Gautam Biswas
Susan Bull
Judy Kay
Antonija Mitrovic (Eds.)

LNAI 6738

Artificial Intelligence in Education

15th International Conference, AIED 2011
Auckland, New Zealand, June/July 2011

Springer

Lecture Notes in Artificial Intelligence 6738

Edited by R. Goebel, J. Siekmann, and W. Wahlster

Subseries of Lecture Notes in Computer Science

Gautam Biswas Susan Bull Judy Kay
Antonija Mitrovic (Eds.)

Artificial Intelligence in Education

15th International Conference, AIED 2011
Auckland, New Zealand, June 28 – July 1, 2011

Springer

Volume Editors

Gautam Biswas
Vanderbilt University, EECS Department, Nashville, TN 37325, USA
E-mail: gautam.biswas@vanderbilt.edu

Susan Bull
The University of Birmingham
Electronic, Electrical and Computer Engineering, UK
E-mail: s.bull@bham.ac.uk

Judy Kay
University of Sydney, School of Information Technologies, Australia
E-mail: judy.kay@sydney.edu.au

Antonija Mitrovic
University of Canterbury, College of Engineering
Department of Computer Science and Software Engineering, New Zealand
E-mail: tanja.mitrovic@canterbury.ac.nz

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-21868-2 e-ISBN 978-3-642-21869-9
DOI 10.1007/978-3-642-21869-9
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011929498

CR Subject Classification (1998): I.2, K.3, H.4, H.5

LNCS Sublibrary: SL 7 – Artificial Intelligence

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

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The 15th International Conference on Artificial Intelligence in Education (AIED 2011) was the next in a longstanding series of biennial international conferences for high-quality research in intelligent systems and cognitive science for educational computing applications. The conference provides opportunities for the cross-fertilization of approaches, techniques and ideas from the many areas that make up this interdisciplinary field, including: agent technologies, artificial intelligence, computer science, cognitive and learning sciences, education, educational technologies, game design, psychology, philosophy, sociology, anthropology, linguistics, and the many domain-specific applications for which AIED systems have been designed, deployed and evaluated.

To reflect the range of interests that combine advanced technology with advanced understanding of learners, learning, and the context of learning, the theme of AIED 2011 was "Next-Generation Learning Environments: Supporting Cognitive, Metacognitive, Social and Affective Aspects of Learning." This grew out of the key requirements identified by the editors of the previous AIED proceedings: Vania Dimitrova, Riichiro Mizoguchi, Benedict du Boulay and Art Graesser. As they pointed out, AIED involves "multidisciplinary research that links theory and technology from artificial intelligence, cognitive science and computer science with theory and practice from education and the social sciences."

The broad theme adopted for AIED 2011 was well-represented in the program, with contributions related to each of the issues. Furthermore, there was much overlap, with individual papers addressing two or more of these areas, and illustrating a variety of the more traditional artificial intelligence techniques as well as those developed to take advantage of growing twenty-first century technologies and related skills. AIED is both keeping up with and leading such developments. We anticipate further growth toward social and collaborative technologies in time for the next conference, as the more mature AIED research is increasingly harnessed to support new (and ever-changing) technologies and learning contexts in formal and informal settings.

The inherently interdisciplinary nature of the field made it very difficult to define specific categories into which to place papers in the Table of Contents for the conference proceedings. Most papers could have been logically categorized into several themes, based on the particular technological approaches they used, the type of system, the methods used in the research, and the teaching domain(s), etc. It is in the nature of our goals to address real problems in supporting learning, and so our work inevitably needs to bring together different stands of research. After much deliberation, rather than make what would to some extent be arbitrary choices, we decided to list papers in alphabetical order by author. We see this as a positive comment on the field of AIED: it is truly multidisciplinary not only in the areas covered in general, but also within specific research projects.

AIED 2011 received 193 submissions in the categories Full Paper, Poster, and Young Researcher Track (YRT), from 28 countries worldwide. Many of these were from North America and Europe, but the increase in submissions from Asia in recent years continued. Many submissions also came from Australia, New Zealand, and nearby places—the location of this conference perhaps playing a part in raising awareness of AIED in the region, and hopefully leading to increased research interest in the coming years.

The international Programme Committee (PC) and Senior Programme Committee (SPC) comprised members from 22 countries. Their areas of expertise matched well with the categories in which papers were submitted. This not only made it easier to assign reviewers, but also confirmed that the PC and SPC were representative of the current areas of interest in AIED.

Of the 153 Full Paper submissions, 49 (32%) were selected for oral presentation at the conference (8 proceedings pages). Some good submissions could not be accepted, as the cut-off was set very high. Posters offer high quality but perhaps less mature research, allowing for dissemination of newer developments and promising ideas (3 proceedings pages). The YRT offers PhD researchers the opportunity to present their research orally (3 proceedings pages), or in poster form, during the YRT session. The acceptance rate for oral YRT presentation was 39%. The aim is to encourage new researchers to discuss their work with other new researchers and swap experiences; and also to talk to more experienced members of the field to gain feedback on their ideas from the international AIED community. Individual mentoring is also available.

All papers, posters and YRT submissions were reviewed by at least three PC members, at least one of whom was a member of the SPC. There was then a discussion phase amongst the reviewers of each submission, where any inconsistencies were considered before a final meta-review was produced by a member of the SPC. Authors received each of the three or four original reviews, as well as the meta-review. We thank the PC and SPC for their diligence in reviewing and providing useful and constructive feedback, and for their willingness to engage in discussion about papers until a consensus (or conclusion) was reached. So many members of the committees did an outstanding job that it would be difficult to highlight particular individuals. We would like the SPC and PC to know that we received unsolicited and very positive comments from authors about the helpfulness of the reviews – not only in cases where papers were accepted, but also in many cases where they were not.

The conference also had three invited keynote speakers: Janet Metcalfe, speaking about metacognitively guided study in the region of proximal learning; Stellan Ohlsson on multiple mechanisms for deep learning; and John Sweller, discussing cognitive load theory and e-learning. These talks were highly relevant to some of the core AIED considerations, as well as being important in underpinning continuing developments and shifts in the field.

In addition to the above, AIED 2011 had an exciting Interactive Events Session, where participants could see demonstrations and try out AIED applications. Workshops allow detailed presentations and discussions focussed around specific themes, and a tutorial provided engaging interaction and discussion of advanced AIED research. Panel discussions provided insight, reflection, and multiple viewpoints (positive and negative) on the current state of the art, and promising directions for the future from some of the field's leaders.

This time the conference was held at the University of Auckland, New Zealand. The originally selected location was the University of Canterbury, New Zealand, but following the earthquake in Christchurch on February 22, 2011, which damaged much of the city's infrastructure, the local Organizing Committee worked hard to find a feasible and affordable alternative at short notice[1]. The University of Auckland very generously offered their space, and we thank them for this, as it was a major factor in helping to continue the conference on (almost) the originally planned dates, and fitting the allocated budget. We also thank Moffat Matthews, from the University of Canterbury, for visiting venues and sorting out many of the unexpected problems as swiftly as was possible under these circumstances.

The Organizing Committee was invaluable in helping to put together a good program, to seek sponsorship, and to publicize the conference. H. Chad Lane and Brent Martin were extremely energetic in bringing together the Interactive Events; Pramuditha Suraweera proficiently oversaw the YRT process, helping newer researchers to understand the purpose of the YRT, as well as answering all their questions; Riichiro Mizoguchi and Bert Bredeweg sought an exciting tutorial – relevant to the many quickly developing directions of the field, while at the same time being sufficiently mature for a tutorial; Cristina Conati and Isabel Fernandez de Castro worked incredibly hard on obtaining workshop proposals, and on organizing the whole workshop process; Tak-Wai Chan and Rafael Morales took over liaison with the local organizers once the final numbers for poster presentations were known, and communicated with the authors about poster requirements; and Jim Greer and Monique Grandbastien chased lively, eloquent people for panel discussions. General publicity for the conference was ably handled by Peter Brusilovsky and Rose Luckin, with Moffat Matthews providing an excellent website and other online support for the conference. Lewis Johnson and Chee-Kit Looi tracked down sponsorship in a global economically difficult time. In addition to those already mentioned, we had help from a few "local" people: James R. Segedy (Vanderbilt), Matthew D. Johnson (Birmingham), and student volunteers at the conference. We also benefitted from previous experiences in various aspects of conference organization offered by Vincent Aleven, Art Graesser and Jack Mostow. To all these people we offer our sincere thanks.

[1] We also express our sympathies for the victims of the terrible earthquake and tsunami in northern Japan. While it is reassuring to learn that most of the researchers from our community are safe, we do extend our sincere support to our colleagues and others who are still recovering from the devastating tragedy.

Finally, we would also like to thank the authors. Of course, we acknowledge their exciting research contributions and are delighted that they chose AIED 2011 as the conference at which to present their work. But this year they also had to deal with uncertainty about the conference location, late information about registration costs because of the necessary re-budgeting, and other associated difficulties. We were impressed by the way in which people took this in their stride, and waited so patiently for decisions to be reached. The AIED community has clearly demonstrated that it is an affable, understanding community.

Despite the unanticipated difficulties, we very much enjoyed putting together this conference. Being scattered around the world meant that at crucial times there was always at least one person awake somewhere with AIED 2011 on their mind. There was also always at least one person ready to take over, to allow us to sleep.

We enjoyed being in the same time zone in what turned out to be a stimulating conference.

Gautam Biswas
Susan Bull
Judy Kay
Antonija Mitrovic

Organization

International Artificial Intelligence in Education Society Management Board

Judy Kay, University of Sydney, Australia - President (2009-2011)
Jack Mostow, Carnegie Mellon University, USA - President Elect
Art Graesser, University of Memphis, USA - Secretary / Treasurer
James Lester, University of North Carolina, USA - Journal Editor

Advisory Board

Claude Frasson	University of Montreal, Canada
Monique Grandbastien	Université Henri Poincaré, France
Jim Greer	University of Saskatchewan, Canada
Lewis Johnson	University of Southern California, USA
Alan Lesgold	University of Pittsburgh, USA

Executive Committee Members

Vincent Aleven	Carnegie Mellon University, USA
Joseph E. Beck	Worcester Polytechnic Institute, USA
Ben du Boulay	University of Sussex, UK
Jacqueline Bourdeau	Télé-Université du Quebec, Canada
Susan Bull	University of Birmingham, UK
TakWai Chan	National Central University, Taiwan
Cristina Conati	University of British Columbia, Canada
Ricardo Conejo	Universidad de Málaga, Spain
Vania Dimitrova	University of Leeds, UK
Ulrich Hoppe	University of Duisburg, Germany
Susanne Lajoie	McGill University, Canada
Rosemary Luckin	University of London, UK
Riichiro Mizoguchi	Osaka University, Japan
Albert Corbett	Carnegie Mellon University, USA
H. Chad Lane	University of Southern California, USA
Chee-Kit Looi	Nanyang Technological University, Singapore
Antonija Mitrovic	University of Canterbury, New Zealand
Jack Mostow	Carnegie Mellon University, USA
Helen Pain	University of Edinburgh, UK
Julita Vassileva	University of Saskatchewan, Canada
Beverly Woolf	University of Massachusetts, USA

Organizing Committee

General Chair

Judy Kay	University of Sydney, Australia

Local Arrangements Chair

Antonija Mitrovic	University of Canterbury, New Zealand

Program Chairs

Susan Bull	University of Birmingham, UK
Gautam Biswas	Vanderbilt University, USA

Interactive Events Chairs

H. Chad Lane	University of Southern California, USA
Brent Martin	University of Canterbury, New Zealand

Young Researcher's Track Chair

Pramuditha Suraweera	University of Canterbury, New Zealand

Tutorial Chairs

Riichiro Mizoguchi	Osaka University, Japan
Bert Bredeweg	University of Amsterdam, The Netherlands

Workshop Chairs

Cristina Conati	University of British Columbia, Canada
Isabel Fernandez de Castro	The University of the Basque Country, Spain

Poster Chairs

Tak-Wai Chan	National Central University, Taiwan
Rafael Morales	Universidad de Guadalajara, Mexico

Panel Chairs

Jim Greer	University of Saskatchewan, Canada
Monique Grandbastien	Université de Nancy, France

Sponsorship Chairs

Lewis Johnson	University of Southern California, USA
Chee-Kit Looi	Nanyang Technological University, Singapore

