(sd)%</i>
EEfind	91.7(15.4)*	71.4(35.6)^	62.5(27.8)	68.7(34.7)	75.0(13.4)^	90.6(12.9)*
EEcorrect	80.2(12.5)*	75.0(21.0)^	68.7(25.9)	60.9(30.2)^	57.8(20.0)	71.9(24.8)*
EE-ConQuest	64.6(25.5)*	60.2(11.0)^	8.3(21.2)	55.2(46.5)	62.5(12.6)*	61.5(19.4)^
EE-total	73.1(18.3)*	5.5(27.8)	51.2(20.8)^	59.1(39.3)	64.1(1.9)^	69.4(17.3)*
Transfer	75.0(46.2)*	71.4(48.8)^	75.0(46.3)*	45.2(45.8)^	38.0(36.0)	67.3(28.5)*
Post-pre-diff	-2.1(33.6)	1.2(21.7)^	2.1(23.9)*	2.4(24.4)^	1.1(21.6)	7.0(18.0)*

Note: *=best , ^=middle learning gains (no marking for learning loss)

7th-8th-Grades Results. Twenty-four (24) students participated, eight in each condition. Their mean term-grade in math was 2.88. Surprisingly NOEE did better in almost all scores, although the differences are small and not significant (cf. Table). An interesting result is that the term-grade is a significant covariate on conceptual questions ($p=0.047$) but not on problem-solving. A possible interpretation is that the math level influences conceptual understanding more than problem-solving of fractions at this class level. The study also revealed that a significant number of students could find the error but not correct it ($t(23)=4.89, p=0.000<0.001$). This may mean that although students have declarative knowledge that helps them identify rule violations, they still have knowledge gaps that are exposed when they correct errors.

Conclusion. Our results support the hypothesis that meta-cognitive skills are fostered by the use of ErrEx and additional help for high-competent 6th-graders, but not for 7th/8th-graders, where prior knowledge seems to play a crucial role. They also indicate a possible dissociation, described by Ohlsson (5), between declarative knowledge (finding rule-related errors) and practical knowledge (correcting errors).

References

1. Borasi, R.: Capitalizing on errors as “springboards for inquiry”: A teaching experiment. *Journal for Research in Mathematics Education* 25(2), 166–208 (1994)
2. Grosse, C.S., Renkl, A.: Finding and fixing errors in worked examples: Can this foster learning outcomes? *Learning and Instruction* 17, 612–634 (2007)
3. Melis, E., Goguadse, G., Homik, M., Libbrecht, P., Ullrich, C., Winterstein, S.: Semantic-aware components and services in ActiveMath. *British Journal of Educational Technology. Special Issue: Semantic Web for E-learning* 37(3), 405–423 (2006)
4. Paas, F.G., Renkl, A., Sweller, J.: Cognitive load theory and instructional design: Recent developments. *Educational Psychologist* 38, 1–4 (2003)
5. Ohlsson, S.: Learning from Performance Errors. *Psychological Review* 103(2), 241–262 (1996)

Feasibility of a Socially Intelligent Tutor

Jozef Tvarožek and Mária Bieliková

Faculty of Informatics and Information Technologies,
Slovak University of Technology,
Ilkovičova 3, 842 16 Bratislava, Slovakia
{jtvarozek,bielik}@fiit.stuba.sk

Abstract. We present a feasibility study of an intelligent tutoring system Peoplia in which a socially intelligent tutoring agent uses common instructional methods that are augmented by social features to help students learn. Peoplia features pseudo-tutor assessments, free-text answering, personalized question generation, and adaptive question selection. It allows students to work both individually and collaboratively while the tutoring friend monitors their social behavior and motivates them by socially relevant interventions.

Keywords: social intelligence, intelligent tutoring friend, motivation.

1 Introduction

Sustained student motivation is important for effective learning. However, providing motivational feedback is often at odds with cognitive scaffolding, and research is still seeking the right balance between the two [1]. In our research we attempt to improve students' motivation in a traditional tutoring environment using a socially intelligent agent, the tutoring friend, which addresses aspects beyond that of an individual student. A tutoring friend is an artificial learning companion that manages relationships with students, monitors their social behavior, and can provide them with interventions appropriate for the social context in which they learn [2]. All in all, research in politeness and its role in effective tutorial dialogue, motivating students and learning [3,4] suggests that intelligent tutors can maintain the appearance of being socially intelligent by carefully selecting the appropriate words at the appropriate time, not requiring the presumably unavoidable labor intensive language processing methods. In our approach we attempt to follow these observations.

2 The Peoplia System and Feasibility Study

Peoplia is an interactive web-based environment that helps students to learn using various types of learning opportunities that are facilitated by a socially intelligent agent, the tutoring friend. It is a rework of our previous idea of a computerized assessment system that supports traditional classroom assessment [5] with an emphasis on problem solving, which is analogous to the Assistments system [6] adding a robust task generator that discourages cheating (during assessment) and surface approaches to learning (during exercise).

Problem solving. The central learning opportunity in Peoplia is problem solving. Students work on pseudo-tutor problems by attempting to solve a starting question (subtask) of the problem description, providing answers either textually or by interacting with an interactive component (e.g. radio button). In collaborative mode, individual problem solving is augmented with: (1) instant messaging, (2) voting for the most agreeable answer in the team, and (3) a multi-user interface. Student answers are graded (matched to the predefined set specified in the problem description) by the two-stage grading process with a human in the loop [5]. Course notes are enhanced to social study mode through the use of text highlights, sticky notes, and a dialogue facility for asking the tutoring agent or currently available fellow students for help.

Tutoring friend. The tutoring friend primarily manages the off-task social dialogue facility and does not participate directly in other learning activities which consequently appear for students to operate autonomously. The social support within the individual learning activities in Peoplia is strictly on structural level e.g. in the form of social recommendation of annotations, voting for best team answer, etc. as mentioned earlier in the descriptions of the learning activities. The tutoring friend influences the transitions between activities by a set of rules that can recommend a good course of action for the student at any given moment. The appearance of social intelligence in the tutoring friend is based on the data collected during the off-task social dialogues. The tutor's dialogue capability is scripted using an ignorant approach [7] enhanced by a dimensional model of relationship with the student [8].

Experiments. We conducted two experiments in middle school mathematics; in the first study, we were interested in how much would students revealed about themselves to an artificial friend that they never met before and that communicates via a text console, all under the assumption that they expect (after entering a computer lab for a math class) some form of computerized exercises or assessment. The tutoring friend was scripted to “go easy” on the student, politely ask how she feels, and inquire about her hobbies under the guise of providing her with personalized exercises.

16 students (6 females, 10 males) were transferred to the computer lab, and were instructed to work on math exercises in Peoplia. We were interested in the word count of students' comments in the welcome dialogue and the number of features (hobbies; such as *to draw*, *sleep*, *watch TV*, *go out*, and *dog*) they disclosed to the artificial tutoring friend. The mean word count per student was 11.625 (st.dev 8.69) and the mean feature count per student was 1.56 (st.dev 1.75), thus on average each student revealed at least one of her hobbies. However, 44% of the students (1 female, 6 males) ignored the welcome dialogue by not using more than 3 words; students that actually cooperated with the tutoring friend used 16.89 (st.dev 4.91) words, and revealed 2.78 (st.dev 1.39) features on average.

In the second study, 32 students (14 females, 18 males) took part in a 3 day long experiment in which first a pre-test was administered, then students took 2 instructional units (45 minutes) – one per day – followed by a post-test, with no instructional unit on the day of the post-test administration. The control group (8 females, 8 males) attended 2 units of traditional classroom instruction vs. 2 units of problem solving in Peoplia in the experimental group. We were interested how much students learned even though they “wasted” 10% of the available time for off-task interactions with an artificial agent, how this compares to traditional classroom instruction, and what differences would the interaction with social agent make. The unpaired *t*-test confirmed

differences in the pre-test scores between the groups, and thus we cannot directly compare gains achieved by the experimental group vs. the control group. By analyzing only the results of the experimental group we get the 95% confidence interval for learning gains in the range 1.2% to 19.5%, thus students in the experimental group did show nonzero learning, which was coincidentally at least as high (1.2%) as in the control group (although not comparable).

We plan to repeat the study on a larger scale. In the questionnaire, students' feelings about how helpful the system was were modest, students tend to feel more positive about using the system again, and liking the system in general. When we filter out the 7 students who did not engage with the social agent we see fewer tasks attempted while solving more correctly, and also the questionnaire answers shift to the positive end. Students that did engage with the tutoring friend liked the system and the tutor more, and were also more successful in solving problems within the tutoring environment.

Acknowledgments. This work was supported by the Scientific Grant Agency of SR, grant No. VG1/0508/09, the Cultural and Educational Grant Agency of SR, grant No. 028-025STU-4/2010, and it is a partial result of the Research & Development Operational Program for the project Support of Center of Excellence for Smart Technologies, Systems and Services II, ITMS 25240120029, co-funded by ERDF.

References

1. Boyer, K.E., Phillips, R., Wallis, M., Vouk, M., Lester, J.C.: Balancing the cognitive and motivational scaffolding in tutorial dialogue. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 239–249. Springer, Heidelberg (2008)
2. Tvarožek, J., Bieliková, M.: The Friend: Socially-Intelligent Tutoring and Collaboration. In: AIED 2009, pp. 763–764. IOS Press, Brighton (2009)
3. Mayer, R.E., Johnson, W.L., Shaw, E., Sandhu, S.: Constructing computer-based tutors that are socially sensitive: Politeness in educational software. *International Journal of Human-Computer Studies* 64(1), 36–42 (2006)
4. McLaren, B.M., Lim, S., Yaron, D., Koedinger, K.R.: Can a Polite Intelligent Tutoring System Lead to Improved Learning Outside of the Lab? In: AIED 2007, pp. 331–338 (2007)
5. Tvarožek, J., Kravčík, M., Bieliková, M.: Towards Computerized Adaptive Assessment Based on Structured Tasks. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 224–234. Springer, Heidelberg (2008)
6. Feng, M., Heffernan, N.T., Koedinger, K.R.: Addressing the Testing Challenge with a Web-Based E-Assessment System that Tutors as it Assesses. In: WWW 2006, pp. 307–316. ACM Press, New York (2006)
7. McCalla, G.I., Murtagh, K.: G.E.N.I.U.S.: An experiment in ignorance-based automated program advising. *AISB Newsletter* 75, 13–20 (1991)
8. Svennevig, J.: *Getting Acquainted in Conversation*. John Benjamins, Philadelphia (1999)

Agent Prompts: Scaffolding Students for Productive Reflection in an Intelligent Learning Environment

Longkai Wu and Chee-Kit Looi

National Institute of Education, Nanyang Technological University, Singapore
longkai.wu@gmail.com, cheekit.looi@nie.edu.sg

Abstract. Recent research has emphasized the importance of reflection for students in an intelligent learning environment. This study tries to investigate whether agent prompts, acting as scaffolding, can promote students' reflection when they act as tutor through teaching the agent tutee in a learning-by-teaching environment. Two types of agent prompts are contrasted in this research, both from the perspective of a tutee, differing in their specificity. Reflective prompts are content-independent tutee questions, aiming at fostering students' general reflection on metacognitive strategies and beliefs. Interactive prompts, on the other hand, are content-dependent tutee questions that encourage students' specific reflection on domain-related and task-specific skills and articulation of their explanatory responses. The result indicates that designers on intelligent learning environment should concentrate on fostering students to reflect on their metacognitive strategies and beliefs, and allow students to take responsibility for directing their own learning autonomy.