Publicity Chairs

Peter Brusilovsky	University of Pittsburgh, USA
Rose Luckin	University of London, UK
Moffat Mathews	University of Canterbury, New Zealand

Program Committee

Senior Program Committee

Akihiro Kashihara	University of Electro-Communications, Japan
Akira Takeuchi	Kyushu Institute of Technology, Japan
Albert Corbett	Carnegie Mellon University, USA
Ana Paiva	Technical University of Lisbon, Portugal
Art Graesser	University of Memphis, USA
Ben du Boulay	University of Sussex, UK
Bert Bredeweg	University of Amsterdam, The Netherlands
Beverly Woolf	University of Massachusetts Amherst, USA
Carolyn Rose	Carnegie Mellon University, USA
Chee-Kit Looi	Nanyang Technological University, Singapore
Cristina Conati	University of British Columbia, Canada
Diane Litman	University of Pittsburgh, USA
Erica Melis	German Research Institute for Artificial Intelligence (DFKI), Germany
Felisa Verdejo	National Distance Learning University (UNED), Spain
Gerhard Weber	University of Vienna, Austria
Gordon McCalla	University of Saskatchewan, Canada
H. Chad Lane	University of Southern California, USA
Helen Pain	University of Edinburgh, UK
Ido Roll	University of British Columbia, Canada
Ivon Arroyo	University of Massachusetts Amherst, USA
Jack Mostow	Carnegie Mellon University, USA
Jacqueline Bourdeau	Télé-université, UQAM, Canada
James Lester	North Carolina State University, USA
Jim Greer	University of Saskatchewan, Canada
John Stamper	Carnegie Mellon University, USA
Julita Vassileva	University of Saskatchewan, Canada
Kalina Yacef	University of Sydney, Australia
Ken Koedinger	Carnegie Mellon University, USA
Kevin Ashley	University of Pittsburgh, USA
Neil Heffernan	Worcester Polytechnic Institute, USA
Niels Pinkwart	Clausthal University of Technology, Germany
Peter Brusilovsky	University of Pittsburgh, USA
Peter Sloep	Open Universiteit, The Netherlands

Pierre Tchounikine	University of Grenoble, France
Richiiro Mizoguchi	Osaka University, Japan
Roger Azevedo	McGill University, Canada
Rose Luckin	London Knowledge Lab, UK
Ryan Baker	Worcester Polytechnic Institute, USA
Stellan Ohlsson	University of Illinois at Chicago, USA
Steven Ritter	Carnegie Learning, USA
Susanne Lajoie	McGill University, Canada
Sydney d'Mello	University of Memphis, USA
Tak-Wai Chan	National Central University of Taiwan, Taiwan
Tsukasa Hirashima	Hiroshima University, Japan
Ulrich Hoppe	University of Duisburg-Essen, Germany
Vania Dimitrova	University of Leeds, UK
Vincent Aleven	Carnegie Mellon University, USA
Wouter van Joolingen	Universiteit Twente, The Netherlands

Program Committee

Adam Giemza
Adam Nkama
Agneta Gulz
Aisha Walker
Alessandro Micarelli
Alexandra Cristea
Amali Weerasinghe
Amy Baylor
Andre Tricot
Andreas Schmidt
Andrew Olney
Andrew Ravenscroft
Ari Bader-Natal
Ashok Goel
Aude Dufresne
Barbara Di Eugenio
Ben Chang
Brent Martin
Bruce McLaren
Carlo Tasso
Chih-Kai Chang
Chris Quintana
Daniela Romano
Darina Dicheva
David Jonassen
Demetrios Sampson
Diego Zapata
Elena Gaudioso
Fabio Akhras
G. Tanner Jackson
Isabel Fernandez-Castro
Jihie Kim
Kazuhisa Miwa
Kiyoshi Nakabayashi
Krittaya Leelawong
Maiga Chang
Marcelo Milrad
Matthew Easterday
Mille Alain
Monique Grandbastien
Nicolas Van Labeke
Noboru Matsuda
Olga Santos
Paul Brna
Pentti Hietala
Peter Dolog
Peter Reimann
Philip Pavlik
Pramudi Suraweera
Pratim Sengupta
Roger Nkambou
Rosa Vicari
Russell Johnson
Scotty Craig
Sridhar Iyer
Stephen B. Blessing

Susan Haller
Tiffany Barnes
Ton De-Jong
Toshio Okamoto
Winslow Burleson
Yam San Chee
Yong Se Kim

Additional Reviewers

Ainhoa Alvarez
Amanda Carr
Amber Strain
Andrea Nickel
Awiad Hossian
Blair Lehman
Brian Sulcer
Carla Limongelli
Christine Steiner
Christo Dichev
Christopher Brooks
Claudio Biancalana
Craig Stewart
David Joyner
Drew Hicks
Elaine Stampfer
Elder R. Santos
Elder R. Santos
Emily Ching
Evelyn Yarzebinski
Felice Ferrara
Filippo Sciarrone
Hitomi Saito
I-Han Hsiao
Ilya Goldin
Iolanda Leite
Isabel Alexandre
James R. Segedy
Jeonhyung Kang
John Kinnebrew
Josh Underwood
Juan-Diego Zapata-Rivera
Julia Svoboda
Junya Morita
Karim Sehaba
Katherine Forbes-Riley
Kazuaki Kojima
Keith Shubeck
Kristy Boyer
Kyle Cheney
Leena Razzaq
Lin Chen
Longhi Rossi
Luiz Henrique
Maite Martin
Mark Core
Martin van Velsen
Martina Rau
Michael Bett
Michael Yudelson
Min Chi
Nathalie Guin
Norio Ishii
Roberto Martinez
Rui Figueiredo
Scott Bateman
Shaghayegh Sahebi
Stefan Weinbrenner
Stéphanie Jean-Daubias
Sujith Gowda
Tilman Goehnert
Wilma Clark
Yen-Cheng Yeh
Yugo Hayashi
Yu-Han Chang
Yusuke Hayashi
Zach Pardos

Sponsors

Computer Science & Software Engineering Department University of Canterbury, NZ

College of Engineering, University of Canterbury, NZ

Asia Pacific Society for Computers in Education

The University of Auckland, NZ

The International Artificial Intelligence in Education Society

Institute for Intelligent Systems, The University of Memphis

Table of Contents

Invited Talks

Full Papers

Posters

Young Researcher's Track

Interactive Events

Workshops

Metacognitively Guided Study in the Region of Proximal Learning

Janet Metcalfe

Columbia University
jm348@columbia.edu

Abstract. Empirical data on people's metacognitively guided study time allocation-- data that resulted in the proposal that metacognitively astute people attempt to study in their own Region of Proximal Learning (RPL)-- will be reviewed. First, the most straightforward study-choice strategy that metacognitively sophisticated learners can use is to decline to study items that they know they already know. If an item has already been mastered, then further study is unnecessary. All theories, including the RPL model, agree on this strategy, and many, but not all, people use it. Its effective use depends on refined metaknowledge concerning the boundary between what is known and what is not known, as well as the implementation of a rule to decline study of items for which judgments of learning are very high. There are many situations in which people are overconfident, and if they are, they may miss studying items that are almost, but not quite, mastered. These items would yield excellent learning results with just a small amount of study, and so this failure to study almost-learned items has detrimental results. Data will be presented showing that young middle childhood children (7 to 9 year olds) tend not only to be overconfident—thinking they know things when they do not—but also to have an implementation deficit in using metacognitively-based item-choice strategies. One result is that many children at this age fail to use even this most obvious study strategy, even though it will be shown that it would benefit their learning. When the computer implements this learning strategy for the children their later performance improves. Second, with already-learned item eliminated, metacognitively sophisticated learners selectively study the items that are closest to being learned first, before turning to more distal items that will require more time and effort. This, as well as studying the materials that are within their cognitive reach, rather than items that are too difficult, is a strategy that conforms to the so-called "Goldilocks principle"—not too easy and not too difficult but just right. As will be detailed, while college-aged learners use this strategy, older middle childhood children (aged 9-11) do not. Children at this age are not without strategies, however. They do use the strategy of declining the easiest items (including the already-learned items). However, the older middle childhood children overgeneralize this strategy to selectively prefer the most difficult items. While their learning is negatively impacted by this, it is improved if the computer implements the Goldilocks principle on their behalf. Third, people use a stop rule that depends upon a dynamic metacognitive assessment of their own rate of learning. They discontinue study when they perceive that continued efforts are yielding little learning return. This stop rule predicts that people will stop studying easy items when they are fully learned

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 1–2, 2011.

them (and the learning rate has reached an asymptote on ceiling). Study will also stop, however, if the item is too difficult to allow noticeable learning. This strategy keeps people from being trapped in laboring on very difficult items in vain. Finally, the value that each item is assigned on a criterion test, if known during study, influences which items metacognitively sophisticated people choose to study and for how long they continue to study them. Items worth many points on a test will be studied sooner, longer and more often, than items worth few points. But not all learners use these strategies to their advantage. To effectively use the strategies that the Region of Proximal Learning framework indicates are effective, the learners must both have adequate metacognitive knowledge and also exhibit good implementation skills. Both metaknowledge and implementation skills vary across people. Age differences, motivational style differences, and metacognitive expertise differences can result in strategies that vary considerably, and which can result in sizable differences in the effectiveness with which the individual is able obtain his or her learning goals.

Bio: Dr. Metcalfe is a full professor in the Department of Psychology at Columbia University. She has worked in many areas of cognitive and metacognitive research and intervention, and has experience with a broad range of research problems, populations, and settings. She is the editor of three books related to various aspects of metacognition—Metacognition: Knowing about Knowing; The Missing Link in Cognition: Origins of Self-Reflective Consciousness, and Metacognition of Agency and Joint Attention (forthcoming). She has also authored, with John Dunlosky, the first textbook on metacognitive processes: Metacognition. Her metacognitive research has been directed both at college-aged students and at school-aged children. She has received numerous awards from such agencies as NIMH (National Institute of Mental Health), NSERC, the National Science and Engineering Research Council of Canada, the Institute for Educational Science, in the Department of Education, and The James S. McDonnell Foundation, to investigate and develop computational models of human memory and metamemory, to study metacognition and control processes, to examine the mechanisms underlying human memory, and to seek ways to enhance human learning and memory. Her recent work has focused on theories of and methods to improve learning and to overcome errors. She has done breakthrough work on the hypercorrection paradigm, in which high confidence errors are shown to be more easily updated than low confidence errors. She proposed and developed the Hot/Cool theory of delay of gratification. She has extensively researched people's metacognition concerning their own agency. She has published seminal papers on metacognition and control processes, developing the Region of Proximal Learning model of effective metacognitively guided study time allocation.

Multiple Mechanisms for Deep Learning: Overcoming Diminishing Returns in Instructional Systems

Stellan Ohlsson

University of Illinois at Chicago, Department of Psychology, Chicago, IL 60607
stellan@uic.edu

Abstract. The design of instructional materials in general and intelligent tutoring systems in particular should be guided by what is known about learning. The purpose of an instructional system is, after all, to supply the cognitive mechanisms in the learner’s mind with the information they need to create new knowledge. It is therefore imperative that the design of instruction is based on explicit models of those mechanisms. From this point of view, research has to date been characterized by two conceptual limitations. The first limitation is that systems are designed to teach to a narrow set of learning mechanisms, sometimes even a single one. There are signs that attempts to build intelligent tutoring systems that address a single mode of learning encounter diminishing returns, in terms of student improvement, with respect to implementation effort. The reason is that people learn in multiple ways. In this talk, I argue that there are approximately nine distinct modes of learning cognitive skills. To be maximally effective, instruction should support all nine modes of learning. This is the way to overcome the diminishing returns of tutoring systems with a narrow bandwidth. The second limitation is the traditional focus in both the science of learning and the practice of instruction on additive or monotonic learning: That is, learning in which the student extends his/her knowledge base without reformulating the knowledge he/she possessed at the outset. Additive extensions of a person’s knowledge are certainly real and important, but they do not exhaust the types of learning of which human beings are capable. In many learning scenarios, the learner must overcome or override the implications of prior knowledge in order to learn successfully. This requires cognitive mechanisms that transform or reject the prior knowledge, in addition to building new knowledge. In this talk, I provide an outline of the essential characteristics of such non-monotonic learning processes. I end the talk by spelling out some implications of the multiple-mechanisms and non-monotonicity principles for the future development of instructional systems.

Bio: Stellan Ohlsson is Professor of Psychology and Adjunct Professor of Computer Science at the University of Illinois at Chicago (UIC). He received his Ph.D. in psychology at the University of Stockholm in 1980. He joined the Learning Research and Development Center (LRDC) in Pittsburgh in 1985 and was promoted to Senior Scientist in 1990. He moved to his present position at UIC in 1996. Dr. Ohlsson has published extensively on computational models of cognitive change, including creative insight, cognitive skill acquisition and conceptual change. He invented the

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 3–4, 2011.

concept of Constraint-Based Modeling (CBM), one of the cornerstones of research on intelligent tutoring systems. He has held grants from the Office of Naval Research (ONR) and the National Science Foundation (NSF), among other agencies. He is one of the co-originators of the AIED conference series, and he co-chaired the 1987 and 1993 conferences. He has been a member of the editorial board of the International Journal for Artificial Intelligence in Education and other cognitive journals. In 2010, Dr. Ohlsson co-chaired the 32nd Annual Meeting of the Cognitive Science Society. Dr. Ohlsson recently completed *Deep Learning: How the Mind Overrides Experience*, a synthesis of his research, published by Cambridge University Press.

Cognitive Load Theory and E-Learning

John Sweller

School of Education, University of New South Wales, Sydney, NSW 2052
j.sweller@unsw.edu.au

Abstract. Cognitive load theory (Sweller, Ayres, & Kalyuga, 2011) is an instructional theory based on some aspects of human cognition. It takes an evolutionary approach to cognition. The theory assumes two categories of knowledge: biologically primary and biologically secondary knowledge. Primary knowledge is knowledge we have evolved to acquire over many generations. Secondary knowledge is cultural knowledge that humans have required more recently and have not specifically evolved to acquire. Cognitive load theory applies to secondary rather than primary knowledge. With respect to secondary knowledge, the theory assumes that human cognition constitutes a natural information processing system that has evolved to mimic another natural information processing system, biological evolution, with both systems characterised by the same basic principles. These principles lead directly to the assumption that biologically secondary knowledge consists of a very large range of domain-specific knowledge structures and that the primary aim of instruction is to assist learners in the acquisition of that knowledge. There are two basic structures associated with human cognitive architecture that are critical to instructional design – working memory and long-term memory.

Cognitive load theory assumes a limited working memory used to process novel information and a large, long-term memory used to store knowledge that has been acquired for subsequent use. The purpose of instruction is to store information in long-term memory. That information consists of everything that has been learned, from isolated, rote-learned facts to complex, fully understood concepts and procedures. Learning is defined as a positive change in long-term memory. If nothing has changed in long-term memory, nothing has been learned.

The theory has been used to generate a wide range of instructional procedures. Each of the procedures is designed to reduce extraneous working memory load in order to facilitate the acquisition of knowledge in long-term memory. One such procedure is based on the transient information effect, an effect that is closely associated with the use of instructional technology to present information.

When technology is used to present information to learners, the modality and format of the presentation is frequently changed. For example, written information may be substituted by spoken information and the static graphics associated with hard copy may be replaced by animations. While instructional designers are usually highly cognizant of these changes, there is another, concomitant but less obvious change that occurs. Relatively transient forms of information such as speech or animations replace a relatively permanent form of information such as written text or visual graphics. Frequently, this change is

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 5–6, 2011.

treated as being incidental and is ignored. Cognitive load theory suggests that it may be critical. Limited human working memory results in transient, technology-based information having considerable instructional consequences, many of them negative. Theory and data associated with the transient information effect will be discussed in relation to e-learning.

Bio: John Sweller is an Emeritus Professor of Education at the University of New South Wales. His research is associated with cognitive load theory. The theory is a contributor to both research and debate on issues associated with human cognition, its links to evolution by natural selection, and the instructional design consequences that follow.

Reference

1. Sweller, J., Ayres, P., Kalyuga, S.: Cognitive load theory. Springer, New York (2011)

Social Communication between Virtual Characters and Children with Autism

Alyssa Alcorn[1], Helen Pain[2], Gnanathusharan Rajendran[3], Tim Smith[4],
Oliver Lemon[1], Kaska Porayska-Pomsta[5], Mary Ellen Foster[1],
Katerina Avramides[5], Christopher Frauenberger[6], and Sara Bernardini[5]

[1] Heriot Watt University
[2] University of Edinburgh
[3] Strathclyde University
[4] Birkbeck College
[5] London Knowledge Lab
[6] University of Sussex
A.Alcorn@hw.ac.uk
http://echoes2.org

Abstract. Children with ASD have difficulty with social communication, particularly joint attention. Interaction in a virtual environment (VE) may be a means for both understanding these difficulties and addressing them. It is first necessary to discover how this population interacts with virtual characters, and whether they can follow joint attention cues in a VE. This paper describes a study in which 32 children with ASD used the ECHOES VE to assist a virtual character in selecting objects by following the character's gaze and/or pointing. Both accuracy and reaction time data suggest that children were able to successfully complete the task, and qualitative data further suggests that most children perceived the character as an intentional being with relevant, mutually directed behaviour.

Keywords: autism spectrum disorder, virtual environment, virtual character, joint attention, social communication, technology-enhanced learning, HCI.

1 Introduction

The autism spectrum encompasses a group of pervasive developmental disorders characterised by notable difficulties in communication and social interaction, plus the presence of repetitive behaviours and interests [1]. Virtual environments and characters are a promising method for supporting social communication in children on the autism spectrum due to the potential for skills to be practiced repeatedly, in a way that may be less threatening, less socially demanding and more controllable than a face-to-face interaction with a human partner [2, 3]. There is potential for supporting skill generalisation by changing the virtual setting of tasks, or introducing multiple characters. To date, the use of VEs in interventions for those with ASD has been often narrowly focused on specific social situations, such as adolescents navigating through a cafe [4], rather than on supporting foundation skills like joint attention. Also, few have targeted young children (though see [5] for an exception).

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 7–14, 2011.

Many general questions about the abilities of young children with ASD to interact with VEs and virtual characters remain unanswered. The ECHOES Technology Enhanced Learning (TEL) project is developing an intelligent multi-modal VE for supporting and scaffolding social communication in children aged 5-8 years, with and without an ASD. It comprises a range of touch screen-based learning activities focused on joint attention initiation and response. AI software modules direct an autonomous virtual character and are capable of intelligent tutorial planning [6]. Learning is embodied [7], with the child an active participant and collaborator, creating an emergent narrative along with a child-like virtual character [8].

The current empirical study used a simplified version of ECHOES as a research tool, with children completing a single session of an object-selection task in which a virtual character, Paul, varied his strategies for initiating and directing joint attention. The character's behaviours were hard-coded rather than generated by the AI planner, in order to attain the necessary control over the interaction. Qualitative data yielded insight into more general questions about the children's interaction with the system. The current study also provides crucial formative evaluation of ECHOES with the target user group [9] which fed back to the design of the full system. It was not intended as an intervention in its own right, but as a means to explore how joint attention and gaze-following skills might be elicited and supported in a VE.

2 Background and Project Objectives

2.1 The Autism Spectrum and Joint Attention

Joint attention is a key skill targeted by many intervention programmes for ASD, as its improvement seems to lead to lasting benefit in many areas, including language [10]. Joint attention is defined as the triadic coordination of attention between two persons and an object, and requires the ability to follow and direct another person's focus of attention [11]. A response involves following the initiator's gaze direction or gesture to a location in space and, perhaps, acting accordingly. Typically developing (TD) individuals frequently initiate joint attention for the purpose of *social sharing*, finding the reciprocal interest and affect strongly motivating. Closely related is *social referencing*, an attentional initiation in which an infant or young child looks towards a parent for information when faced with a novel event or object.

Attentional initiations through gaze and pointing are inherently ambiguous, and the two social partners must share a context in order for the respondent to understand the motivation for that initiation and infer the appropriate response (if any). Without joint attention, two or more persons have difficulty in establishing a shared focus of activity or communication. Individuals with ASD often show extreme difficulty with the gaze following and social inference necessary for successful joint attention, attaining proficiency with a great deal of effort, if at all.

2.2 Virtual Environments for Intervention and Social Scaffolding

Very little is currently known about how young children with ASD interact with VEs, or how they perceive and interact with virtual characters. Previous research is unclear about whether they might respond to joint attention initiations in such a context, and about which *specific* behaviours might be effective for directing and eliciting responses to attention. Before we can consider developing an intervention, we must first test the assumptions and prerequisites about the interaction between the child and the virtual character. A social-skills-focused programme supported by a virtual character crucially depends on the user's perception of the character as an *intentional being*, with agency, intentions and desires. He must not be perceived as an inanimate object nor as a cartoon to be passively watched, but as intending to communicate with the child and behaving in mutually relevant ways [12]. The literature suggests that *mutual gaze*, or gazing at the joint attention respondent before gazing at an object, may be a crucial method for establishing *mutuality* between the initiator and respondent [13]. These findings lead to our assumption that interactions which begin with the virtual character establishing this mutuality should increase a user's impressions of his intentionality.

2.3 The Present Study

The goals of this study were a) to investigate how young children with ASD interact with the ECHOES environment, and b) to analyse which combination of the character's mutual gaze and pointing gestures were most successful for eliciting the gaze-following behaviour necessary to complete the joint attention task.

Observational and video data collected in the course of the study illuminate the more general, exploratory questions regarding how children with ASD interact with the interface and the virtual character, and whether they perceive him as intentional and mutually-directed. Given the lack of existent research in this area, no specific predictions were made about these questions. The study results will inform the design of the full ECHOES system by highlighting which character behaviours are effective at directing attention, lead to perceived intentionality, and are fun and engaging for young users with ASD.

In relation to b), this population was predicted to exhibit at least *some* gaze-following, even if infrequent or inconsistent, with mutual gaze (engagement) conditions predicted to produce gaze-following to an object in the environment more often and more rapidly than in non-engagement conditions. The character's pointing cues were predicted to increase accuracy and decrease reaction times on all trials. These two behaviours (mutual gaze and pointing) could potentially interact; more rapid, more accurate, or more frequent gaze following may require both.

3 Methodology

3.1 Design of the Joint Attention Task

Users' gaze-following behaviours were measured during a simple selection task. Each trial involved three flowers[1] (two distractors and a target) to which Paul tried to direct the child's attention. A virtual character can initiate attention with the child and the object in the same way as would a human partner, by first looking into their partner's eyes (*mutual gaze*). Paul looking out from the screen gives an illusion of looking "at"

[1] One flower of each colour (red, yellow, blue) was presented per trial, with colour and screen position of the target counterbalanced across trials.

the viewer, similar to [13]. Varying the character's mutual gaze created 2 levels of the gaze following task: *engagement* and *non-engagement*. In the former, Paul established "mutual gaze", whereas in the latter condition he never gazed at the child, only directly to the object. Paul also varied his use of pointing, creating two levels within each gaze condition for a total of four trial types (mutual gaze + point, non-mutual gaze + point, mutual gaze + no point and non-mutual gaze + no point). Pointing shares several important features with gaze: both actions direct the respondent to a location in space, but are ambiguous and take their meaning from context [14]. Pointing also provides a visual cue in the form of directed motion, with greater potential for capturing the child's attention than gaze alone (see Figure 1, right).

Each participant was assigned a uniquely ordered trial script of 36 possible trials divided into three blocks of 12. Trials were randomly ordered and counterbalanced within each block; a child completing only 12 or 24 trials would see the same number of trials from each of the four types.

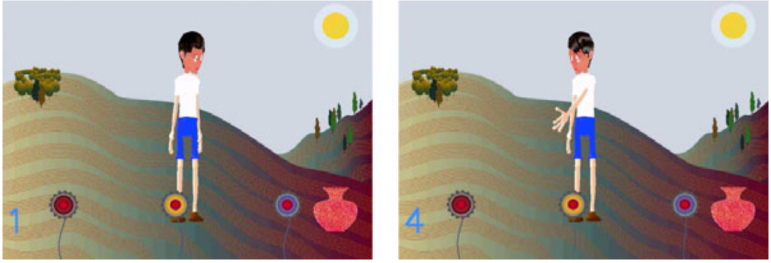


Fig. 1. Paul uses gaze only (left) and gaze plus gesture (right) to indicate his target flower

3.2 Participants and Procedure

Prior to the study, the study design and virtual environment were tested by 4 TD children aged 4-7 years (1 male, 3 female). This resulted in a number of changes to the environment, including adding a trial counter, adding background garden sounds and adjusting the timing between actions. Additionally, testing identified the need for prompting and support (e.g. reminding the child to wait for the character's indication). Experimental participants in the main study were primary-aged pupils at a specialised school for children with ASD (n= 32, 29 males, 3 females), aged 5 to 14 years (mean age 10.67 years, SD= 2.46 years) who represented a range of ability. All had previously received an autism spectrum diagnosis by a senior paediatrician or child psychiatrist, with evaluation of communication, reciprocal social interaction, and repetitive behaviours, using observational assessments including the *Autism Diagnostic Observation Schedule* [15].