Keywords: Reflection, Reflective Prompts, Scaffolding.

1 Introduction

Reflection, which is perceived as an active process of learning through experience, has been prominent in educational literature [1]. The ability to carry out meaningful reflection is considered as indicative of the highest level of deep learning [2]. E.g., Chi, Siler Jeong, Yamauchi, and Hausmann [3] find that only the number of student turns coded as "reflection", which were comprehension-monitoring statements, is positively correlated with deep learning gains in the face-to-face (FTF) tutorial dialogues.

Question prompts, as the literature suggests, can be an effective way of fostering reflection [2, 4], because they provide the cognitively complex ways learners think about, feel about, and make connections in experience [5]. Specially, recent research shows the evidence of learning benefits to tutors from tutee's question prompts in the context of peer tutoring [6-8]. Roscoe and Chi [8] note that tutee questions can motivate tutor explanations and metacognition, and thus have a significant and positive influence on the tutor's learning activities and opportunities.

In this paper, we investigate the use of agent tutee as an active and inquisitive learning partner to scaffold student to elicit general or specific reflection when taking the role of a tutor in an adapted learning-by-teaching agent environment (Betty's

Brain [9]). We compare two types of agent prompts to address the challenge to facilitate reflection of student tutor. Interactive prompts (IP), as the specific tutee questions, are content-dependent and provide students a structure through the learning-by-teaching process. They lead the students to complete the specific cognitive task and articulate their explanatory responses. On the other hand, reflective prompts (RP), as general tutee questions, are content-independent and stimulate students to monitor their learning-by-teaching processes and consider various perspectives and values regarding their learning-by-teaching activities.

2 Classroom Study

To investigate the impact of agent prompts to student's reflection and learning, we took a classroom study on 33 students from two local secondary schools (ages ranged from 13 to 15) on a voluntary basis. They were randomly assigned to one of three conditions. Eventually, 29 students (76%), 20 female (69%) and 9 male (31%) completed all activities of the experiment, resulting in the following division over the three conditions: no prompts (NP) condition as control group: $n = 10$, interactive prompts (IP) condition: $n = 10$ and reflective prompts (RP) condition: $n = 9$,

Figure 1 indicates that the three conditions were in the approximately same level in pretest while both the two prompted conditions (RP and IP) outperformed the non-prompted condition in the posttest.

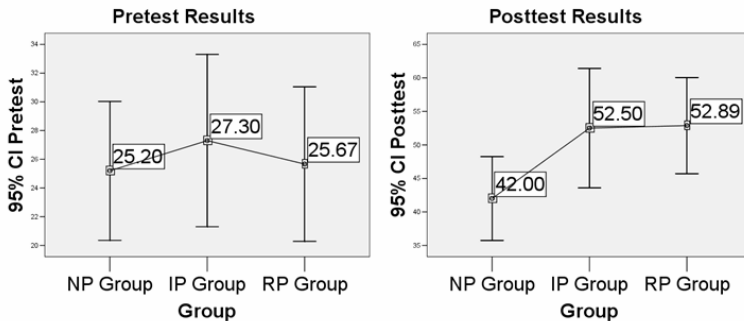


Fig. 1. Domain Knowledge Pre- and Post-Test Results

The ANOVA test of domain knowledge pretest-to-posttest gains indicated a significant effect of the two prompted conditions compared to the non-prompted condition ($F(2, 25) = 20.145$). The calculation of pre-test-to-posttest effect sizes (Cohen's d) showed a prominent difference between the IP group ($d = 2.37$) and the RP group ($d = 3.30$). However, the pair-wise comparison showed that there was no significant difference ($\text{Sig.} = .154$) between the IP and RP groups as to the learning gains from pretest to posttest.

The results of ANOVA tests of students' response statements to agent prompts showed the difference in the level of reflection between the RP and IP groups. The RP group was more likely to respond with contemplative statements representing a higher

level of reflection ($F(1, 17) = 20.015$, $\text{Sig.} < .05$). Comparatively, the IP group responded more with reacting statements representing a lower level of reflection which means they pay more attention to the task-specific aspects than the RP group ($F(1, 17) = 18.520$, $\text{Sig.} < .05$).

3 Conclusion and Future Work

In this study, we explored the inquisitive agent tutee as a learning partner in learning-by-teaching activities. Overall results in learning outcomes showed that the agent prompts did add value and encouraged student in reflection and achieving better learning outcomes because the prompted students performed better, on pretest-to-posttest gains than non-prompted students. Students generated better response statements more frequently when they received reflective prompts than interactive prompts. Based on the analysis of response statements, it was concluded that the reflective prompts did promote deeper contemplative reflection and interactive prompts elicited more reactive reflection. Future work of this study should be on exploring the relationship between agent prompts and intellectual flow, which will provide an in-depth understanding of intellectual enjoyment that students encounter when using agent tutee systems as learning partners.

References

1. Strampel, K., Oliver, R.: Using technology to foster reflection in higher education. In: *ICT: Providing choices for learners and learning*. Proceedings asilite 2007, Singapore (2007)
2. Moon, J.A.: *Reflection in Learning and Professional Development: Theory and Practice*. Kogan Page Limited, London (1999)
3. Chi, M.T.H., Siler, S., Jeong, H., Yamauchi, T., Hausmann, R.G.: Learning from human learning. *Cognitive Science* 25, 471–533 (2001)
4. Lai, G.: Examining the effects of selected computer-based scaffolds on preservice teachers' levels of reflection as evidenced in their online journal writing. PhD. Georgia State University, Atlanta, GA (2008)
5. Davis, E.A., Linn, M.: Scaffolding students' knowledge integration: Prompts for reflection in KIE. *International Journal of Science Education* 22(8), 819–837 (2000)
6. Cohen, P.A., Kulik, J.A., Kulik, C.C.: Educational outcomes of tutoring: A meta-analysis of findings. *American Educational Research Journal* 19(2), 237–248 (1982)
7. Graesser, A.C., Person, N.K., Magliano, J.P.: Collaborative dialogue patterns in naturalistic one-to-one Tutoring. *Applied Cognitive Psychology* 9, 495–522 (1995)
8. Roscoe, R.D., Chi, M.T.H.: Tutor learning: the role of explaining and responding to questions. *Instructional Science* 36, 321–350 (2008)
9. Leelawong, K., Biswas, G.: Designing learning by teaching agents: The Betty's Brain system. *International Journal of Artificial Intelligence in Education* 18, 181–208 (2008)

Identifying Problem Localization in Peer-Review Feedback

Wenting Xiong and Diane Litman

University of Pittsburgh

Abstract. In this paper, we use supervised machine learning to automatically identify the problem localization of peer-review feedback. Using five features extracted via Natural Language Processing techniques, the learned model significantly outperforms a standard baseline. Our work suggests that it is feasible for future tutoring systems to generate assessments regarding the use of localization in student peer reviews.

Keywords: peer-review, problem localization, Natural Language Processing.

1 Introduction

There is increasing interest in building systems such as SWoRD¹ to facilitate peer-review practices, which involve students writing essays on certain prompts, reviewing essays for their peers by providing feedback and then revising their previous draft essays based on the peer feedback. However, such systems do not tutor students to write better reviews. A study of a SWoRD corpus [1] shows that the helpfulness of the feedback (in terms of the likelihood of students' revising based on it) is significantly affected by certain feedback features, among which problem localization is most important. While such feedback features were used as mediators in the analysis of feedback helpfulness in [1], we believe that they could also be used as indicators in evaluating feedback quality automatically. As a first step, we focus on predicting problem localization based on the findings noted above, while our long-term goal is to enrich current peer-review systems with an assessment component on student reviewing performance. We illustrate the successful use of supervised machine learning to automatically identify the problem localization for a given piece of feedback based on features obtained using Natural Language Processing (NLP) techniques.

2 Data and Method

This study uses an annotated peer review corpus [1] collected from a college history class. It consists of 874 pieces of feedback expressing criticism accompanied

¹ Scaffolded Writing and Reviewing in the Discipline,
<http://www.lrdc.pitt.edu/schunn/sword/index.html>

by 24 corresponding essays. The feedback has been segmented at the idea-unit level and coded for problem localization as a binary feature ($Kappa=0.69$).

We developed four groups of features to capture different perspectives of localized expressions as follows:

Regular expression features: regTag

Simple regular expressions were employed to recognize common phrases of location (e.g., “on page 5”, “the section about”). If any regular expression is matched, the binary feature regTag is true.

Domain lexicon features: dwCNT

Using standard statistical NLP techniques provided by NLTK² (to extract frequent lexical bigrams from text), a dictionary of domain words was generated automatically from the collection of the 24 essays. We counted those words (dwCNT) contained in each piece of feedback.

Syntactic features: SO_domain, DET_CNT

Besides just counting the domain words, we also extracted information from the syntactic structure of the feedback sentences. We investigated whether there is any domain word between the subject and the object (SO_domain) in any sentence, and also counted demonstrative determiners (this, that, these and those) in the feedback (DET_CNT).

To illustrate how these features were computed, consider the sentence below, which is an idea unit that is coded as “problem localization = true”. The regTag is true because one regular expression is matched with “the section of”; dwCNT is 9, because the sentence contains “African” (2), “American”, “Americans”, “federal”, “governments”, “civil”, “political” and “rights”. There is no demonstrative determiner, thus DET_CNT is zero; “African Americans” is between the subject “section” and the object “attention”, so SO_domain is true.

***Example:** The section of the essay on African Americans needs more careful attention to the timing and reasons for the federal governments decision to stop protecting African American civil and political rights.*

Overlapping-window features: windowSize, overlapNum

The three types of features above are based on our intuition about localized expressions, while the following features are derived from an overlapping-window algorithm that was shown to be effective in a similar task – identifying quotation from reference works in primary materials for digital libraries [2]. To match a possible citation in a reference work, it searches for the most likely referred window of words through all possible primary materials. We applied this algorithm for our purpose, and considered the length of the window (windowSize) plus the number of overlapped words in the window (overlapNum).

3 Results

Our binary classifier of problem localization is learned by using the Decision Tree (J48) algorithm provided by WEKA³ based on the features explained in the

² Natural Language Toolkit: <http://www.nltk.org/>

³ <http://www.cs.waikato.ac.nz/ml/weka/>

Metric	Baseline	Learned model
Accuracy	0.529	0.774 *
Precision	0.279	0.779 *
Recall	0.529	0.773 *
Kappa	0	0.549 *

Fig. 1. Performance of identification of problem localization. * indicates $P < 0.05$.

```

regTag = False
| dwCNT <= 5: false
| dwCNT > 5:
| | windowSize <= 20
| | | SO_domain = True: true
| | | SO_domain = False
| | | | DET_CNT <= 0: true
| | | | DET_CNT > 0: false
| | windowSize > 20: true
regTag = True: true

```

Fig. 2. Learned decision tree

previous section. We evaluated our model via 10-fold cross validation and compared its performance against a standard baseline – majority class (always predict “true”).

The results presented in Fig 1. show that our model performs significantly better than the baseline. Accuracy is 77.4% (against 52.9%), and both precision and recall are around 77% (against 27.9% and 52.9% respectively). Fig. 2 shows the decision tree based on all the features we investigated. WEKA automatically selects the most powerful features (i.e. regTag, dwCNT, windowSize, SO_domain, DET_CNT) and ignores the less useful ones. The learned model first uses regular expressions to recognize the localized feedback; for feedback whose regTag is false, it then looks at the occurrences of domain words. For domain-word counts greater than 5, the overlapped content between feedback and its targeting essay is then considered, and so on.