Each child individually completed a single session of the flower-selection task in the ECHOES virtual environment, working in a quiet room with two experimenters present. Their interaction with ECHOES was video recorded. Each child followed the same order of events but heard a pre-recorded greeting personalised with their names. Paul introduced himself and asked: "*Will you help me pick some flowers for my mum? I will show you which ones to pick.*" The experimenters repeated these instructions, informing the child that they could pick a flower by touching it onscreen. After a correct choice, the flower flew across the screen to Paul's vase with a fanfare; an incorrect choice led him to say: *"Not that one",* and indicate the target again. This simple narrative provides a framework to support joint attention and to motivate a shared (and repeated) activity between the child and the character.

In the initial trials, many participants had difficulty self-regulating and often touched the screen *before* Paul had indicated a flower. Consequently, most received additional verbal prompts from the experimenters to help them use the touch screen interface at the appropriate time (e.g. *"Touch the flower if you think that's the one that Paul wants"* or *"Wait until Paul shows you"*).[2] There were no training trials, as the participants' behaviour when they were still unfamiliar with Paul's cues was of prime interest. After each trial block Paul thanked the child and invited him or her to pick more flowers. When the child chose to end participation or completed all 36 trials, they heard a final goodbye and thank-you message. Each child's total participation lasted 10-30 minutes, varying with the number of trials completed.

4 Results and Discussion

4.1 Reaction Time and Accuracy Data

Each child in the ASD group (N=32) completed an average of 23.12 trials (range 4-36 trials, SD=10.21). A response was classified as *accurate* if the first touch after the character's indication[3] was to the target flower. An *error* was a touch to any non-target area, or a trial which timed out before the child responded (64 trials or 8.68%). Mean accuracy was very high, at 88.12% (SD=20.22%, median accuracy 95.14%). Accuracy did *not* correlate with age (r=0.23, p=0.21). Contrary to predictions, accuracy did not vary significantly between the four trial types.

Both the high percentage of correct trials and the *pattern* of errors strongly suggest that most children learned to complete the task accurately, despite their brief period of interaction with the VE. Many participants made several errors in the first 6 trials, but after this point most appeared to have completely grasped the task and responded correctly. A small number of participants had occasional errors until early in the second block of 12 trials.[4] Only 3 participants made repeated errors, and did not appear to have learned any kind of causal relationship between Paul's various actions, their own touch screen interaction and the environment's subsequent response.

A 2 (mutual gaze) x 2 (gesture) repeated measures analysis of variance (ANOVA) examined reaction times from the character's indication of the chosen flower to the user's first touch.[5] There were 699 correct trials across 30 participants.[6] The ANOVA revealed a significant interaction of mutual gaze and pointing cues, p<.01 (F=1, 30), with a strong effect size (Cohen's f= 0.477) (see Figure 2). The lack of a strong main

[2] Children were never prompted to attend to the virtual character's face, gaze, or gesture.

[3] Touches prior to the character's indication were ignored.

[4] Twelve participants, mostly older students (aged 11 to 14), made no errors.

[5] Trials with reaction times less than 200 ms were excluded: such responses may be due to the user touching the screen repeatedly before, during and after the character's flower indication.

[6] Two participants were excluded from the reaction time analysis, one for not completing at least one trial of each type, and a second for making some responses with feet, elbows, etc.

effect suggests that it is the *conjunction* of gaze and pointing cues which creates significantly different reaction times, and that this user group may more rapidly process combined gaze and gesture cues than single cues, emphasising the importance of including both types of cues in future virtual environments.

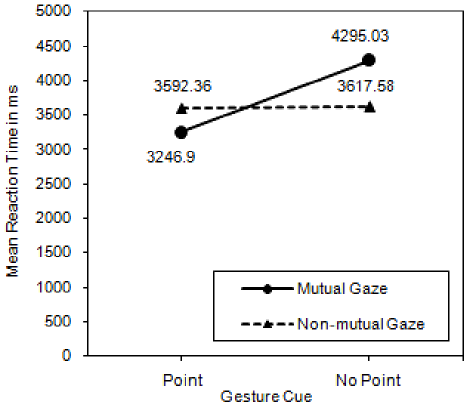


Fig. 2. The effect of character's mutual gaze and gesture on participants' mean reaction time

4.2 Qualitative Analysis and Observation

The video data collected in this study has yet to be fully analysed, but preliminary analysis has been fruitful. The combination of mean accuracy and the experimenters' qualitative observations indicate that young children with ASD successfully—often enthusiastically—engaged with Paul and followed his joint attentional bids to complete the flower selection task. Examples included spontaneously greeting him, answering him directly when he posed questions such as: *"Would you like to help me pick some more [flowers]?"* and expressing surprise or curiosity when Paul did *not* respond to being poked or could not "hear" them. Paul greeting each child by name seemed to be a major factor in generating liking for his character, and in indicating that his actions were both responsive and directed specifically to the child. As far as a perception of mutuality could be said to constitute intentionality, a large proportion of the children across the age range treated Paul as an intentional being.

There were numerous observations of participants spontaneously directing social behaviours to the experimenters and other adults. A large proportion of participants spontaneously gazed to adults, for example after Paul had indicated a flower, but before touching the screen (a possible instance of social referencing, as task demands may still have been unclear in early trials). Also common was spontaneous gaze to adults *after* the child made a flower choice (see Figure 3), when there was no further action demanded by the environment, a possible example of social sharing. Video data shows children concurrently smiling when looking to the adult, or engaging in behaviour regulation such as waving arms or jumping in excitement.[7] Some children

[7] For this population, such behaviours are often considered a means of emotional regulation.

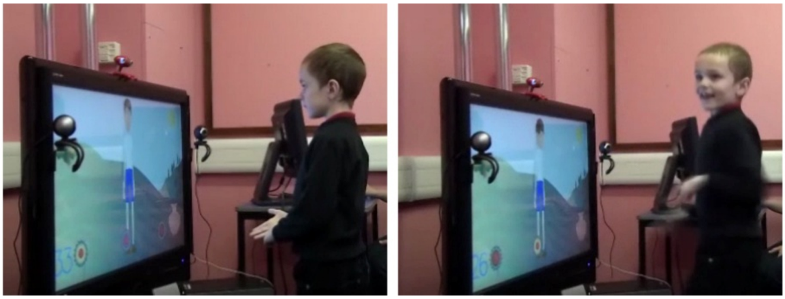


Fig. 3. Left: A child (age 5) watches Paul and waits for his flower indication. Right: The same child turns to look excitedly at his classroom aide (not shown) in an instance of social sharing.

verbally commented on their own success, exclaiming *"I did it!"* or pointing out that they had progressed to a "new level" (e.g. the trial counter had incremented).

5 Conclusions and Further Work

These results are highly encouraging and suggest that young children with ASDs can learn to follow a virtual character's gaze and gesture cues, and to respond through the touch screen interface.[8] The degree and variety of the children's reactions to the character and the high, rapidly-achieved mean accuracy exceeded the experimenters' expectations based on previous literature.[9] We interpret these results as evidence that the children read Paul's actions as mutually relevant and directed toward them specifically, i.e., they perceived him as an intentional being— a "rich" interpretation in developmental psychology terms.

The frequency with which many children initiated social sharing while interacting with the VE is also noteworthy, given that children with ASD are typically impaired in such behaviours. The participants appeared to find the virtual environment novel, exciting, or rewarding enough that they were motivated to share some aspect of that experience with an additional social partner. The observed instances of spontaneous and socially-directed gaze are particularly positive, especially if some are interpreted as instances of social referencing: gazing to another person in an ambiguous situation only makes sense if the child believes that person could be a source of support. This study did not collect baseline video (e.g. in the home or classroom) or questionnaire data documenting each child's verbal ability or social skills, so it is impossible to say whether their social initiations during the experiment notably deviated from their usual behaviour. Overall, the results of this study are an affirmation of the potential for virtual characters as engaging and motivating tools to support social interaction, both within an environment and between child and additional social partners. They inform the full ECHOES system design, and form the basis of future interventions.

[8] What *cannot* be claimed is that these participants have demonstrated any skill learning.

[9] As well as those of the Head Teacher when shown video of the children's interactions.

Acknowledgements. The ECHOES project is funded by ESRC/EPSRC TEL (RES-139-25-0395). This work was undertaken as part of a Masters' dissertation by the first author, at the University of Edinburgh. In addition to the authors, other members of the ECHOES team contributed to the design and development of the environment (see www.echoes2.org). Special thanks to the participating school staff and students.

References

1. DSM-IV, A.P.A.T.F.: DSM-IV: Diagnostic and statistical manual of mental disorders. American Psychiatric Association, Washington, DC (1994)
2. Rajendran, G., Mitchell, P.: Text Chat as a Tool for Referential Questioning in Asperger Syndrome. J. of Speech, Language, and Hearing Research 49, 102–112 (2006)
3. Schmidt, C., Schmidt, M.: Three-dimensional virtual learning environments for mediating social skills acquisition among individuals with autism spectrum disorders. In: IDC 2008: Proceedings of the 7th International Conference on Interaction Design and Children, pp. 85–88. ACM, New York (2008)
4. Parsons, S., Mitchell, P., Leonard, A.: The use and understanding of virtual environments by adolescents with autistic spectrum disorders. Journal of Autism and Developmental Disorders 34(4), 449–466 (2004)
5. Tartaro, A., Cassell, J.: Playing with virtual peers: bootstrapping contingent discourse in children with autism. In: ICLS 2008: Proceedings of the 8th International Conference for the Learning Sciences, pp. 382–389. International Society of the Learning Sciences (2008)
6. Foster, M., Avramides, K., Bernardini, S., Chen, J., Frauenberger, C., Lemon, O., Porayska-Pomsta, K.: Supporting Children's Social Communication Skills through Interactive Narratives with Virtual Characters. In: Proc. of the ACM Multimedia Conference (2010)
7. Goldman, A., Vignemont, F.: Is social cognition embodied? Trends in Cognitive Sciences 13(4), 154–159 (2009)
8. Porayska-Pomsta, K., Bernardini, S., Rajendran, G.: Embodiment as a means for Scaffolding Young Children's Social Skill Acquisition. In: Proc. IDC 2009 (2009)
9. Porayska-Pomsta, K., Frauenberger, C., Pain, H., Rajendran, G., Smith, T.J., Menzies, R., Foster, M.E., Alcorn, A., Wass, S., Bernadini, S., Avramides, K., Keay-Bright, W., Chen, J., Waller, A., Guldberg, K., Good, J., Lemon, O.: Developing Technology for Autism: an interdisciplinary approach. Personal and Ubiquitious Computing (in press)
10. Sigman, M., Ruskin, E.: Continuity and change in the social competence of children with autism, Down syndrome, and developmental delays. Wiley-Blackwell, Malden (1999)
11. Charman, T.: Why is joint attention a pivotal skill in autism? Philosophical Transactions: Biological Sciences 358, 315–324 (2003)
12. Behne, T., Carpenter, M., Tomasello, M.: One-year-olds comprehend the communicative intentions behind gestures in a hiding game. Developmental Science 8(6), 492–499 (2005)
13. Pellicano, E., Macrae, C.N.: Mutual eye gaze facilitates person categorization for typically developing children, but not for children with autism. Psychonomic Bulletin & Review 16(6), 1094–1099 (2009)
14. Tomasello, M., Carpenter, M., Liszkowski, U.: A new look at infant pointing. Child Development 78(3), 705–722 (2007)
15. Lord, C., Rutter, M., DiLavore, D., Risi, S.: Autism Diagnostic Observation Schedule (ADOS). Western Psychological Services, Los Angeles (1999)

A Comparison of the Effects of Nine Activities within a Self-directed Learning Environment on Skill-Grained Learning

Ari Bader-Natal, Thomas Lotze, and Daniel Furr*

Grockit, Inc.
San Francisco, CA USA
{ari,thomas}@grockit.com
http://grockit.com

Abstract. Self-directed learners value the ability to make decisions about their own learning experiences. Educational systems can accommodate these learners by providing a variety of different activities and study contexts among which learners may choose. When creating a software-based environment for these learners, system architects incorporate activities designed to be both effective and engaging. Once these activities are made available to students, researchers can evaluate these activities by analyzing observed usage and performance data by asking: Which of these activities are most engaging? Which are most effective? Answers to these questions enable a system designer to highlight and encourage those activities that are both effective and popular, to refine those that are either effective or popular, and to reconsider or remove those that are neither effective nor popular. In this paper, we discuss Grockit – a web-based environment offering self-directed learners a wide variety of activities – and use a mixed-effects logistic regression model to model the effectiveness of nine of these supplemental interventions on skill-grained learning.