4 Conclusion and Future Work

In this paper, we proposed a model for detecting problem localization of peer-review feedback. We found simple NLP techniques (i.e. regular expressions, lexicon dictionaries and text mapping) are effective in our identification task.

In future work, we would like to explore more sophisticated NLP techniques to improve our current model; we would also like to investigate how to generate assessment regarding problem localization based on the noisy output of the model. Furthermore, we hope to incorporate this assessment component into the peer-review system (e.g. SWoRD) so as to provide meaningful feedback for students to enhance their reviewing skills in focused aspects (currently it is just problem localization) from the peer review assignment.

Acknowledgement. We thank LRDC for financial support, M. Nelson and C. Schunn for the corpus, and Rebecca Hwa for the guidance.

References

1. Nelson, M.M., Schunn, C.D.: The nature of feedback: how different types of peer feedback affect writing performance. *Instructional Science* 37, 375–401 (2009)
2. Ernst-Gerlach, A., Crane, G.: Identifying quotations in reference works and primary materials. In: Christensen-Dalsgaard, B., Castelli, D., Ammitzbøll Jurik, B., Lippincott, J. (eds.) *ECDL 2008. LNCS*, vol. 5173, pp. 78–87. Springer, Heidelberg (2008)

AlgoTutor: From Algorithm Design to Coding

Sung Yoo and Jungsoon Yoo

Computer Science Department
Middle Tennessee State University, United States
{cssung, csyoojp}@mtsu.edu

Abstract. Problem solving using a programming language such as C++ is a complex multi-step task. AlgoTutor trains introductory computer science students to start the problem solving process with algorithm design. The system then helps students learn how to trace an algorithm with an execution trace visualization tool. The final step of implementing an algorithm in code is accomplished with the ProgramPad portion of AlgoTutor. In ProgramPad students can see the connection between the algorithm and the converted C++ program. We have shown that some of the difficult programming concepts such as function parameters and array concepts can be addressed using ProgramPad. An additional benefit of AlgoTutor is that students can practice algorithm development skills and algorithm implementation skills in a web-based, interactive environment. The system provides online feedback for their algorithm so that students can work at their own pace at a convenient time.

Keywords: algorithm development, learning environment.

1 Introduction

Teaching problem solving is a primary goal of introductory computer science courses. However, problem solving using a computer programming language such as C++ involves several steps:

- step 1. Develop an algorithm for a given problem.
- step 2. Verify the correctness of the algorithm by tracing each step of the algorithm.
- step 3. Convert the algorithm into a C++ program.
- step 4. Verify whether the converted program is syntactically correct.
- step 5. Test and debug the program's correctness.

Steps 4 and 5 are done with the help of compilers and debuggers; however, steps 1, 2, and 3 are typically done manually. In this paper, we describe components of AlgoTutor that we have developed to help students with these manual steps.

2 AlgoTutor: Algorithm Tutor

AlgoTutor is a pedagogical tool that trains introductory computer science students to start the problem solving process with algorithm design using pseudo code. The

system has been developed to address the undesirable attitude that developing algorithms prior to writing code is not necessary [4]. *The algorithm composer* which is the major component of AlgoTutor can help students develop an algorithm in step 1 [2]. Algorithms are constructed by assembling instructor defined pseudo code (predefined operations) and control structures via a graphical drag-and-drop interface. The drag-and-drop features of Alice [1] were a major inspiration in the design of the AlgoTutor interface. In the process of developing an algorithm, students may need help when they run into a problem that cannot be resolved on their own. AlgoTutor provides online grading and feedback so that students can check immediately whether or not their algorithm is correct and what mistakes, if any, they made. The system employs both top-down and bottom up approaches so that students are able to practice these techniques on various problems.

Even though teaching algorithmic problem solving is a typical goal of the computer science I (CS-I) course, much of the teaching effort is expended in teaching correct syntax and use of the programming environment. Shackelford [3] observed that the details of a particular programming language tended to “annoy and distract attention from the core issue of algorithmic problem solving.” Furthermore, even when they learn to construct algorithms on paper, novice students are typically unable to follow the execution of the algorithm to check for its correctness. AlgoTutor helps students learn how to trace an algorithm with an execution trace visualization tool, *the algorithm tracer* in step 2. The algorithm tracer is a subcomponent of the algorithm composer that allows students to verify the correctness of their algorithm or to locate possible errors by providing step-by-step visual tracing of the student algorithm. The algorithm tracer visualizes the execution of the algorithm by stepping one operation or control structure part at a time. The student can follow the control flow and monitor changes in variable values.

The algorithm developed using the composer eventually needs to be converted to code for implementation. Students in CS-I often don't see the relationship between the algorithm and the code. For this reason and step 3, we have developed another component of AlgoTutor called *ProgramPad* which converts most of the operations/steps in the algorithm into C++ code and provides an interface for completing the program. By observing the actual conversions in ProgramPad, students can better understand the connections between the algorithm that is developed using the composer and the C++ program that is converted from the algorithm. ProgramPad also provides most of capabilities of a simple IDE: edit, save, load, build, run, and print. The programming concepts that can be addressed with ProgramPad are: variable declaration, syntax of control structures, local variables vs. function parameters, the syntax related to function, testing a function using a driver, and implication of abstract operations as well as implication of bottom-up and top-down approaches.

We used AlgoTutor with the algorithm tracer and ProgramPad for several labs during the fall of 2009 including while-loop, function, and array concepts. Participants in these studies were students enrolled in the CS-I course. To assess the students' perception of the usefulness of the algorithm tracer and ProgramPad, exit survey responses were collected online after each lab. The survey question related to the algorithm tracer was “The tracing feature helped me verify my algorithm.” A total of 47 students participated in the exit survey, 74.5% of the students agreed or strongly agreed with the question and only one student disagreed. The exit survey also asked

students to provide any comments related to their experience with the system, and most of the comments were positive. Negative comments were mostly related to very specific situations such as a system glitch which has been addressed, misunderstanding problem specification, or inability to solve that specific problem.

Students were also asked whether the exercise helped them understand the targeted concept. Survey results for the while-loop concept (79% agreed) were better than the function concepts (57% agreed). We think it is because the programming concepts related to function are more difficult than the while-loop concept. For the question “ProgramPad helped me understand the connection between the algorithm and the C++ code,” 58% found it helpful. For the question “Did ProgramPad help you understand the process of developing a program using the algorithm for this problem?” 78% found it helpful. We have also measured the effectiveness of these labs using AlgoTutor, the complete results of the experiment are reported in [5].

3 Conclusion and Future Work

Problem solving using a programming language such as C++ is a complex multi-step task which requires careful planning and many programming skills. Using AlgoTutor, students can practice algorithm development skills and algorithm implementation skills in a web-based, interactive environment. The system provides online feedback for their algorithms so that students can work at their own pace at a convenient time. The system also provides a convenient graphical user interface for teachers so that they can prepare exercises, assess student performance, and get insight for the improvement of student learning. Our survey shows that students feel that AlgoTutor helped them organize their thoughts and develop algorithms. In the upcoming semesters we plan to run more experiments with new features such as reviewing related concepts and problem specifications. We are also investigating ways to promote student thinking and engaging student interest.

References

1. Cooper, S., Dann, W., Pausch, R.: Alice: A 3-D tool for introductory programming concepts. In: Proceedings of the 5th Annual CCSC Northeastern Conference, Ramapo, NJ (2000)
2. Pettey, C., Yoo, J., Yoo, S., Seo, S., Dong, Z.: A Tool for Promoting Algorithm Development in Introductory CS Classes. In: Proceedings of ED-MEDIA 2009, Honolulu, Hawaii, pp. 87–95 (2009)
3. Shackelford, R.L.: Introducing computer science fundamentals before programming. In: FIE 1997: Proceedings of the Frontiers in Education Conference, 1997. On 27th Annual Conference. Teaching and Learning in an Era of Change, Washington, DC, USA, vol. 1, pp. 285–289. IEEE Computer Society, Los Alamitos (1997)
4. Yoo, J., Dong, Z., Yoo, S., Seo, S., Pettey, C.: Improving Student Performance by Enforcing Algorithm Development. In: Proceedings of ED-MEDIA, Honolulu, Hawaii, pp. 119–127 (2009)
5. Yoo, J., Pettey, C., Seo, S., Yoo, S.: Teaching Programming Concepts Using Algorithm Tutor. In: Proceedings of ED-MEDIA 2010, Toronto, Canada (2010)

Adaptive, Assessment-Based Educational Games

Diego Zapata-Rivera

Educational Testing Service, Princeton, NJ 08541 USA
DZapata@ets.org

Abstract. Assessment-based educational games can produce useful information to guide student instruction. This paper describes an approach for integrating components of video games with those of adaptive technologies and assessment into the design of educational games. Three examples in the areas of English language learning and mathematics are also presented.

Keywords: Adaptive technologies, assessment and video games.

1 Introduction

Researchers have proposed applying adaptive techniques to the development of educational games (e.g., [1, 2]). Recent advances in assessment and learning technologies make it possible to develop adaptive learning systems that use assessment information from different sources to guide student learning (e.g., [3]).

Educational video games can be used to engage students in learning inside and outside of the classroom (e.g., [4]). However, in order to better address educational goals, video games must provide evidence that learning of valued skills or knowledge takes place in the game. The use of valid embedded assessments may help video games become valid instructional instruments.

We have developed assessment-based gaming and learning environments that employ an evidence-based methodology that reconciles the needs for obtaining valid assessment information and creating engaging interactive tools that students want to use. This paper describes the evidence-based approach and three instances of learning and gaming environments implemented by applying it.

2 Assessment-Based Learning and Gaming Environments

A major goal of assessment-based learning and gaming environments is to provide adaptive gaming scenarios that can be used to help students learn and provide valid assessment information to students and teachers.

Gaming scenarios are composed of various interactive activities (i.e., assessment tasks). Each scenario has an underlying storyline aimed at defining: (a) the behavior to be observed and (b) the interactive activities needed to elicit such behaviors. Creating such scenarios requires input from an interdisciplinary team including users (i.e., students or players), domain experts (e.g., teachers and researchers), assessment specialists and interactive design experts. This development process encompasses the

following activities: (a) gather domain knowledge information; (b) design initial competency and evidence models; (c) select initial competencies and required evidence to focus on; (d) brainstorm about scenarios and activities that can be used to elicit desired behavior; (e) describe scenarios and activities (i.e., define the role of the student, the role of the teacher, the role of the pedagogical agents, level of feedback or scaffolding, assessment activities to be administered in particular situations, establish work products for each activity, and describe the evidence rules for the activity); (f) update task models, competency and evidence models; and (g) iterate until all the target competences have been covered. Once a scenario is described, interactive design experts and system developers create a prototype, pilot test it with users, and make changes based on the feedback that is gathered. More information about this process can be found in [6]. Next we describe three assessment-based learning and gaming environments: English ABLE [5], English and Math ABLE (EM ABLE) [6] and The Request Game [7].

2.1 English ABLE

English ABLE (Assessment-Based Learning Environment) uses assessment information to support student learning of English grammar. English ABLE draws upon a database of TOEFL® CBT tasks to create new packages of enhanced tasks that are linked to particular component ELL skills. In English ABLE, students try to help a virtual student (Carmen or Jorge) learn English by correcting this student's writing from a notebook of facts (sentences –enhanced TOEFL® tasks). To make the game more compelling Carmen and Jorge are able to express basic emotions, which are triggered by a list of predefined rules that take into account recent student performance on particular tasks. A character named Dr. Grammar provides adaptive instructional feedback (i.e., rules, procedures, examples and definitions) based on the student model.

English ABLE implements a Bayesian student model that divides English grammar into three main categories: use, form and meaning. The Bayesian model is used to capture and propagate evidence of student knowledge regarding some aspects of English grammar including sentence-level grammatical concepts (e.g., agreement) as well as word-level concepts (e.g., individual parts of speech). Tasks are linked to grammar concepts using IRT (Item Response Theory) task parameters.

2.2 EM ABLE

EM ABLE (English and Math ABLE) models both English language and math competencies. It combines game elements (e.g., immediate feedback, sound effects, and progress indicators: points and power levels), pedagogical agents and various forms of scaffolding. The game starts when the student chooses and customizes a student character with which to play the game. The student also selects a friend to accompany the character while playing the game. The student's mission is to help his/her student character interact in the EM (English-Math) "city." The student character is invited to participate in various activities (e.g., a pizza party). Each activity provides an integrated learning and assessment scenario for the student. As part of each activity, the student character interacts with virtual people who provide guidance, feedback, and, at the same time, administer embedded assessment tasks to the learner related to predefined

vocabulary and math proficiencies. Evidence of student knowledge is obtained through the student's interaction with these characters and his/her performance on various math and vocabulary activities. Activities vary in difficulty based on the student's prior performance and include short, text-based dialogues using a virtual cell phone (i.e., conversations) as well as math completion tasks (i.e., math activities). As the learner advances in the game, s/he accumulates points for his/her student character.

EM ABLE implements a Bayesian student model. Knowledge-level estimates (i.e., power levels) are continuously updated based upon performance and are visible to the learner through his/her virtual cell phone. These power levels are externalized as progress bars (one for vocabulary and one for math) and are referred to as the character's knowledge levels.

2.3 The Request Game

The Request Game is a prototype of an assessment-based educational game aimed at supporting non-native English speakers' need for pragmatic instruction. This game allows users to engage in interactive written dialogue with a virtual professor (or pedagogical agent) in multiple academic contexts. Students explore contextually and socially appropriate request strategies while the system scores each attempt, assigns points, and provides immediate and summative feedback. The Request Game implements a finite automata dialogue engine that is used to recognize student utterances and determine the next actions of the virtual professor.

Usability studies have been conducted using these assessment-based educational games. Initial evidence shows that students enjoy interacting with them and teachers appreciate the evidence of student performance provided by the system. Future work includes exploring student learning effects in controlled contexts.

References

1. Peirce, N., Conlan, O., Wade, V.: Adaptive Educational Games: Providing Non-invasive Personalised Learning Experiences. In: Second IEEE International Conference on Digital Games and Intelligent Toys Based Education, pp. 28–35 (2008)
2. Carro, R., Breda, A., Castillo, G., Bajuelos, A.: A methodology for developing adaptive educational-game environments. In: De Bra, P., Brusilovsky, P., Conejo, R. (eds.) AH 2002. LNCS, vol. 2347, pp. 90–99. Springer, Heidelberg (2002)
3. Razzaq, L., et al.: The Assistentment Project: Blending Assessment and Assisting. In: Proceedings of the 12th Artificial Intelligence in Education, pp. 555–562. ISO Press (2005)
4. Klopfer, E., Osterweil, S., Salen, K.: Moving Learning Games Forward. The Education Arcade. MIT, Cambridge (2009)
5. Zapata-Rivera, D., VanWinkle, W., Shute, V., Underwood, J., Bauer, M.: English ABLE. *Artificial Intelligence in Education* 158, 323–330 (2007)
6. Zapata-Rivera, D., VanWinkle, W., Doyle, B., Buteux, A., Bauer, M.: Combining Learning and Assessment in Assessment-based Gaming Environments: A Case Study from a New York City School. *Journal: Interactive Technology and Smart Education* 6(3), 173–188 (2009)
7. Yang, H., Zapata-Rivera, D.: An Exploratory Study into Interlanguage Pragmatics of Requests: A Game of Persuasion. ETS Research Report RR-09-13. ETS, Princeton (2009)

ITS Authoring through Programming-by-Demonstration

Vincent Aleven, Brett Leber, and Jonathan Sewall

Human-Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA USA 15213
{aleven,bleber,sewall}@cs.cmu.edu
<http://ctat.pact.cs.cmu.edu>

The Cognitive Tutor Authoring Tools (CTAT) [1] are a suite of software programs meant to make the creation of web-based ITS practical for non-programmers. CTAT supports a relatively novel type of tutors, called *example-tracing* tutors, which use examples of problem-solving approaches to assess and guide students as they practice solving problems. CTAT employs a programming-by-demonstration paradigm that relies on creating examples of how problems are to be solved, rather than defining general rules or constraints that characterize solutions or solution processes. The result is an *editable behavior graph* that contains the tutor's intelligence about how to react to student actions and what hints to give for next steps. Although relatively easy to build, example-tracing tutors support the key behaviors identified by VanLehn [2] as characteristic of ITS. Data from over 26 research studies using CTAT indicate that these tools lower the cost of ITS development by a factor of 4-8.

The interactive event will illustrate the development of an example-tracing tutor intended to help elementary school students learn a procedure for adding fractions. Event participants will see how the tutor's author used several advanced CTAT authoring techniques to provide the following behaviors in the tutor:

- Knowledge tracing: authors can induce the ITS to trace students' skill acquisition by annotating steps with the cognitive skills they require.
- Multiple paths and interpretations: authors can demonstrate any number of different sequences of steps, to represent different solution strategies.
- Formulas and variables: to extend the tutor's generality, authors can specify a formula to calculate how the specific value on any step depends on prior steps.
- Fine-grained ordering constraints: authors can choose groups of steps and specify whether the student must perform them in the given sequence; groups can be nested, to provide, e.g., unordered subgroups within ordered groups.

Participants will be encouraged to make suggestions regarding these and other CTAT facilities that they think would improve this suite of ITS authoring tools.

References

1. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A new paradigm for intelligent tutoring systems: Example-tracing tutors. *International Journal of Artificial Intelligence in Education* 19(2), 105–154 (2009)
2. VanLehn, K.: The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education* 16(3), 227–265 (2006)

A Coordinate Geometry Learning Environment with Game-Like Properties

Dovan Rai, Joseph E. Beck, and Neil T. Heffernan

Computer Science, Worcester Polytechnic Institute
{dovan, josephbeck, nth}@wpi.edu

Although educational games can be engaging and motivating for students, the game aspects tend to take up time that could have been used for instruction, and they add to the cognitive load of the students. Therefore, instead of completely integrating educational content into a game framework, we instead choose to start with a computer tutor and selectively incorporate those features of games that are engaging but do not overly detract from learning. Following this approach, we created a learning environment for coordinate geometry with *game-like properties*. We define game-like properties as the elements of games such as visual representation, cover story, and animated feedback, which are responsible for their engaging nature. While there have been many attempts to use game-like environment in tutors (e.g. Wayang Outpost), our approach differs by using an iterative process in a space of ITS and games, within the framework of stimulation and overload.

We are taking a measured and minimalist approach to make a balance between stimulation and overload. We believe that regular games can afford a complex game environment with complicated game rules because there the purpose is for students to learn the game. However, in educational games, the additional cognitive load could be a hindrance as students need mental resources to learn the domain. Our aim is to use game-like properties to create an environment for rich learning experiences that arouse emotional as well as cognitive interest among students. Solving math problems in real-world scenarios can be more interesting and meaningful. Concrete pictures help to make mental models of the context; immediate visual feedback enables students to see how their response relates to the correct solution. Using a coherent cover story can help reduce the cognitive load on the learners by saving the effort to learn an entire new context for each question. Based on these ideas, we have created a flash-based environment for coordinate geometry wrapped in a visual cover story.

Currently, we have two cover stories: *Mily's World* and *Monkey's Revenge*. In *Monkey's Revenge*, a boy is thrown out of class for playing a game on his cell phone; he then encounters a monkey who steals his phone and runs away. To move the story forward, students must solve coordinate problems like calculating the slope of the roof and walls of the house, finding points where the monkey tied to a rope cannot reach the bananas, and finally figure out slopes, intercepts and equations of the line of the path of the ball. We have constructed mathematically identical problems in *Mily's World*, where students help a younger character, Mily to build a doghouse and play soccer. We contrast student experiences in these two environments, and discuss which game-like properties are predictive of students' perception of the environment; demos of these are available at <http://users.wpi.edu/~dovan/coordinates.html>

Adaptive Tutorials and the Adaptive eLearning Platform

Dror Ben-Naim

School of Computer Science and Engineering, University of New South Wales,
Sydney, Australia
drorb@cse.unsw.edu.au
<http://www.adaptiveelearning.com>

In this interactive demo we will present the Adaptive eLearning Platform – a web-based solution for the development, deployment and analysis of Adaptive Tutorials. Adaptive Tutorials (AT) are online instructional activities that exhibit three levels of adaptivity: students experience adaptive feedback with remediation targeted to their intrinsic misconceptions, while their activities are also sequenced adaptively based on performance. The third level of adaptivity is content adaptation through analysis and reflection.

The AeLP is an implementation of Virtual Apparatus Framework (VAF) - a content development paradigm inspired by the simplicity and elegance of the teaching laboratory. Its premise is that teachers should be able to develop electronic courseware in a way that is analogous to how they develop laboratory activities. In other words, they need not be concerned about building the software or understanding exactly how it works, but rather they should be able to import prefabricated “apparatus” into a learning environment, and then author lesson plans that guide students through interaction with the apparatus. VAF’s basic building blocks - Virtual Apparatus (VA) - are virtual equivalents to real-world laboratory equipment. As such they include simulations or software tools with which students interact in the context of an online, laboratory-like activity.

The AeLP has been fielded since 2006 at the University of New South Wales, Sydney Australia, where Adaptive Tutorials developed using the AeLP have been incorporated into the syllabi of 10 major courses (ranging between 50 to 700 students per semester), and are accessed by over 3000 students per semester.

The purpose of the interactive event is to present the lifecycle workflow associated with Adaptive Tutorials – pedagogical design, authoring, deployment, reflection and adaptation, and to show how in each step, teachers are supported as first-class participants in the process. Different tools that are used to assist teachers in their tasks will be presented. For example the Adaptive Tutorial Analyzer which is a set of data-visualization and data-mining tools is used for the purpose of Reflection and Adaptation.

Participants in this interactive event will learn how adaptive educational activities can be authored in VAF, which we believe is a powerful ITS design paradigm. We are also seeking collaborators who are interested in using Adaptive Tutorials in their academic institutions.

DomainBuilder – An Authoring System for Visual Classification Tutoring Systems

Eugene Tseytlin, Melissa Castine, and Rebecca Crowley

Department of Biomedical Informatics, University of Pittsburgh School of Medicine
tseytlin@pitt.edu, {castinem2, crowleyrs}@upmc.edu

In previous work, we developed SlideTutor - an Intelligent Tutoring System that teaches visual classification problem-solving in Pathology, and shown that use of the system is associated with rapid learning gains. Development of the SlideTutor system and content has required many years of effort by system developers, knowledge engineers and domain experts. Both cases and domain ontologies must be manually created and validated. The scope of medical knowledge that must be covered is extremely large. The further development of tutoring systems for visual classification in medical domains (including our own) requires software that reduces this high development burden. Towards this goal, we sought to create a generic framework for developing visual classification tutoring systems in medical fields such as Pathology or Radiology. In this interactive event, we present the first and most difficult step towards such a generic visual classification ITS authoring system – the component for creating and validating cases and domain ontologies.