Keywords: self-directed learning, learner control, skill-grained evaluation.

Educational software designed for the classroom is often only effective in the classroom, simply because students use this software only when they are required to do so. For non-compulsory learning software to be effective, being *engaging* is a necessary (but not sufficient) precondition. As the notion of engagement is subjective, one approach to building a system that many learners find engaging is to support a variety of modes and activities and allow each learner to find their preferred niche. Grockit, a web-based learning environment designed for individual students who share a common domain-specific learning goal, takes this approach by incorporating two dimensions of variety/flexibility: context and control. At any point in time, learners can choose from among three contexts of study: individual practice, peer group study, and instructor-led lessons. The learner can also choose the amount of control that he or she wishes to exert to define the learning experience [5]: with learner-driven control offered through HCI affordances and system-driven control provided via AI approaches (such as an adaptive

* Work done at Grockit. Present address: School of Education, University of California Berkeley.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 15–22, 2011.

problem selection algorithm based on an Item Response Theory model [2]). Grockit pursues an engaging learning experience by means of game design and social interactions both addressed in prior work [1,2], and internal surveys continues to indicate that the vast majority of participants find Grockit's learning environment to be engaging. The variety introduced to increase engagement does, however, add complexity to the attribution of the effectiveness. In this work, we summarize nine of the interventions incorporated into the Grockit system, and evaluate the extent to which each of these is an effective addition to the learning platform.

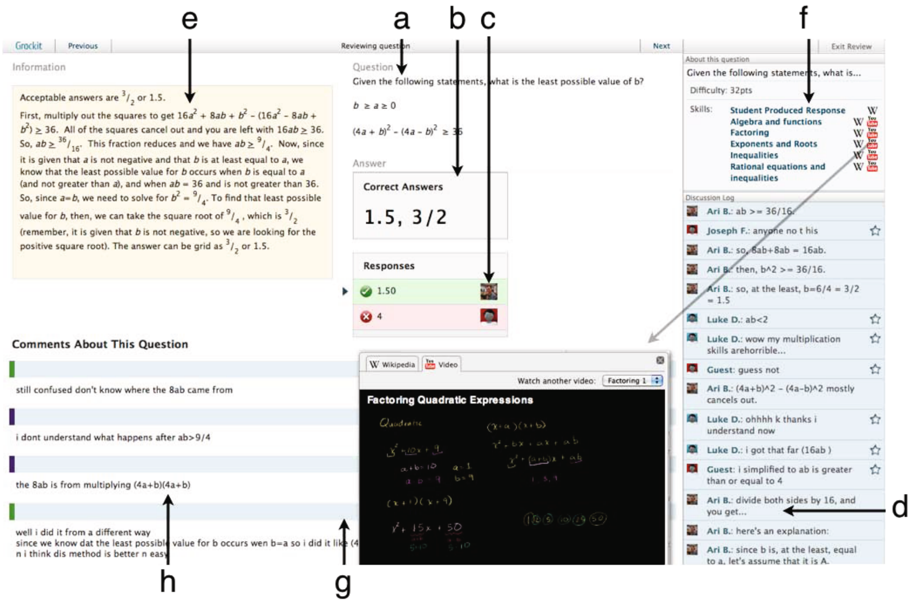
1 Interventions within Grockit

Grockit provides a place for students to master new skills and exercise what they learn through three contexts for problem solving: (*a.*) *individual study*, which uses an Item Response Theory model to provide that student with appropriate challenges for learning [7], (*b.*) *small group study*, which leverages collaborative learning dynamics to provide students with a social learning network that can help motivate and assist them [2], (*c.*) *instructor-led classes*, which draw on an expert's domain knowledge and experience to provide a guided and structured path for larger groups of learners.

The core activity within all three learning contexts involves answering multiple-choice and numeric response problems in some well-defined learning domain (e.g. an Algebra I course, the GMAT exam, a Grade 8 English Language Arts course), and then reviewing expert-authored solutions and explanations for each of these problems. In the small group and instructor-led settings, all participants see the same question at the same time, enabling group discussion around problems and solutions. In addition to the core problem-solving activity, learners have access to a number of supplemental learning activities motivated by work in prior systems, and introduced to the Grockit environment with the goal of contributing to the learning gains of participating students. In this study, we focus on nine of these activities:

explanation_read: ***Read an explanation of the question immediately after answering it.***
For each question in Grockit's item database, the author of the question prepared an explanation of the solution and, for multiple-choice questions, explanations or comments about each answer choice. After testing a variety of different contexts within the application for incorporating these explanations, we chose to make these explanations available only during individual study sessions.[1] These in-game explanations were introduced in order to provide students with a cohesive expert-authored solution – visible after the student answers the question and sees which answer choice is correct. Viewing these explanations is presented as an optional activity: a "view an explanation for this question" link is displayed above each question, and the student must click the link to reveal the explanation. When given the opportunity to view an explanation after answering a question and seeing the correct response, 48% of students *who answered incorrectly* and

[1] We found that the time required to benefit from explanations varied widely among students, and was therefore a better fit for self-paced review rather than for the real-time group study. For a more details on decisions around interaction synchronicity, see Bader-Natal [2].



Information

Acceptable answers are $^3/_2$ or 1.5.

First, multiply out the squares to get $16a^2 + 8ab + b^2 - (16a^2 - 8ab + b^2) \geq 36$. All of the squares cancel out and you are left with $16ab \geq 36$. So, $ab \geq {}^{36}/_{16}$. This fraction reduces and we have $ab \geq {}^9/_4$. Now, since it is given that *a* is not negative and that *b* is at least equal to *a*, we know that the least possible value for *b* occurs when *b* is equal to *a* (and not greater than *a*), and when $ab = 36$ and is not greater than 36. So, since $a = b$, we need to solve for $b^2 = {}^9/_4$. To find that least possible value for *b*, then, we can take the square root of $^9/_4$, which is $^3/_2$ (remember, it is given that *b* is not negative, so we are looking for the positive square root). The answer can be grid as $^3/_2$ or 1.5.

Comments About This Question

still confused don't know where the 8ab came from

i dont understand what happens after ab>9/4

the 8ab is from multiplying (4a+b)(4a+b)

well i did it from a different way
since we know dat the least possible value for b occurs wen b=a so i did it like (4
n i think dis method is better n easy

Question

Given the following statements, what is the least possible value of b?

$b \geq a \geq 0$

$(4a + b)^2 - (4a - b)^2 \geq 36$

Answer

Correct Answers

1.5, 3/2

Responses

✓ 1.50

✗ 4

Factoring Quadratic Expressions

About this question

Given the following statements, what is...

Difficulty: 32pts

Skills: Student Produced Response
Algebra and functions
Factoring
Exponents and Roots
Inequalities
Rational equations and inequalities

Discussion Log

Ari B.: ab >= 36/16.
Joseph F.: anyone no t his
Ari B.: so, 8ab+8ab = 16ab.
Ari B.: then, b^2 >= 36/16.
Ari B.: so, at the least, b=6/4 = 3/2 = 1.5
Luke D.: ab<2
Luke D.: wow my multiplication skills arehorrible...
Guest: guess not
Ari B.: (4a+b)^2 - (4a-b)^2 mostly cancels out.
Luke D.: ohhhh k thanks i understand now
Luke D.: i got that far (16ab)
Guest: i simplified to ab is greater than or equal to 4
Ari B.: divide both sides by 16, and you get...
Ari B.: here's an explanation:
Ari B.: since b is, at the least, equal to a, let's assume that it is A.

Fig. 1. A Review includes several components, including: (a.) the original question and answer choices, (b.) the correct answer, (c.) each of the answers submitted by the students in the session, (d.) the discussion transcript from that session, (e.) expert explanations of the question and each answer choice, (f.) metadata about the problem including difficulty level and list of associated skills, (g.) access to videos and blog posts discussing each these concepts, and (h.) an asynchronous discussion thread among all students who have reviewed that question

18% of students *who answered correctly* chose to view the explanation.[2] If we find that viewing an explanation immediately following an incorrect response is an effective intervention, we might start displaying these explanations to all students in individual study sessions following an incorrect response, without them needing to request it.

reviewed: ***Reviewed a question from a study session.*** As mentioned above, a post-hoc review of study sessions is available to students, in which no per-question time constraints are necessary, since the solo nature of the activity means that synchronizing pace with other students is not necessary. Over the past few years, these reviews have grown to include an assortment of resources for the student to draw on, illustrated in Fig. 1. Of these components, three involve actions that are addressed separately in below. Beyond the practical logistics of time necessary to engage in these activities, the reviews serve to distribute skill practice over time (rather than to compress all practice into the initial practice session), an approach that seems to be supported by data on the spacing-effect [4].

watched_video: ***Watched an instructional video about the skill.*** Watching an instructor explain a concept and work example problems is one of the primary modes

[2] Based on item responses from 10/1/2010 - 12/31/2010 from people studying for the GMAT.

of face-to-face instruction, and a common component of online learning environments. For each question in Grockit, the set of skills required to solve the problem are listed next to the question in the reviews, along with other question metadata. For each concept listed, the student can choose to watch short videos explaining the concept, embedded from public video sharing sites such as YouTube, with videos selected by content authors based on relevance and quality.

viewed_textbook: ***Read an expert-authored description of the underlying skill.*** Similar to the videos described above, each of the skills associated with the question are correlated with written explanations of those skills (but not of the specific question.) These skill-explanations were originally prepared as a series of blog posts.

question_comment: ***Appended a message to a question during a review session.*** Within group study sessions, students are able to discuss questions as they work on them, in real-time. In reviews, students can read their past discussions, but cannot get real-time answers to their questions. We introduced an asynchronous discussion thread for each question to allow students to discuss with others who have seen the question, even if at a different time.[3]

discussed: ***Typed a message after answering a question in group study.*** In group study sessions, a chat box is displayed next to the question that the students are attempting to solve. While discussions about a question may include no participants with knowledge or expertise, studies by Smith et al. suggest that small group discussions following a question can be beneficial even when none of the participants had correctly answered the question initially [9].

questioned: ***Asking questions, in game discussions.*** We use the presence of a question-mark in the discussion as a low-fidelity indicator of a request for help. The discussions that transpire are generally a combination of on-task peer-assistance and off-task conversations.[4] While this signal is clearly quite noisy and the outcome not definitive, we prefer to include this rather than nothing at all.

tutor_led: ***Participating in an instructor-led lesson.*** The three modes of study in Grockit – instructor-led sessions, group study, and individual practice – have parallels in Dron and Anderson's distinction between groups, networks, and collectives [6]. Of the three, the instructor-led sessions most closely resemble a traditional classroom: The instructor schedules a session and some number of students attend. The instructor can incorporate slides, whiteboards, and shared text editors into the session, and while practice problems are done, the primary focus is on instruction. We include this to determine if these structured lessons are of measurable value.

with_tutor: ***Participated in a group study session in which a tutor was present.*** The instructors who lead lessons also frequently join ongoing peer-group study sessions. Instructors generally participate and encourage discussion in these sessions,

[3] Comments in the asynchronous discussion threads are often more thoughtfully prepared and are less context-dependent than the more casual and interactive discussion messages in synchronous group games. We include comment authoring in this analysis because we wish to see if taking the time to participate in this forum has an effect on skill learning outcomes.

[4] The casual nature of off-task discussions serves to reduce the stress associated with studying, so we do not discourage these discussions.

but they do not lead them in the formal way that they lead lessons. We include this to determine if this more casual participation in group study is beneficial.