Design Objectives: The following design objectives were developed based on our analysis of the domain, in conjunction with a review of previous systems:

- Combine knowledge authoring, case authoring and validation tasks into a single work environment, allowing users to transit between tasks easily, enabling bottom-up, top-down, and hybrid authoring strategies
- Provide an intuitive graphical user interface (GUI) for authoring that enables any user with domain knowledge to construct domain ontologies and cases without the need to understand the underlying, complex OWL knowledge representation
- Allow authors to create complex diagnostic rules using familiar tabular representations
- Integrate Natural Language Processing (NLP) methods for parsing existing clinical reports to speed case authoring
- Streamline digital image annotation work flow to enable easy tagging of image annotations to concepts in the domain ontology
- Provide validation during case authoring, by leveraging ontology validation tools to infer diagnosis from the diagnostic findings provided by the case author, advising user when domain ontology and case authoring are inconsistent
- Support collaboration between knowledge engineers and domain experts by easily visualizing complex relationships between concepts in domain ontology

- Support collaboration between multiple authors and institutions
- Reuse existing resources such as ontologies from BioPortal, synonyms and definitions from Enterprise Vocabulary Service (EVS)

Details are available at http://slidetutor.upmc.edu/domainbuilder/DomainBuilder_ITS.pdf

Currently, all cases and knowledge developed for the SlideTutor system are developed with DomainBuilder. In future work, we will integrate DomainBuilder with additional tutor authoring components to produce a generic framework for creating visual classification tutoring systems.

AWESOME Computing: Using Corpus Data to Tailor a Community Environment for Dissertation Writing

Vania Dimitriva, Royce Neagle, Sirisha Bajanki, Lydia Lau, and Roger Boyle

School of Computing, University of Leeds, UK

Keywords: Dissertation writing, Ill-defined domains, Social semantic web, Learning communities.

This demonstration will present a novel community environment ‘AWESOME Dissertation Environment (ADE)’ which uses semantic wikis to implement the pedagogical approach of ‘social scaffolding’. ADE was developed within an interdisciplinary UK research project called AWESOME (Academic Writing Empowered by Social Online Mediated Environments) which involved the universities of Leeds, Coventry and Bangor¹. The environment was instantiated in several domains: Education, Fashion and Design, Philosophy and Religious Studies, and an Academic Writing Centre. Following both the encouraging feedback from the trial instantiations and the challenges faced in deploying the environment in practice, we conducted a second stage of the project which aimed at adapting the ADE to dissertation writing in computing. Following the lessons learnt from the first stage, we now performed a systematic approach to tailor the existing community environment to meet dissertation writing needs in a specific domain and in a particular educational practice.

Dissertation writing, which is a major challenge faced by most students in the higher education, is an example of soft skill training. Training of soft skill is becoming paramount in today’s educational and societal climate, and receives increasing attention in the area of intelligent learning environments for ill-defined domains. A fundamental step in developing such environments is to articulate *what problems learners are facing and how to shape the learning environment to effectively address these problems*. Both issues have been examined in the tailoring of ADE for Computing. Based on a study that analyzed written feedback given to undergraduate students by tutors at a key stage of dissertation preparation, we identified main hurdles faced by students during the dissertation journey and collected examples of tutor discourse used to facilitate the dissertation process. This allowed us to tailor ADE by:

- extending a core dissertation writing with specific problems faced by this community;
- integrating examples of previous dissertations;
- seeding the environment with content that corresponds to typical student problems and tutor feedback.

We will present example scenarios of students and tutors interacting with the AWESOME Computing environment, which illustrate the process of social scaffolding.

¹ See the AWESOME web site <http://awesome.leeds.ac.uk/> for more information.

Collaboration and Content Recognition Features in an Inquiry Tutor^{*,**}

Mark Floryan, Toby Dragon, Beverly Woolf, and Tom Murray

University of Massachusetts Amherst
140 Governors' Dr. Amherst, MA USA
{mfloryan, dragon, bev, tmurray}@cs.umass.edu

Abstract. This demonstration will show how a tutor can detect the content of collaborative behavior and offer relevant domain level interventions. Rashi is a domain independent intelligent tutor providing students with practice using inquiry skills. When working on human biology, students interact with a virtual sick patient whom they must successfully diagnose. Rashi supports students as they create hypotheses and collect data to support and refute these hypotheses. In order to increase the efficacy of Rashi, we incorporated collaborative tools that support group efforts by supporting students as they dynamically share experiences and work together to reach a diagnosis. In addition to this, Rashi contains an intelligent agent that examines collaborative efforts and automatically detects the expert knowledge students are working with. Visitors to this demo will first explore these collaborative tools in detail. Two people will collaborate about a diagnosis and the intelligent agent will examine their collaborative activity and compare it with an expert knowledge base, to determine what domain content is relevant to their activities. The tutor will provide interventions to the visitors that leverage this content recognition.

This demonstration provides evidence that expert knowledge bases are a plausible development option for intelligent tutoring systems because they can leverage content recognition to provide more useful feedback. In Rashi, some collaborative content is recognized when students manually match discussion items to expert knowledge. However, the greatest impact comes when the tutor recognizes participants' content by matching words and phrases in the chat conversation. Experiments show that the tutor can recognize this content correctly with more than 70% accuracy. Thus, it can provide interventions that suggest what direction students might take if they reached an impasse.

This demonstration provides evidence that complicated NLP techniques are not always necessary; a tutor can understand domain level student activity and provide useful interventions using a well-built expert knowledge base. In addition, we show that even though the lack of more complicated techniques may lead to some error in content recognition, we can provide unique forms of feedback that are not detrimental to students when the content is incorrectly recognized, but is significantly helpful when it is correctly recognized.

* The project webpage (including software download) can be found at:
<http://rashi.cs.umass.edu>

** The situated scenario for this demonstration can be downloaded at:
http://www.cs.umass.edu/~mfloryan/ITS2010/InteractiveEvent/Rashi_Scenario.pdf

The Science Assistments Project: Scaffolding Scientific Inquiry Skills

Janice D. Gobert^{1,2}, Orlando Montalvo¹, Ermal Toto¹,
Michael Sao Pedro², and Ryan S.J.d. Baker^{1,2}

¹ Department of Social Sciences and Policy Studies ² Department of Computer Science
Worcester Polytechnic Institute
100 Institute Road, Worcester, MA USA
{jgobert, amontalvo, toto, mikesp, rsbaker}@wpi.edu

Keywords: intelligent tutoring, scientific inquiry.

We present our computer-based learning environment, Science Assistments (http://users.wpi.edu/~sci_assistments/; NSF-DRL # 0733286; NSF-DGE #0742503; U.S. Dept of Ed. # R305A090170), for Physics, Life Science, and Earth Science that scaffolds middle school students' scientific process skills, namely, hypothesis-generation, design of experiments, data collection, data interpretation, and warranting claims with evidence. Our project builds on prior development by the investigators of the Math Assistments project (<http://www.assistent.org/>). Specifically, we utilized the existing authoring functionality of the Math Assistments system and extended the logging functionality in order to capture students' fine-grained actions within interactive microworlds. In addition, we developed a suite of inquiry tools to support students' inquiry in terms of the five skills mentioned above. Together, the logging functionality and the inquiry tools provide the basis for adaptive scaffolding of students' inquiry in real time. By reacting to students' inquiry strategies in real time, we hypothesize that it will be possible to positively affect both students' science process skills, shown by more goal directed inquiry and more systematic experimentation, measured through log files, as well as students' content learning, as measured by pre-post test gains. We plan to test our adaptive scaffolding in a series of randomized controlled studies in our four partner schools; the demographics of these students represent a wide range of SES and ethnic backgrounds, and thus, our data should generalize well. Goal outcomes include empirical data regarding the efficacy of our system at improving students' science learning, namely, inquiry skills and content learning, across several dependent measures in each content domain.

Incorporating Interactive Examples into the Cognitive Tutor

Robert G.M. Hausmann, Steven Ritter, Brendon Towle, R. Charles Murray,
and John Connelly

Carnegie Learning, Inc., 437 Grant Street, Pittsburgh, PA 15219
{bhausmann, sritter, btowle, cmurray,
jconnelly}@carnegielearning.com

Description

Mixing worked-out examples with problem solving has been shown to be an effective blend of educational activities [1]. Given the positive impact on learning, some Intelligent Tutoring Systems attempt to incorporate worked-out examples into their learning environments. The approach taken by Carnegie Learning's Cognitive Tutor, called *Interactive Examples*, was created to solve two design challenges.

The first design challenge was to encourage students to actively process each step of the example. Too often, students passively attend to the example [2], especially when the examples span a year-long curriculum. To address this issue, the Interactive Example requires that the student perform each step. The steps are highly scaffolded, with the tutor automatically providing the answer after two errors. This approach ensures that students work through the example quickly.

The second challenge was to support the student behavior of switching between examples and problem solving [3]. We dealt with this challenge by implementing a *breadcrumb mode*. After the student completes the example, she is able to see an ordered sequence of steps in the form of small numbered icons, or *breadcrumbs*, near each user-interface element where a problem-solving action occurred. She can click on individual breadcrumbs at will, for a review of the instructional text explaining how or why each corresponding step was taken.

The Interactive Event demonstrated each of these features by sampling across the commercial product, including units from Geometry and Algebra II. A version of the demonstration can be found at <http://www.carnegielearning.com/its2010>

References

1. Paas, F.G.W.C., Van Merriënboer, J.J.G.: Variability of worked examples and transfer of geometry problem-solving skills: A cognitive-load approach. *Journal of Educational Psychology* 86, 122–133 (1994)
2. Ross, B.H., Kilbane, M.C.: Effects of principle explanation and superficial similarity on analogical mapping in problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 23, 427–440 (1997)
3. Renkl, A., Atkinson, R.K., Grosse, C.S.: How fading worked solution steps works – A cognitive load perspective. *Instructional Science* 32, 52–82 (2004)

Acquiring Conceptual Knowledge about How Systems Behave

Jochem Liem¹, Bert Bredeweg^{1,*}, Floris Linnebank¹, René Bühling²,
Michael Wißner², Jorge Gracia del Río³, Wouter Beek¹, and Asunción Gómez Pérez³

¹ University of Amsterdam, Informatics Institute, Amsterdam, Netherlands

B.Bredeweg@uva.nl

² University of Augsburg, Multimedia Concepts and Applications, Augsburg, Germany

³ Universidad Politécnica de Madrid, Ontology Engineering Group, Madrid, Spain

There is a need for software that supports learners in actively dealing with theoretical concepts by having them create models and perform concept prediction and explanation (e.g. [3,4,5]). DynaLearn seeks to address this by developing a domain independent Interactive Learning Environment (ILE) based on Qualitative Reasoning (QR) [1]. The QR vocabulary fits the nature of *conceptual* knowledge, and the explicit representation of these notions in the software provides the handles to support an automated communicative interaction that actually discusses and provides feedback at the *conceptual* level.

DynaLearn seeks to provide an instrument for studying the characteristics under which learners develop conceptual knowledge, particularly for ill-defined domains. The DynaLearn ILE offers a suite of technical advances for educational research. The following features can be manipulated (see [2] for design specifications). (i) *Use-levels* adapt the interface and tool interaction to tailor for groups of a specific age or experience. (ii) Different types of *knowledge-based feedback*, such as recommending terminology, model quality feedback, and suggestions for model improvements. (iii) Learner interaction through *virtual characters* with roles such as student, peer, teacher, critic and quizmaster. More specifically the following features will be demonstrated during the interactive event: (a) Workbench and its multiple use-levels, (b) Basic help, (c) Grounding, (d) Teachable agent, and (e) Quiz. The audience will be allowed to work with the software as if they were students.

References

1. Bredeweg, B., Linnebank, F., Bouwer, A., Liem, J.: Garp3 — Workbench for Qualitative Modelling and Simulation. *Ecological Informatics* 4(5-6), 263–281 (2009)
2. André, E., Bee, N., Bühling, R., Gómez-Pérez, J.M., Häring, M., Liem, J., Linnebank, F.: Technical design and architecture. In: Bredeweg, B. (ed.) *DynaLearn*, EC FP7 STREP project 231526, Deliverable D2.1 (2009)
3. Hucke, L., Fischer, H.E.: The link of theory and practice in traditional and in computer-based university laboratory experiments. In: Psillos, D., Niedderer, H. (eds.) *Teaching and learning in the science laboratory*, pp. 205–218. Kluwer, Dordrecht (2002)
4. Otero, V., Johnson, A., Goldberg, F.: How Does the Computer Facilitate the Development of Physics Knowledge Among Prospective Elementary Teachers? *Journal of Education* 181(2), 57–89 (1999)
5. Osborne, J., Simon, S., Collins, S.: Attitudes towards science: a review of the literature and its implications. *Int. Journal of Science Education* 25(9), 1049–1079 (2003)

* Corresponding author.

Learning by Teaching SimStudent*

Noboru Matsuda¹, Victoria Keiser¹, Rohan Raizada¹, Gabriel Stylianides²,
William W. Cohen¹, and Ken Koedinger¹

¹ School of Computer Science, Carnegie Mellon University,
5000 Forbes Ave. Pittsburgh PA 15213 USA

² School of Education, University of Pittsburgh,
5517 Posvar Hall, Pittsburgh PA 15260 USA

{noboru.matsuda, keiser, atu, wcohen, koedinger}@cs.cmu.edu,
rraizada@andrew.cmu.edu, gstylian@pitt.edu

The effect of tutor learning has been studied in various contexts, providing ample evidence to suggest that students learn when they teach others. Yet, the cognitive and social factors that facilitate or inhibit tutor learning are still not well understood. One factor that prohibited research progress in this area is that studying the tutor learning effect could often be done only at the cost of tutees' learning. To address this problem, we built an on-line learning environment where students learn by teaching a computer agent, called *SimStudent*, rather than their peers [1].

SimStudent is a lively computer agent that inductively learns skills through its own tutored-problem solving experience. SimStudent is integrated into an on-line learning environment where students can *interactively* tutor SimStudent in how to solve equations. In this learning environment, the student is asked to tutor SimStudent well enough so that SimStudent passes the built-in quiz prepared by the designer.

The tutoring interactions between students and SimStudent in the learning environment are designed to be much like the ones in human-to-human tutoring. Students pose problems for SimStudent to solve, provide feedback for the steps SimStudent performs, and provide a hint for any steps that SimStudent cannot perform correctly. To provide a hint, the student simply performs the step.

When students cannot provide a hint (which can happen often when *they* are learning equation solving as well!), they are encouraged to review examples that appear in the tutoring interface. A Curriculum Browser is also available that shows summary of the subject to tutor (e.g., algebra equation solving), skills involved (i.e., basic operations to solve equations), and worked-out examples.

Further information about SimStudent (including video clips and deliverables) can be found on our project web page at <http://www.SimStudent.org>

Reference

1. Matsuda, N., et al.: Learning by Teaching SimStudent: Technical Accomplishments and an Initial Use with Students. In: Kay, J., Aleven, V. (eds.) ITS 2010, Part II. LNCS, vol. 6095, p. 449. Springer, Heidelberg (2010)

* This study is supported by National Science Foundation Award No. DRL-0910176 and by Department of Education (IES) Award No. R305A090519. This work is also supported in part by the Pittsburgh Science of Learning Center, which is funded by the National Science Foundation Award No. SBE-0836012.

Authoring Problem-Solving ITS with ASTUS

Jean-François Lebeau, Luc Paquette, and André Mayers

Université de Sherbrooke, Québec, Canada

{jean-francois.lebeau2, andre.mayers}@usherbrooke.ca

<http://astus.usherbrooke.ca>

Step-based ITS have been proven successful for well-defined domains, particularly in well-defined tasks, but their success is mitigated by their cost. Typically, the main factor behind the cost is the efforts needed to model the task domain. Different approaches have been investigated to reduce these efforts: Model-Tracing Tutors (e.g. Cognitive Tutors, Andes), Constraint-Based Tutors (e.g. SQL-Tutor, ASPIRE) and Example-Tracing Tutors (e.g. CTAT's, ASSISTment).

With ASTUS, we aim to offer to the ITS community a support for the development of tutors for problem-solving tasks in a well-defined domains. In such context, building a framework based on a generative model of the task domain was deemed the most interesting approach, as it appeared as the one leading to a comprehensive and flexible solution. A solution which includes, for instance, not only the capacity to show next-step hints, but to generate them by instantiating domain-independent templates using data extracted from knowledge components. In other words, a modular framework (based on the classic four-module architecture) designed to facilitate the experimentation of different pedagogical approaches.

Using ASTUS's knowledge representation system, tutored skills (scripted dynamic plans) are encoded as glass-box components and the underlying ones (mental inferences and atomic actions) as black-box components. Thus a model consists of formatted definitions and executable code. Thanks to an authoring language (prototyped in Groovy) the model can be encoded in a single, coherent file. UI code is still needed to produce the learning environment (LE); debugging and visualization tools are available at runtime and as for IDE-based tools, we rely on Eclipse.

We are prototyping a linear algebra tutor with multiple LEs which follow a *form-building* design approach. Our hypothesis is that this approach offers benefits from both a form-filling (easier fine tracing) and a free-form (a more comprehensive assessment) one. As with a free-form approach, the goal sequence is not (entirely) reified in the LE; however the steps are (as in the form-filling approach) as a visual tool set. Thus, the learners build "forms" by manipulating both tools and data.

As the ITSs move from the labs to the classrooms, the next logical step may be to largely move the authoring efforts from highly specialized graduate students to domain experts (including teachers), but we are interested in investigating an intermediate step that consist in a comprehensive, flexible and usable framework for people with programming and knowledge-based systems skills. We are aware that our solution, based on generative models, may be justified only in well-defined domains and that some ill-defined tasks, such as design-based ones, may be challenging at best. However, there is no equivalent tool easily available for the ITS community.

A Better Reading Tutor That Listens

Jack Mostow, Greg Aist*, Juliet Bey, Wei Chen, Al Corbett, Weisi Duan, Nell Duke*, Minh Duong, Donna Gates, José P. González, Octavio Juarez, Martin Kantorzyk, Yuanpeng Li, Liu Liu, Margaret McKeown*, Christina Trotochaud*, Joe Valeri, Anders Weinstein, and David Yen

Project LISTEN**, School of Computer Science
Carnegie Mellon University, Pittsburgh, PA 15213, USA

mostow@cs.cmu.edu
www.cs.cmu.edu/~listen

Project LISTEN's Reading Tutor listens to children read aloud, and helps them learn to read, as illustrated on the Videos page of our website. This Interactive Event encompasses both this basic interaction and new extensions we are developing.

To accelerate fluency development, we are generating real-time visual feedback on children's oral reading expressiveness by mapping prosodic features such as timing, pitch, and intensity to graphical features such as position, shape, and color. To design more effective practice on individual words, we are conducting an experiment to investigate whether and how the amount of context in which the student practices a word – in a sentence, in a phrase, in a bigram, in isolation, or not at all – affects the time to read the word subsequently in connected text.

To accelerate vocabulary development, we are augmenting children's encounters of words in stories with additional instruction and encounters in multiple contexts required to acquire word meaning. To foster active processing required for successful learning, these encounters challenge the child to think about how words relate to context and to other words. We are developing automated methods to help generate effective contexts for learning word meaning, to generate useful challenges, to compute their answers, and to provide informative feedback to children's responses.

To teach explicit reading comprehension strategies, we are adapting expert human instruction into scripted scenarios for Reading Tutor dialogue. The strategies include activating background knowledge, visualizing, asking questions, and summarizing. We are working to automate the scripting process of generating comprehension instruction, for example by generating good questions about a story and scaffolding children to make up their own. As Chen, Mostow, and Aist's ITS2010 paper reports, we are attacking the problem of recognizing children's free-form spoken responses to tutor prompts by training them to respond more predictably, and by exploiting this predictability to improve speech recognition. This work aims to enable the Reading Tutor, and perhaps other tutors some day, to listen to children not just read but talk.

* Various other institutions.

** The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grants R305A080628, R305A080157, and R305B070458. The opinions expressed are those of the authors and do not necessarily represent the views of the Institute and the U.S. Department of Education. We thank the educators and students who help generate and analyze our data.

Research-Based Improvements in Cognitive Tutor Geometry

Steven Ritter, Brendon Towle, R. Charles Murray, Robert G.M. Hausmann,
and John Connelly

Carnegie Learning, Inc., 437 Grant Street, Pittsburgh, PA 15219
{sritter, btowle, cmurray, bhausmann}@carnegielearning.com,
jconnelly@carnegielearning.com

Carnegie Learning's Cognitive Tutors for mathematics have been the subject of a wide variety of research [3,4] and are the most widely deployed Intelligent Tutoring Systems. Currently, over 560,000 students and 2,700 schools in all 50 United States are using them. Many ITS researchers understand these tutors from printed works but have not had the opportunity to use the tutors in a hands-on fashion.

This exhibit focuses on a unit of instruction that has been substantially revised for the 2010/2011 school year: angles formed by a transversal. Some of the research leading to the design of this unit has focused on the relationship between finding angle measures and expressing the geometric reason for the angle measure [1]. More recently, research at the Pittsburgh Science of Learning Center has focused on the importance of representing student work within the diagram (rather than in a spatially-separated table), highlighting aspects of the diagram and more carefully articulating geometry reasoning within the context of the diagram [2].

This exhibit discusses the design solutions to these research issues. Participants will get an idea of the design and implementation considerations that go in to taking a research implementation to wide-scale use. Further information can be found at www.carnegielearning.com/ITS2010.

References

1. Aleven, V., Koedinger, K.R.: An effective meta-cognitive strategy: learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science* 26(2), 147–179 (2002)
2. Butcher, K., Aleven, V.: Diagram Interaction during Intelligent Tutoring in Geometry: Support for Knowledge Retention and Deep Transfer. B. C. Love, K. McRae, & V. M. Sloutsky (Eds.). In: 30th Annual Conference of the Cognitive Science Society, pp. 1736–1741. Cognitive Science Society, Austin (2008)
3. Ritter, S., Anderson, J.R., Koedinger, K.R., Corbett, A.: The Cognitive Tutor: Applied research in mathematics education. *Psychonomics Bulletin & Review* 14(2), 249–255 (2007)
4. Ritter, S., Kulikowich, J., Lei, P., McGuire, C.L., Morgan, P.: What evidence matters? A randomized field trial of Cognitive Tutor Algebra I. In: Hirashima, T., Hoppe, U., Young, S.S. (eds.) *Supporting Learning Flow through Integrative Technologies*, vol. 162, pp. 13–20. IOS Press, Amsterdam (2007)

A Cognitive Tutor for Geometric Proof

Steven Ritter, Brendon Towle, R. Charles Murray,
Robert G.M. Hausmann, and John Connelly

Carnegie Learning, Inc., 437 Grant Street, Pittsburgh, PA 15219
{sritter, btowle, cmurray, bhausmann}@carnegielearning.com,
jconnelly@carnegielearning.com

Geometric proof has long been a topic of study within Intelligent Tutoring Systems [3,4]. Proof is interesting because it supports a variety of solutions and strategies. However, implementation is challenging and such tutors have not been widely deployed.