2 Methods

We formulate the effect of available interventions on skill-grained learning as follows: *After a student incorrectly answers a question involving some skill, engages in an intervention involving that skill, and then attempts a subsequent question involving that same skill, what effect does that intervention have on second response accuracy?*

For this analysis, we consider data collected during a two-month period (October 1 - December 1, 2010). We consider two types of data: item responses and item interventions, and exclude item responses and interventions from all user accounts belonging to teachers, tutors, system administrators, and anonymous guests. Each item in the Grockit database is associated with one or more skill tags describing the concepts required to solve the problem. Both responses and interventions can be associated with skills, and here we use skills as the granularity for analysis.

Each student's performance on a specific skill can be organized into a timeline of *item responses* on that skill, which may be correct or incorrect, and *item interventions* on that same skill, which are intended to improve the student's performance. When an item intervention is followed by an item response, we have the opportunity to see how the intervention impacted the user's performance on that skill.

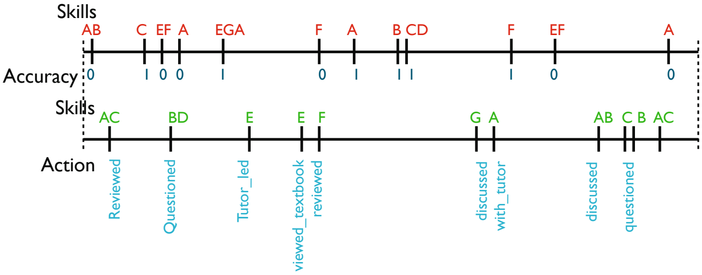


Fig. 2. An dual-timeline example for a student. The upper line contains skill-tagged item responses and the lower line contains skill-tagged interventions.

For *item responses*, we use $r_n^{(s,k)}$ and $t_n^{(s,k)}$ to denote the response accuracy and timestamp, respectively, for the n^{th} response by to skill k by student s (where $r_n^{(s,k)} \in \{0,1\}$). For *item interventions*, we use $T_{(j,u)}^{(s,k)}$ to denote the time at which student s participated in their u^{th} intervention of type j (where $j = 1..9$ for the nine intervention types) on skill k. We may then determine, for each user response, which interventions the student participated in before that response. If the student participated in a certain type of intervention for the skill between two subsequent item responses on that skill, we record this as a 1. If there was no such intervention, we record it as a 0:

$$i_{(j,n)}^{(s,k)} = \begin{cases} 1 & \mathit{if} \;\; \exists T_{(j,u)}^{(s,k)} : t_{n-1}^{(s,k)} < T_{(j,u)}^{(s,k)} < t_n^{(s,k)} \\ 0 & \mathit{otherwise} \end{cases}$$

Table 1. Example rows from the combined dataset used for analysis

person	skill	first_response_time	second_response_time	second_difficulty	reviewed	explanation_read	discussed	questioned	watched_video	viewed_textbook	question_comment	tutor_led	with_tutor	first_accuracy	second_accuracy
u_1	s_a	2010-10-22 18:19:20	2010-10-22 18:21:38	-0.68	0	0	1	1	0	0	0	1	1	0	0
u_2	s_a	2010-10-22 18:21:38	2010-10-22 18:23:12	-1.09	0	1	0	0	0	0	0	0	0	0	0
u_2	s_b	2010-10-22 18:23:12	2010-10-22 18:25:17	-0.98	1	0	0	0	1	0	0	0	0	0	1

We only measure interventions after the user's previous response on this skill, as we consider these to be the strongest indicators of an improvement due to the intervention. Table 1 illustrates a few example rows from the resulting data set.

We use a mixed-effects regression to model the second response accuracy. As we are looking for evidence of learning, we consider only those records for which the previous response was incorrect (i.e. responses $r_n^{(s,k)}$ where $r_{n-1}^{(s,k)} = 0$); we view a correct response to the second item to be an indicator of learning. Among these records, we treat the nine interventions as fixed effects. We also include the difficulty of the second item (d_{q_2}) as a fixed effect, as we expect the second question's difficulty to (negatively) impact the person's response accuracy on that question. We treat the variance between students as a random effect in this model, $\alpha_s \sim N(0, \psi^2)$:

$$\textit{logit}\left\{P\left(r_n^{(s,k)} = 1\right)\right\} = \beta_0 + \beta_d d_{q_2} + \beta_1 i_{(1,n)}^{(s,k)} + \cdots + \beta_9 i_{(9,n)}^{(s,k)} + \alpha_s$$

We note a few weaknesses in this approach. This adjacent-pair analysis provides insight into short-term effects of individual interventions. Learners generally respond to a sequence of items for each skill, and these cumulative effects are not captured in this model, resulting in a weak signal of learning. Additionally, most questions are tagged with more than one skill, and an incorrect response cannot be attributed to a single skill. Finally, we recognize that when a student engages in a particular intervention, they are both benefitting from it and signaling that they believe that they will benefit from it. The benefits may therefore be affected by the biased sample. Overall, since this is not a randomized controlled experiment and learners can self-select their interventions, we can attribute correlation but not causation.

3 Results

Table 2 reports the coefficients estimated from the mixed-effects logistic regression model, obtained using the *lme4* package for the R statistical environment [3,8]. The difficulty of the second item (*second_difficulty*) has a statistically significant effect on the second response accuracy, as was expected. The more difficult the item, the lower the expected response accuracy.[5] Of the nine interventions examined in all, five had a statistically significant positive effect (at the $\alpha = 0.05$ level), one had a statistically

[5] Item difficulty is estimated based on a three-parameter item response theory model.

Table 2. Model coefficients from the Generalized Linear Mixed Model. The student is treated as a random effect (variance: 0.98). Stars indicate significance at the $\alpha = 0.05$ level.

	Estimate	Std. Error	z value	Pr(>|z|)	
(Intercept)	-0.17	0.01	-14.10	0.00	*
reviewed	0.04	0.02	2.51	0.01	*
explanation_read	0.04	0.01	2.77	0.01	*
discussed	0.05	0.01	5.18	0.00	*
questioned	-0.02	0.01	-1.55	0.12	
watched_video	-0.82	0.52	-1.57	0.12	
viewed_textbook	-0.36	0.17	-2.12	0.03	*
question_comment	0.14	0.10	1.36	0.17	
tutor_led	0.22	0.11	1.97	0.05	*
with_tutor	0.10	0.02	5.85	0.00	*
second_difficulty	-0.68	0.00	-223.07	0.00	*

significant negative effect, and three had no statistically significant effect. The interventions with the highest coefficients involved the expert instructors, with a 0.22 increase in the log odds of learning in instructor-led lessons (*tutor_led*) and a 0.10 increase in group games in which an instructor participates (*with_tutor*). Reviewing items (*reviewed*) also has a significant effect, with an estimated coefficient of 0.04. This coefficient represents the increase in the log odds of success (i.e. a correct to the following attempt at a question of the same skill) for this student, if this student reviewed a question involving that skill prior to the second response. Participating in group game discussions was estimated to increase the log odds (logits) of learning by 0.05. Choosing to view an explanation (*explanation_read*) after answering a question in a individual practice session increased the outcome by 0.04 logits, and reviewing a question (*reviewed*) increased the outcome by 0.04 logits. Neither watching a video (*Watched_video*) nor leaving a comment (*question_comment*) had a statistically significant effect (beyond that of reviewing itself). Unexpectedly, viewing the "textbook" concept explanations *viewed_textbook* had a statistically significant negative effect. Asking a question within a group discussion (*questioned*) was not found to have a significant effect.

4 Discussion

This analysis represents our first effort to quantify and evaluate the learning outcomes associated with individual activities available within Grockit. The variety of available tools in the learning environment adds both richness to the experience and complexity to the attribution of learning gains. The results here suggest which of the interventions analyzed were most effective and, coupled with an understanding of how engaging each activity is, these results can inform decisions around which interventions to highlight, which to refine, and which to reconsider. Given the positive effect observed among students who choose to view an question explanation in an solo practice after an incorrect response, we might automatically show these, rather than requiring students to opt-in each time. As for activities displaying no statistical significance, we are now discussing modifications expected to make them more effective.[6]

[6] We suspect that the non-significant effect of asking a question during discussion may be due to imperfect identification, which includes both on-task and off-task (e.g. social) questions. This could be clarified if questions were coded as such and tested separately.

In another study currently in progress, we use a randomized controlled design to evaluate overall learning gains from participation, without attribution to interventions by type. Where the current analysis only examines select interventions, the A/B design is more comprehensive, incorporating the core problem solving practice and intermittent assessments that are not captured in the present analysis. To understand the effect of a complex learning environment, we believe that both approaches are valuable.

While students are generally required to use (and continue using) educational software introduced in a formal learning setting, no such obligation governs use of educational software by self-directed learners. In order to be capable of impacting learning for these students, a system must be *both* sufficiently engaging for students to continue using it *and* effective. Different people find different learning contexts and activities engaging, so Grockit chose to introduce and leverage *variety* – learner choice and control over how, when, and with whom one learns – to address the assorted needs and preferences of self-directed learners. A large (and growing) number of students do, in fact, find the platform engaging, as evidenced by internal survey data and observed time-on-task. In this analysis, we find that several of the learning interventions incorporated into the platform are effective, with participation associated with skill-grained learning. By building a platform that is engaging and incorporates effective interventions, Grockit has created an environment uniquely-suited to the needs of the self-directed learner.

References

1. Bader-Natal, A.: Incorporating game mechanics into a network of online study groups. In: Craig, S.D., Dicheva, D. (eds.) Supplementary Proceedings of the 14th International Conference on Artificial Intelligence in Education (AIED-2009), Intelligent Educational Games workshop, July 2009, vol. 3, pp. 109–112. IOS Press, Brighton (2009)
2. Bader-Natal, A.: Interaction synchronicity in web-based collaborative learning systems. In: Bastiaens, T., Dron, J., Xin, C. (eds.) Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2009, pp. 1121–1129. AACE, Vancouver (2009)
3. Bates, D., Maechler, M.: lme4: Linear mixed-effects models using S4 classes (2010)
4. Donovan, J.J., Radosevich, D.J.: A meta-analytic review of the distribution of practice effect: Now you see it, now you don't. Journal of Applied Psychology 84, 795–805 (1999)
5. Dron, J.: Control and constraint in e-learning: Choosing when to choose. Information Science Publishing, United Kingdom (2007)
6. Dron, J., Anderson, T.: Collectives, networks and groups in social software for e-Learning. In: Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education Quebec, vol. 16, p. 2008 (2007) (retrieved February)
7. Lord, F.M.: Applications of Item Response Theory to practical testing problems. Lawrence Erlbaum Associates, Hillsdale (1980)
8. R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2010)
9. Smith, M.K., Wood, W.B., Adams, W.K., Wieman, C., Knight, J.K., Guild, N., Su, T.T.: Why peer discussion improves student performance on in-class concept questions. Science 323(5910), 122–124 (2009)

Towards Predicting Future Transfer of Learning

Ryan S.J.d. Baker[1], Sujith M. Gowda[1], and Albert T. Corbett[2]

[1] Department of Social Science and Policy Studies, Worcester Polytechnic Institute
100 Institute Road, Worcester MA 01609, USA
{rsbaker,sujithmg}@wpi.edu
[2] Human-Computer Interaction Institute, Carnegie Mellon University, 5000 Forbes Avenue,
Pittsburgh, PA 15213, USA
corbett@cmu.edu

Abstract. We present an automated detector that can predict a student's future performance on a transfer post-test, a post-test involving related but different skills than the skills studied in the tutoring system, within an Intelligent Tutoring System for College Genetics. We show that this detector predicts transfer better than Bayesian Knowledge Tracing, a measure of student learning in intelligent tutors that has been shown to predict performance on paper post-tests of the same skills studied in the intelligent tutor. We also find that this detector only needs limited amounts of student data (the first 20% of a student's data from a tutor lesson) in order to reach near-asymptotic predictive power.

Keywords: Transfer, Bayesian Knowledge Tracing, Educational Data Mining, Student Modeling, Robust Learning.

1 Introduction

Over the previous two decades, knowledge engineering and educational data mining (EDM) methods have led to increasingly precise models of students' knowledge as they use intelligent tutoring systems and other AIED systems. Modeling of student knowledge has been a key theme in AIED from its earliest days. Models of student knowledge have become successful at inferring the probability that a student knows a specific skill at a specific time, from the student's pattern of correct responses and non-correct responses (e.g. errors and hint requests) up until that time [cf. 8, 14, 16, 19]. In recent years, the debate about how to best model student knowledge has continued, with attempts to explicitly compare the success of different models at predicting future correctness within the tutoring software studied [cf. 12, 16].