In the 2010/2011 school year, Carnegie Learning will be introducing an Intelligent Tutoring System for geometric proof. Students create proof diagrams, much like in Angle [3], but they also translate the diagram into a two-column proof. Both forward and backward reasoning are supported [4].

The tutor was built using the Cognitive Tutor SDK [1] and deployed with the Tutor Runtime Environment [2]. This is a novel approach; the implementation exhibits all of the flexibility of a full production system but, at runtime, does not implement such a system. Participants will be given the opportunity to use this new tutor and discuss the design and implementation considerations that go in to taking a research implementation to wide-scale use. Further information can be found at www.carnegielearning.com/ITS2010.

References

1. Blessing, S.B., Gilbert, S.G., Oureda, S., Ritter, S.: Authoring model tracing cognitive tutors. *International Journal of AI in Education* 19 (2009)
2. Ritter, S., Blessing, S.B., Wheeler, L.: User modeling and problem-space representation in the tutor runtime engine. In: Brusilovsky, P., Corbett, A.T., de Rosis, F. (eds.) *User Modeling 2003*, pp. 333–336. Springer, Johnstown (2003)
3. Koedinger, K.R., Anderson, J.R.: Effective use of intelligent software in high school math classrooms. In: *Proceedings of the 1993 Conference on Artificial Intelligence in Education*, AACE, Charlottesville (1993)
4. Matsuda, N., VanLehn, K.: Advanced Geometry Tutor: An intelligent tutor that teaches proof-writing with construction. In: Looi, C.-K., McCalla, G., Bredeweg, B., Breuker, J. (eds.) *Proceedings of The 12th International Conference on Artificial Intelligence in Education*, pp. 443–450. IOS Press, Amsterdam (2005)

Multiplayer Language and Culture Training in ISLET

Kevin Saunders and W. Lewis Johnson

{ksaunders, ljohnson}@alelo.com
<http://alelo.com/products/islet/>

ISLET is a multiplayer role-playing game whose goal is to help players develop intercultural communication skills for French-speaking Sub-Saharan Africa. Its vision is to utilize multiplayer gameplay to create the most compelling language learning environment available. This vision is accomplished through three pillars:

1. Addictive advancement.
2. Engaging multiplayer conversations.
3. Compelling multiplayer quests.

The first pillar is a staple of role-playing games. The latter two are innovative elements unique to ISLET.

Addictive Advancement

ISLET has a sophisticated reward system that drives players to master language and the game. Massively multiplayer games such as *World of Warcraft* utilize advancement to keep customers returning even though the core gameplay doesn't change for hundreds of hours of play. Advancement systems are powerful motivators and support the goal of language learning, where repetition is important.

Engaging Multiplayer Conversations

ISLET is one of the first games to provide conversations with non-player characters in which multiple players can participate. These conversations are at the convergence of the two key aspects of the project: language learning and multiplayer gameplay. Unlike most role-playing games, these conversations are spoken.

Compelling Multiplayer Quests

While conversations provide immediate gameplay and advancement provides long-term goals, quests provide the short-term objectives and direction for the player. ISLET will emphasize quests that are uniquely designed for multiple players, creating a collaborative learning environment. This focus is in contrast to most games in which multiplayer quests are rare and typically merged single player quests.

In this interactive event, up to 5 attendees at a time, will play a short segment of ISLET. This segment will not be sufficient to result in language learning, but will serve as an introduction to ISLET's gameplay and pedagogy. Rudimentary knowledge of French will aid attendees, but is not necessary to participate in the event.

PSLC DataShop: A Data Analysis Service for the Learning Science Community

John Stamper¹, Ken Koedinger¹, Ryan S.J.d. Baker², Alida Skogsholm¹,
Brett Leber¹, Jim Rankin¹, and Sandy Demi¹

¹ Carnegie Mellon University, Human-Computer Interaction Institute

{jstamper, krk, alida, bleber, jimbokun, sdemi}@cs.cmu.edu

² Worcester Polytechnic Institute, Department of Social Science and Policy Studies

rsbaker@wpi.edu

The Pittsburgh Science of Learning Center's DataShop is an open data repository and set of associated visualization and analysis tools. DataShop has data from thousands of students deriving from interactions with on-line course materials and intelligent tutoring systems. The data is fine-grained, with student actions recorded roughly every 20 seconds, and it is longitudinal, spanning semester or yearlong courses. Currently over 188 datasets are stored including over 42 million student actions and over 150,000 student hours of data. Most student actions are "coded" meaning they are not only graded as correct or incorrect, but are categorized in terms of the hypothesized competencies or knowledge components needed to perform that action.

DataShop provides a number of features to facilitate data analysis including a data schema that allows researchers to import data into DataShop or export data from the repository in order to perform additional analysis. DataShop offers a number of online analysis tools to perform functions, such as visualizing student performance and analyzing learning curves. Researchers can export cognitive models, make changes, and upload the changed model for further analysis. One new feature that has been added to DataShop is an easy-to-use API for using web services to access the repository. These web services allow developers to identify data sets in the repository and directly export data from them at the transaction or student step level. In the near future, developers will be able to add new fields back into the repository with the use of our web services for custom fields.

Researchers have analyzed these data to better understand student cognitive and affective states and the results have been used to redesign instruction and demonstrably improve student learning [1]. Researchers can find out more and sign up for access to DataShop from our website: <http://pslcdatashop.org>

Reference

1. Koedinger, K.R., Baker, R.S.J.: A Data Repository for the EDM community: The PSLC DataShop. In: Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.J. (eds.) To appear in Handbook of Educational Data Mining. CRC Press, Boca Raton

A DIY Pressure Sensitive Chair for Intelligent Tutoring Systems

Andrew M. Olney and Sidney D'Mello

University of Memphis, Memphis TN 38152, USA
aolney@memphis.edu

This interactive event presents a pressure sensitive chair constructed out of Wii Fit game controller boards. During this event, we will demonstrate how to arrange the boards to detect seat and back pressure, configure a PC to receive a bluetooth datastream from the boards, and modify the power supply of the boards to increase uptime and reliability. We claim that the pressure sensitive chair so constructed is highly suitable for recovering posture information from a student interacting with an ITS.

Research in intelligent tutoring systems is paying increased attention to physiological measures for student modeling. Of particular interest is student affect and engagement which has been measured by multiple groups using posture sensors [12]. Unfortunately, commercially available posture sensors have traditionally been expensive limiting their widespread distribution with ITS.

However, as an alternative Nintendo's Wii Fit game controller can be repurposed as a functional, effective, and cost-efficient posture sensor. Using two Wii Fits, for posterior and back, one can obtain 8 pressure data streams (4 per Fit). Wii Fit boards communicate wirelessly with a PC using bluetooth. Though bluetooth association is a finicky process that is best demonstrated, once associated, the boards remain connected until either the PC is powered down or the boards run out of power. Thus it is ideal to leave the PC on continuously and modify the power supply to accept an AC adapter rather than the native batteries. Though the power requirements are unpublished, a 1000ma adapter soldered to the battery terminals provides good results. Once the boards have been associated, a free software library can be used to interface with the bluetooth stack and retrieve the information sent by the boards, e.g. WiimoteLib (<http://www.codeplex.com/WiimoteLib>). It is simple to log the data streams of a participant for offline analyses, or even process the streams in real time for posture reactive tutor. Provided with an existing chair, a complete system can be constructed for under 200 dollars in less than 1 hour, whereas commercially pressure sensors are available for \$10,000. The cost and robustness of the boards are such that scale up deployment is practical.

References

1. D'Mello, S., Picard, R.W., Graesser, A.: Toward an Affect-Sensitive AutoTutor. *IEEE Intelligent Systems* 22(4), 53–61 (2007)
2. Arroyo, I., Cooper, D.G., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion sensors go to school. In: *AIED 2009*, pp. 17–24 (2009)

Author Index

- Aagard, Hans II-281
Adam, Jean-Michel II-380
Agapito, Jenilyn II-263
Aghaei Pour, Payam I-264
Ai, Hua I-156, II-134
Aïmeur, Esma II-340
Aist, Gregory I-65, II-451
Albrechtsen, Justin S. II-144
Alcañiz, Mariano I-296
Aleven, Vincent I-115, I-174, I-413,
II-221, II-438
Allbritton, David I-204
Altman, Max II-346
AlZoubi, Omar I-264
Anglo, Elizabeth A. II-260
Anwar, Faisal II-209
Arnott-Hill, Elizabeth I-204
Arroyo, Ivon I-327, I-423
Ashish, Naveen II-352
Ashley, Kevin D. I-95
Auerbach, Daniel I-274
Azevedo, Roger I-369
- Baid, Palak II-209
Bain, Michael II-266
Bajanki, Sirisha II-443
Baker, Ryan S.J.d. I-25, II-263,
II-321, II-445, II-455
Barnes, Tiffany II-31, II-215, II-233,
II-239
Beck, Joseph E. I-35, I-194, II-254,
II-321, II-399, II-439
Beek, Wouter II-272, II-448
Ben-Naim, Dror II-266, II-440
Bernardini, Andrea I-125
Beuth, Jack L. I-156
Bey, Juliet II-451
Bieliková, Mária II-423
Biswas, Gautam II-405
Blanchard, Emmanuel G. II-269
Blessing, Stephen B. II-365
Bodnar, Stephen II-352
Boonthum, Chutima II-349
Bowen, Kyle II-281
- Boyer, Kristy Elizabeth I-55
Boyle, Roger II-443
Brandão, Leônidas O. II-447
Bredeweg, Bert II-272, II-448
Britt, Anne II-327
Broisin, Julien II-402
Brown, Christopher II-315
Brown, Jennifer II-178
Bühling, René II-272, II-448
Bull, Susan II-275
Burluson, Winslow I-184, I-327
Butcher, Kirsten R. II-278, II-414
- Cade, Whitney II-178
Cai, Zhiqiang II-327
Calvo, Rafael A. I-45, I-264
Capeli, Olimpio M. II-92
Castine, Melissa I-338, II-441
Cetintas, Suleyman I-15, II-281
Chae, Hui Soo II-209
Chaffar, Soumaya II-285
Chakravarty, Sugato II-281
Chalfoun, Pierre II-288
Champaign, John II-212
Chaouachi, Maher II-291
Chauncey, Amber I-369
Chen, Lin II-315
Chen, Wei I-65, II-451
Chi, Michelene T.H. I-401
Chi, Min I-224
Christopherson, Robert M. I-327
Cocea, Mihaela II-330
Cohen, Robin II-212
Cohen, William W. I-317, II-368, II-449
Combs, Rebekah I-245
Conati, Cristina I-125
Connelly, John II-446, II-452, II-453
Contero, Manuel I-296
Cooper, David G. I-327
Corbett, Al II-451
Core, Mark G. I-274
Courtemanche, François II-340
Cox, Richard II-224
Cristea, Alexandra I. II-82