However, the ultimate goal of AIED systems is not to promote better future performance within the system itself. Ideally, an intelligent tutoring system or other AIED system should promote "robust" learning [13] that is retained over time [15], transfers to new situations [20], and prepares students for future learning [6]. Historically, student modeling research has paid limited attention to modeling the robustness of student learning. Although studies have demonstrated that learning in intelligent tutors can be made robust [1, 7, 17], student models used in intelligent tutors have typically not explicitly modeled robustness, including whether knowledge will

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 23–30, 2011.

transfer. In fact, only a handful of studies have even attempted to predict immediate posttest performance on the same skills studied in a tutor [e.g., 3, 8, 10, 19], a very limited form of transfer. For instance, Bayesian Knowledge Tracing models of student knowledge have been shown to predict this type of post-test performance [8], but with a small but consistent tendency to overestimate students' average post-test performance, systematic error that can be corrected by incorporating pretest measures of students' conceptual knowledge into the knowledge tracing model [9]. Other student models have modeled the inter-connection between skills, within a tutor [cf. 14]. However, it is not clear whether this can in turn support prediction of transfer to different skills and situations outside of the tutor.

Within this paper, we present a model designed to predict student performance on a transfer post-test, a post-test involving related but different skills than the skills studied in the tutoring system, within a Cognitive Tutor for genetics problem solving [10]. This model is generated using a combination of feature engineering and linear regression, and is cross-validated at the student level. We compare this model to Bayesian Knowledge Tracing – a student model shown to predict post-test performance – as a predictor of transfer. As a student model predicting transfer will be most useful if it can be used to drive interventions fairly early during tutor usage, we also analyze how much student data is needed for the model to be accurate.

2 Data Set

The data set used in the analyses came from the Genetics Cognitive Tutor [10]. This tutor consists of 19 modules that support problem solving across a wide range of topics in genetics. Various subsets of the 19 modules have been piloted at 15 universities in North America. This study focuses on a tutor module that employs a gene mapping technique called *three-factor cross*, in which students infer the order of three genes on a chromosome based on offspring phenotypes, as described in [3]. In this laboratory study, 71 undergraduates enrolled in genetics or in introductory biology courses at Carnegie Mellon University used the three-factor cross module. The students engaged in Cognitive Tutor-supported activities for one hour in each of two sessions. All students completed standard three-factor cross problems in both sessions. During the first session, some students were assigned to complete other cognitive-tutor activities designed to support deeper understanding; however, no differences were found between conditions for any robust learning measure, so in this analysis we collapse across the conditions and focus solely on student behavior and learning within the standard problem-solving activities. The 71 students completed a total of 22,885 problem solving attempts across 10,966 problem steps in the tutor.

Post-tests, given by paper-and-pencil, consisted of four activities: a straightforward problem-solving post-test discussed in detail in [3], a transfer test, a test of preparation for future learning, and a delayed retention test. Within this paper we focus on predicting performance on the transfer test of robust learning. The transfer test included two problems intended to tap students' understanding of the underlying processes. The first was a three-factor cross problem that could not be solved with the standard solution method and required students to improvise an alternative method.

The second problem asked students to extend their reasoning to four genes. It provided a sequence of four genes on a chromosome and asked students to reason about the crossovers that must have occurred in different offspring groups.

Students demonstrated good learning in this tutor, with an average pre-test performance of 0.31 (SD=0.18), an average post-test performance of 0.81 (SD=0.18), and an average transfer test performance of 0.85 (SD=0.18). The correlation between the problem-solving post-test and the transfer test was 0.590 suggesting that, although problem-solving skill and transfer skill were related, transfer may be predicted by more than just simply skill at problem-solving within this domain.

3 Analysis of Model Using Cross-Validation

In this paper, we introduce a model that predicts each student's performance on the transfer test, using a hybrid of data mining and knowledge engineering methods. Within this approach, a small set of features are selected based on theory and prior work to detect related constructs. These features are based on thresholds which are given initial values but are also optimized by grid search, using as goodness criterion the cross-validated correlation between an individual feature and each student's performance on the transfer test. Finally a model is trained on these features (using both the original and optimized thresholds) to predict each student's performance on the transfer test, and is cross-validated. We then compare this model to a baseline prediction of transfer, Bayesian Knowledge Tracing (BKT) [8] fit using brute force, which has been previously shown to predict student post-test problem-solving performance reasonably well within this lesson [3]. Recent work in other tutoring systems has suggested that other algorithms (BKT fit using Expectation Maximization; Performance Factors Analysis) may fit within-tutor performance slightly better than BKT fit using Brute Force [12, 16], but thus far no published studies have demonstrated that these algorithms fit post-test performance better. As BKT accurately predicts problem-solving post-tests, and the transfer test was reasonably correlated to the problem-solving post-test in this study, it should correlate reasonably well to transfer. Hence, a useful detector predicting transfer should perform better than BKT, under cross-validation.

3.1 Feature Engineering

The first step of our process was to engineer the feature set. As we were predicting performance on a measure external to the tutor, given after tutor usage, we focused on proportions of behavior across the full period of use of the tutoring system (e.g. what proportion of time a student engaged in each behavior). Our data features consisted of the following behaviors (the prime notation connotes a feature closely related to the previous feature): 1) Help avoidance [2]; 1') Requesting help on relatively poorly known skills; 2) Long pauses after receiving bug messages (error messages given when the student's behavior indicates a known misconception), which may indicate self-explanation; 2') Short pauses after receiving bug messages, indicating failure to self-explain; 3) Long pauses after reading hint messages; 4) Long pauses after reading hint message(s) and then getting the next action right [cf. 18]; 5) Off-task behavior;

5') Long pauses that are not off-task; 6) Long pauses on skills assessed as known; 7) Gaming the system [4]; 7') Fast actions that do not involve gaming; 8) Carelessness, detected as contextual slip [3]; 9) Learning spikes [5].

Three of these features were incorporated into the final model predicting transfer: 1, 2' and 7'. We will discuss our model development process in a subsequent section, but in brief, no additional feature both achieved better cross-validated performance than zero on its own, and also improved cross-validated predictive power in a model already containing these three features. The exact operational definition of these features was:

1: Proportion of actions where the student has a probability under N of knowing the skill, according to Bayesian Knowledge Tracing [8], does not ask for help, and makes an error on their first attempt. Initial value of N = 60% probability.
2': Proportion of actions where the student enters an answer labeled as a bug, and then makes their next action in under N seconds. Initial value of N = 5 seconds.
7': Proportion of actions where the student enters an answer or requests a hint in under N seconds, but the action is not labeled as gaming, using a gaming detector previously trained on a full year of data from high school algebra [4]. Initial value of N = 1 s.

Each of these three features depends on a threshold parameter, N; adjusting a feature's parameter can result in very different behavior. In some analyses below, we used an arbitrary but plausible value of N chosen prior to optimization, as given above. Features were then optimized to select optimal thresholds, using grid search. Parameters involving probabilities were searched at a grid size of 0.05; parameters involving time were searched at a grid size of 0.5 seconds.

3.2 Detector Development

Our first step towards developing a detector was to fit a one-parameter linear regression model predicting transfer from each feature, using leave-out-one-cross-validation (LOOCV), in RapidMiner 4.6. LOOCV was conducted at the student level, the overall level of the analysis. The cross-validated correlations for single-feature regression models are shown in Table 1. This process was conducted for both original and optimized threshold parameters. Both help avoidance (1) and making fast responses after bugs (2') were found to be negatively associated with transfer. Fast non-gaming actions (7') were positively correlated with transfer, perhaps because these actions are a signal that the skill has been acquired very strongly (additionally, for low values of the threshold, very few fast non-gaming responses are help requests, which is some additional evidence for interpreting this feature in this fashion).

Table 1. Goodness of single-feature linear regression models at predicting transfer

Feature	Direction of relationship	Cross-validated r (orig. thresholds)	Cross-validated r (optimized thresholds)
1. Help Avoidance	Neg.	0.362	0.376
2'. Fast After Bugs	Neg.	0.167	0.269
7'. Fast Not Gaming	Pos.	0.028	0.189

Given each set of features, we developed linear regression models using RapidMiner 4.6. To find the set of parameters, Forward Selection was conducted by hand. In Forward Selection, the best single-parameter model is chosen, and then the parameter that most improves the model is repeatedly added until no more parameters can be added which improve the model. Within RapidMiner, feature selection was turned off, and each potential model was tested in a separate run, in order to determine how well a specific set of features predicts transfer. Keeping feature selection on would result in some features being filtered out for some sub-sets of the data, making it harder to infer how well a specific set of features predicts transfer. The goodness metric used was the LOOCV correlation between the predictions and each student's performance on the transfer test. In addition, as an additional control on over-fitting, we did a first pass where we eliminated all features that, taken individually, had cross-validated correlation below zero. We give differences in cross-validated correlation rather than statistical significance tests, as a measure of model generalizability; comparing cross-validated correlations is a redundant test [cf. 11].

The cross-validated correlation of the model to the transfer test was 0.407, for the original thresholds, and 0.416 for the optimized thresholds. By comparison, the Bayesian Knowledge Tracing estimates of student knowledge achieved a cross-validated correlation of 0.353 to the transfer test. Hence, the transfer model appears to perform better than this reasonable baseline.

We then investigated the possibility that multiplicative interaction features (where one feature is multiplied with another feature) would lead to a better model. To reduce the potential for over-fitting, we restricted our analysis to multiplicative features consisting of the 3 features above, and the 3 original features. This model achieved a cross-validated correlation of 0.435 to the transfer test, for the original thresholds, and 0.428 for the optimized thresholds.

One question is whether the resultant models are better predictors solely of transfer or of student knowledge overall. This can be investigated by examining how well the transfer prediction models predict the regular problem-solving post-test, with no re-fitting. If we predict the problem-solving post-test using Bayesian Knowledge-Tracing, we obtain a correlation of 0.535. As seen in Table 2, each of the four transfer prediction models perform better than this at predicting the post-test, with the optimized model without multiplicative interactions performing best (r=0.633).

Table 2. Cross validated correlation between models and transfer test

Model	Cross-validated correlation to transfer test	Correlation to problem-solving test
Only BKT	0.353	0.535
Model with optimized features (no interactions)	0.416	0.633
Model with original features (no interactions)	0.407	0.546
Model with optimized features (multiplicative interactions)	0.428	0.615
Model with original features (multiplicative interactions)	0.435	0.598

4 Analysis of Model for Use in Running Tutor

One potential concern with models developed using proportions of behavior across entire episodes of tutor use is that the models may not be usable to drive interventions in a running tutor. If an entire tutor lesson worth of data is required for accurate inference, the detector may have low usefulness for intervention compared to approaches such as Bayesian Knowledge Tracing which make a prediction after each problem-solving step [8]. However, it is possible to make a version of the transfer detector that can be used in a running tutor. Specifically, it is possible to take the data up to a specific problem step, compute the model features using only the data collected up until that point, and make an inference about the probability of transfer. In this section, we investigate how much data is needed for the model to make accurate predictions within this data set, comparing our model's predictive power to Bayesian Knowledge-Tracing, when both are given limited data.

Our first step in this process is to construct 20 subsets of data containing the first N percent of each student's interactions within the tutor, using every increment of 5% of the data. Our process for doing this does not take skills into account – e.g. data from some skills may not be present in the first 5%. We then compute the feature values for each data subset, using the optimized thresholds. Next, we apply the transfer prediction model generated using the full data set to the new data sets (e.g. we do not refit the models for the new data sets). We also apply Bayesian Knowledge Tracing on the limited data sets without re-fitting the BKT parameter estimates. After obtaining the predictions we compute the correlation between each of the predictions and each student's performance on the transfer test. Cross-validation is not used, as the model is not being re-fit in either case.

Figure 1 shows the predictive performance of the transfer prediction model and BKT based on having the first N percent of the data. From the graph we can see that the transfer prediction model performs substantially better than BKT for small amounts of data. For instance, with only the first 20% of the data, the transfer prediction model achieves a solid correlation of 0.463 while the BKT model achieves a much weaker correlation of 0.254. These findings suggest that it may be possible to use the transfer prediction model to drive interventions, from very early in tutor usage.