- Crowley, Rebecca I-338, II-441
 Croy, Marvin II-31
 Curran, James R. II-303

 Daigle, Rosaire I-245
 Dailey, Matthew D. II-41
 Dalmon, Danilo L. II-447
 de Albuquerque, Antonio R.P.L. II-92
 de Freitas, Sara II-393
 Delgado Kloos, Carlos II-384
 Demi, Sandy II-455
 Dempsey, Kyle II-294
 Derbali, Lotfi II-297
 Dickison, Daniel II-300
 Dietrich, Michael II-420
 Di Eugenio, Barbara II-72, II-315
 Dimitriva, Vania II-443
 D'Mello, Sidney I-245, I-264, II-1, II-62,
 II-178, II-456
 Dolan, Robert I-327
 Dominguez, Anna Katrina II-303
 Dragon, Toby II-113, II-444
 Drummond, Joanna II-306
 Duan, Weisi II-451
 Duke, Nell II-451
 Dunwell, Ian II-393
 Duong, Minh II-451

 Eagle, Michael II-215
 Easterday, Matthew W. II-218
 English, Sara I-423

 Feenstra, Laurens II-221
 Feng, Mingyu II-309, II-312
 Finger, Susan II-387
 Floryan, Mark II-113, II-444
 Forbes-Riley, Kate I-379
 Forbes-Summers, Elijah I-194
 Forsyth, Carol II-327
 Fortin, Mikaël II-236
 Fossati, Davide II-315
 Foss, Jonathan G.K. II-82
 Fournier-Viger, Philippe II-318
 Frasson, Claude II-11, II-285, II-288,
 II-291, II-297, II-337

 Garcia Garcia, Grecia II-224
 Gates, Donna II-451
 Giguere, Stephen II-321
 Gilbert, Stephen II-365

 Girard, Sylvie I-307
 Gluga, Richard I-85, II-227
 Gobert, Janice D. II-257, II-445
 Gogvadze, George II-420
 Goldin, Ilya M. I-95
 Goldstein, Adam B. I-25
 Gómez Pérez, Asunción II-272, II-448
 Gong, Yue I-35, I-194
 González, José P. II-451
 Gounon, Patricia II-324
 Gracia del Río, Jorge II-272, II-448
 Graesser, Art I-245, II-327
 Gratch, Jonathan I-165
 Grzybicki, Dana I-338
 Gutierrez-Santos, Sergio I-105, II-330
 Gweon, Gahgene II-387

 Haciahmetoglu, Yonca II-334
 Haddawy, Peter I-75
 Ha, Eun Young I-55
 Halpern, Diane II-327
 Harris, Thomas K. II-300
 Hastings, Peter I-204
 Hausmann, Robert G.M. II-300, II-446,
 II-452, II-453
 Hays, Matthew J. I-274
 Hays, Patrick II-178
 Heffernan, Neil T. I-25, I-35, I-194,
 I-349, II-41, II-254, II-309, II-312,
 II-399, II-439
 Heraz, Alicia II-337
 Hirashima, Tsukasa II-343
 Hong, Yuan-Jin II-269
 Hord, Casey I-15
 Howley, Iris K. II-230
 Hussain, M. Sazzad I-264

 Ingram, Amy I-55
 Ishikawa, Masatoshi I-135
 Isotani, Naoko II-92
 Isotani, Sadao II-92
 Isotani, Seiji II-92, II-346, II-447

 Jackson, G. Tanner II-294, II-349
 Johan, Rasyidi II-275
 Johnson, Hilary I-307
 Johnson, Matthew II-233, II-275
 Johnson, W. Lewis I-165, II-352, II-454
 Jones, Christopher I-174
 Jordan, Pamela II-72

- Jraidi, Imène II-11
 Juarez, Octavio II-451
 Judd, Andrew II-21
 Jukic, Drazen I-338
 Jung, Sung-Young II-355

 Kabanza, Froduald II-51, II-251
 Kang, Jeon-Hyung II-359
 Kantorzyk, Martin II-451
 Kashihara, Akihiro I-389
 Kato, Yukari I-135
 Katz, Sandra II-72
 Kawai, Ryoya I-389
 Kayashima, Michiko II-362
 Kay, Judy I-85
 Kazi, Hameedullah I-75
 Keiser, Victoria I-317, II-449
 Kersey, Cynthia II-72
 Kim, Jihie II-188, II-359
 Kim, Julia I-174
 Klahr, David II-198, II-408
 Kodavali, Sateesh Kumar II-365
 Koedinger, Kenneth R. I-115, I-145,
 I-214, I-317, II-312,
 II-368, II-449, II-455
 Kopp, Kris II-327
 Kumar, Amruth N. I-359
 Kumar, Rohit I-156, II-134

 Lagud, Maria Carminda V. I-255
 Lajoie, Susanne P. II-242, II-269
 Lane, H. Chad I-274, II-144
 Lau, Lydia II-443
 Lebeau, Jean-François II-236, II-248,
 II-450
 Leber, Brett II-438, II-455
 Lee, Seung Y. II-155
 Lee-Shim, Kris II-275
 Legowski, Elizabeth I-338
 Lehman, Blair I-245, II-1
 Lepp, Dmitri II-396
 Leroux, Pascal II-324
 Lester, James C. I-55, I-285,
 II-155, II-166
 Lever, Tim I-85
 Liem, Jochem II-272, II-448
 Limongelli, Carla II-371
 Li, Nan II-368
 Linnebank, Floris II-272, II-448
 Lipschultz, Michael II-374

 Litman, Diane I-224, I-379,
 II-306, II-374, II-429
 Liu, Liu II-451
 Liu, Ming I-45
 Li, Yuanpeng II-451
 Lloyd, Tim II-275
 Loll, Frank II-377
 Looi, Chee-Kit I-1, II-426
 Luengo, Vanda II-380

 Mabbott, Andrew II-275
 Madran, Nadine II-380
 Magaro, Cressida II-198, II-408
 Magoulas, George I-105, II-330
 Ma, Jun II-188
 Marcus, Nadine II-266
 Marsella, Stacy I-2
 Martín-Gutiérrez, Jorge I-296
 Matsuda, Noboru I-317, II-449
 Maull, Keith II-245, II-278
 Mavrikis, Manolis I-105
 Mayers, André II-236, II-248, II-318,
 II-450
 McCuaig, Judi II-21
 McKeown, Margaret II-451
 McLaren, Bruce M. II-346, II-377,
 II-420
 McNamara, Danielle S. II-294, II-349
 McQuiggan, Scott I-285
 Medvedeva, Olga I-338
 Meissner, Christian A. II-144
 Melis, Erica II-420
 Mephu Nguifo, Engelbert II-318
 Meyer, Ann-Kristin II-420
 Michael, Stephen W. II-144
 Michelet, Sandra II-380
 Millis, Keith II-327
 Mitamura, Teruko I-214
 Mizoguchi, Riichiro II-92, II-362
 Moguel, Patrice II-123
 Montalvo, Orlando II-445
 Mostafavi, Behrooz II-239
 Mostow, Jack I-65, II-451
 Mott, Bradford W. II-155, II-166
 Mowery, Dana II-198, II-408
 Mpondo Eboa, Franck Hervé II-340
 Muldner, Kasia I-184, I-327
 Muñoz-Merino, Pedro J. II-384
 Muñoz-Organero, Mario II-384

- Murray, R. Charles II-300, II-446,
 II-452, II-453
 Murray, Tom II-113, II-444

 Nabos, Julieta II-263
 Nagasunder, Amrut II-134
 Naismith, Laura II-242, II-269
 Natriello, Gary II-209
 Neagle, Royce II-443
 Nguyen, Dong II-134, II-387
 Nixon, Tristan II-300
 Nkambou, Roger II-318

 Oberoi, Sharad V. II-387
 Ogan, Amy I-174
 Ohlsson, Stellan II-315
 Okoye, Ifeyinwa II-245, II-278
 Olney, Andrew M. II-178, II-390, II-456

 Panzoli, David II-393
 Paquette, Luc II-236, II-248, II-450
 Pardos, Zachary A. II-41
 Pavlik Jr., Philip II-103
 Pearlstein, Mike II-21
 Perkins, Lydia I-245
 Person, Natalie I-235, II-1
 Petridis, Panagiotis II-393
 Phillips, Robert I-55
 Pinkwart, Niels II-377
 Prank, Rein II-396

 Quek, Francis II-334
 Qureshi, Adam II-393

 Raab, Stephen I-338
 Rahati, Amin II-51, II-251
 Rai, Dovan II-254, II-399, II-439
 Raizada, Rohan I-317, II-449
 Ramandalahy, Triomphe II-402
 Rankin, Jim II-455
 Rau, Martina A. I-413
 Raziuddin, Juella J. II-257
 Razzaq, Leena I-349
 Rebolledo-Mendez, Genaro II-393
 Renkl, Alexander I-3
 Repalam, Ma. Concepcion II-263
 Reyes Jr., Salvador S. II-263
 Ritter, Steven I-4, II-300, II-446,
 II-452, II-453
 Robison, Jennifer I-285

 Rodrigo, Ma. Mercedes T. I-255, II-260,
 II-263
 Roll, Ido I-115
 Rosé, Carolyn Penstein I-156, II-134,
 II-230, II-387
 Rowe, Jonathan P. II-166
 Royer, James M. I-423
 Roy, Marguerite I-401
 Rummel, Nikol I-145, I-413, II-221
 Rus, Vasile I-45

 Sagae, Alicia II-352
 Salles, Paulo II-272
 Sao Pedro, Michael A. II-257, II-445
 Saunders, Kevin II-454
 Scheuer, Oliver II-377
 Schneider, Mike II-144
 Sciarrone, Filippo II-371
 Segedy, James II-405
 Sewall, Jonathan II-438
 Shaw, Erin II-188, II-359
 Shores, Lucy R. II-166
 Siler, Stephanie II-198, II-408
 Si, Luo I-15, II-281
 Skogsholm, Alida II-455
 Sottolare, Robert A. II-411
 Stamper, John II-31, II-455
 Stylianides, Gabriel I-317, II-449
 Suebnukarn, Siriwan I-75
 Sulcer, Brian II-405
 Sullins, Jeremiah I-245
 Sumner, Tamara II-245, II-278, II-414

 Taatgen, Niels II-221
 Tai, Minghui I-423
 Tchounikine, Pierre II-123
 Temperini, Marco II-371
 Thomas, James M. II-417
 Toth, Joe II-103
 Toto, Ermal II-445
 Towle, Brendon II-300, II-446,
 II-452, II-453
 Tricot, André II-123
 Trotochaud, Christina II-451
 Tseytlin, Eugene I-338, II-441
 Tsovaltzi, Dimitra II-420
 Tu, Arthur I-317
 Tvarožek, Jozef II-423

 Valeri, Joe II-451
 Van de Sande, Brett I-184

- VanLehn, Kurt I-184, I-224, II-355
Vaste, Giulia II-371
Velan, Gary II-266
Vidal, Philippe II-402
Vogt, Kimberly I-245
Volgas, Nick I-235
Vouk, Mladen I-55
- Wagner, Lynn I-338
Waki, Hiromi II-343
Walker, Erin I-145
Walker, Sean I-145
Wallace, Patty II-327
Wallis, Michael I-55
Wang, Ning I-165
Weinstein, Anders II-451
Wetzler, Philipp II-414
Williams, Betsy I-235
Williams, Claire I-235, II-62, II-178
Willows, Kevin II-198, II-408
- Wiseman, Jeffrey II-269
Wißner, Michael II-272, II-448
Woolf, Beverly Park I-5, I-327,
I-423, II-113, II-444
Wu, Longkai II-426
Wylie, Ruth I-214
- Xin, Yan Ping I-15
Xiong, Wenting II-429
- Yacef, Kalina II-303
Yamamoto, Sho II-343
Yen, David II-451
Yoo, Jungsoon II-432
Yoo, Sung II-432
Young, R. Michael II-417
Yuan, Brian I-235
- Zapata-Rivera, Diego II-435