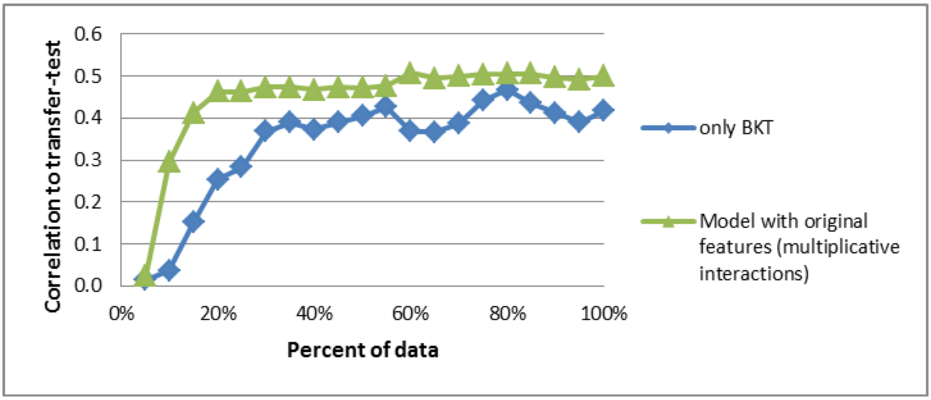


Fig. 1. Predicting transfer with first N percent of the data

5 Conclusions

Within this paper, we have presented a model which can predict with reasonable accuracy how well a student will perform on a transfer post-test, a post-test involving related but different skills than the skills studied in the tutoring system, within a Cognitive Tutor for College Genetics. This model is based on the percentage of student actions that involve help avoidance [2], fast actions which do not involve gaming the system [4], and fast responses after receiving a bug message. Interestingly, two of these features (help avoidance and fast responses after bugs) appear to reflect meta-cognitive behavior rather than reflecting what students know, at least according to prior theory that these behaviors are meta-cognitive in nature [e.g. 2,18]. The result is in line with theory that suggests a key role for meta-cognition in transfer [13].

We examine several variants of this model, and find that a variant of the model based on multiplicative interactions of non-optimized versions of these features achieves the best cross-validated prediction of the transfer test. This is substantially higher than the cross-validated correlation of Bayesian Knowledge Tracing, a measure of skill learning within the tutor software. Furthermore, we find that the transfer detector achieves near-asymptotic predictive power by the time the student has completed 20% of the tutor software, suggesting that the transfer detector can be used to drive intervention early enough to influence overall learning. Another potential use of future work is to investigate the degree to which the transfer detector correlates to other measures of robust learning, such as retention [cf. 15] and preparation for future learning [cf. 6], in order to improve understanding of how these constructs relate to one another. Overall, we view this detector as a potential early step towards intelligent tutors that can predict and respond automatically to differences in the robustness of student learning, an important complement to ongoing research on designing tutors that promote robust learning [e.g. 1, 7, 17].

Acknowledgements. This research was supported via grant “Empirical Research: Emerging Research: Robust and Efficient Learning: Modeling and Remediating Students’ Domain Knowledge”, National Science Foundation award number #DRL-0910188, via grant “Promoting Robust Understanding of Genetics with a Cognitive Tutor that Integrates Conceptual Learning with Problem Solving”, Institute of Education Sciences award number #R305A090549, and by the Pittsburgh Science of Learning Center, NSF award number #SBE-0836012.

References

1. Aleven, V., Koedinger, K.R.: An effective metacognitive strategy: learning by doing and explaining with a computer-based Cognitive Tutor. Cognitive Science 26, 147–179 (2002)
2. Aleven, V., McLaren, B., Roll, I., Koedinger, K.: Toward meta-cognitive tutoring: A model of help seeking with a Cognitive Tutor. International Journal of Artificial Intelligence and Education 16, 101–128 (2006)
3. Baker, R.S.J.d., Corbett, A.T., Gowda, S.M., Wagner, A.Z., MacLaren, B.M., Kauffman, L.R., Mitchell, A.P., Giguere, S.: Contextual Slip and Prediction of Student Performance After Use of an Intelligent Tutor. In: Proceedings of the 18th Annual Conference on User Modeling, Adaptation, and Personalization, pp. 52–63 (2010)

4. Baker, R.S.J.d., Corbett, A.T., Roll, I., Koedinger, K.R.: Developing a Generalizable Detector of When Students Game the System. User Modeling and User-Adapted Interaction 18(3), 287–314 (2008)
5. Baker, R.S.J.d., Goldstein, A.B., Heffernan, N.T.: Detecting the moment of learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 25–34. Springer, Heidelberg (2010)
6. Bransford, J.D., Schwartz, D.: Rethinking transfer: A simple proposal with multiple implications. Review of Research in Education 24, 61–100 (1999)
7. Butcher, K.R.: How Diagram Interaction Supports Learning: Evidence from Think Alouds during Intelligent Tutoring. In: Goel, A.K., Jamnik, M., Narayanan, N.H. (eds.) Diagrams 2010. LNCS, vol. 6170, pp. 295–297. Springer, Heidelberg (2010)
8. Corbett, A.T., Anderson, J.R.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. User Modeling and User-Adapted Interaction 4, 253–278 (1995)
9. Corbett, A., Bhatnagar, A.: Student Modeling in the ACT Programming Tutor: Adjusting Procedural Learning Model with Declarative Knowledge. In: User Modeling: Proceedings of the 6th International Conference, pp. 243–254 (1997)
10. Corbett, A.T., Kauffman, L., MacLaren, B., Wagner, A., Jones, E.: A Cognitive Tutor for Genetics Problem Solving: Learning Gains and Student Modeling. Journal of Educational Computing Research 42(2), 219–239 (2010)
11. Efron, B., Gong, G.: A leisurely look at the bootstrap, the jackknife, and cross-validation. American Statistician 37, 36–48 (1983)
12. Gong, Y., Beck, J.E., Heffernan, N.T.: Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 35–44. Springer, Heidelberg (2010)
13. Koedinger, K.R., Corbett, A.T., Perfetti, C.: (under review) The Knowledge-Learning-Instruction (KLI) Framework: Toward Bridging the Science-Practice Chasm to Enhance Robust Student Learning (manuscript under review)
14. Martin, J., VanLehn, K.: Student Assessment Using Bayesian Nets. International Journal of Human-Computer Studies 42, 575–591 (1995)
15. Pavlik, P.I., Anderson, J.R.: Using a Model to Compute the Optimal Schedule of Practice. Journal of Experimental Psychology: Applied 14(2), 101–117 (2008)
16. Pavlik, P.I., Cen, H., Koedinger, J.R.: Performance Factors Analysis – A New Alternative to Knowledge Tracing. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, pp. 531–540 (2009)
17. Salden, R.J.C.M., Koedinger, K.R., Renkl, A., Aleven, V., McLaren, B.M.: Accounting for Beneficial Effects of Worked Examples in Tutored Problem Solving. Educational Psychology Review 22, 379–392 (2010)
18. Shih, B., Koedinger, K.R., Scheines, R.: A response time model for bottom-out hints as worked examples. In: Proc. 1st Int'l Conf. on Educational Data Mining, pp. 117–126 (2008)
19. Shute, V.J.: SMART: Student modeling approach for responsive tutoring. User Modeling and User-Adapted Interaction 5(1), 1–44 (1995)
20. Singley, M.K., Anderson, J.R.: The Transfer of Cognitive Skill. Harvard University Press, Cambridge (1989)

Modeling Engagement Dynamics in Spelling Learning

Gian-Marco Baschera[1], Alberto Giovanni Busetto[1,2], Severin Klingler[1], Joachim M. Buhmann[1,2], and Markus Gross[1]

[1] Department of Computer Science, ETH Zürich
[2] Competence Center for Systems Physiology and Metabolic Diseases, Zürich, Switzerland
{gianba,busettoa,kseverin,jbuhmann,grossm}@inf.ethz.ch

Abstract. In this paper, we introduce a model of engagement dynamics in spelling learning. The model relates input behavior to learning, and explains the dynamics of engagement states. By systematically incorporating domain knowledge in the preprocessing of the extracted input behavior, the predictive power of the features is significantly increased. The model structure is the dynamic Bayesian network inferred from student input data: an extensive dataset with more than 150 000 complete inputs recorded through a training software for spelling. By quantitatively relating input behavior and learning, our model enables a prediction of focused and receptive states, as well as of forgetting.

Keywords: engagement modeling, feature processing, domain knowledge, dynamic Bayesian network, learning, spelling.

1 Introduction

Due to its recognized relevance in learning, affective modeling is receiving increasing attention. There are two reasons why modeling affective dynamics is considered a particularly challenging task. First, ground truth is invariably approximated. Second, experimental readouts and state emissions often exhibit partial observability and significant noise levels. This paper entertains the idea that intelligent tutoring systems can adapt the training to individual students based on data-driven identification of engagement states from student inputs.

Problem Definition. The goal of this study consists of modeling engagement dynamics in spelling learning with software tutoring. In our scenario, student input data and controller-induced interventions are recorded by the training software. Input behavior is assumed to be time- and subject-dependent.

Related Work. Affective models can be inferred from several sources: sensor data [1,2] and input data [3,4,6]. These sources differ in quality and quantity. On the one hand, sensor measurements tend to be more direct and comprehensive. They have the potential to directly measure larger numbers of affective features.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 31–38, 2011.

On the other hand, input measurements are not limited to laboratory experimentation. The measurement of student interaction with a software tutoring system offers a unique opportunity: large and well-organized sample sets can be obtained from a variety of experimental conditions. Recorded inputs have the potential to characterize the affective state of the student in a learning scenario. It has been shown that highly informative features, such as seconds per problem, hints per problem, and time between attempts, can be extracted from log files [6]. The identification of informative features and the incorporation of domain knowledge, either as implicit or as explicit assumptions, can substantially increment the predictive power of the inferred models [5]. Median splitting [6], thresholding [4], and input averaging [3] are conventional preprocessing techniques in affective modeling.

Contributions. We introduce a model which relates input behavior to learning, and explains the dynamics of engagement states in spelling training. We show how domain knowledge about dynamics of engagement can be incorporated systematically in the preprocessing of extracted input behavior to significantly increase their predictive power. The dynamic Bayesian network (DBN) is inferred from user input data recorded through a training software for spelling. Focused and receptive states are identified on the basis of input and error behaviors alone.

2 Methods

Our approach is articulated in four steps: (1) description of training process; (2) specification of extracted features; (3) feature processing based on domain knowledge; (4) feature selection and model building.

Learning Environment. The tutoring system consists of Dybuster, a multimodal spelling software for children with dyslexia [8]. During training, words are prompted orally and have to be typed in via keyboard by the student. As soon as incorrect letters are typed, an acoustic signal notifies the error. The system allows prompt corrections, which prevent the user from memorizing the erroneous input. Every user interaction is time-stamped and stored in log files.

Our analysis is based on the input data of a large-scale study in 2006 [9]. The log files span a time interval of several months, which permits the analysis of multiple time scales: from seconds to months. The German-speaking participants, aged 9-to-11, trained for a period of three months and with a frequency of four times a week, during sessions of 15-to-20 minutes. On average, each user performed approximately 950 minutes of interactive training. The training predominantly took place at home, except once per week, when the children attended a supervised session at our laboratory to ensure the correct use of the system. Due to technical challenges, a subset of 54 log files were completely and correctly recorded (28 dyslexic and 26 control). This dataset records 159 699 entered words, together with inputs, errors, and respective timestamps.

Feature Extraction. We identified a set of recorded features which are consistent with previous work [3,4,6]. Table 1 lists the features, which are evaluated for each word entered by the learner. The set contains measures of input and error behavior, timing, and variations of the learning setting induced by the system controller.

Engagement states are inferred from the repetition behavior of committed errors and without external direct assessments. We subscribe to the validated hypothesis of interplay between human learning and affective dynamics [7]. Committed errors and the knowledge state at subsequent spelling requests of the same word are jointly analyzed. Error repetition acts as a noisy indicator for learning and forgetting. We restrict the analysis on phoneme-grapheme matching (PGM) errors [12], which is an error category representing missing knowledge in spelling, in contrast to, e.g., typos. We extracted 14 892 observations of PGM errors with recorded word repetitions from the log files.

Feature Processing. The processing of continuous features is based upon the following central assumptions: emotional and motivational states come in spurts [4], and they affect the observed features on a short-to-medium time scale. Time scale separation enables a distinction between sustainable progress in the observed input behavior ($f(i)$) and other local effects ($p(x_i)$), such as the influence of engagement states. The terms are separated as

Table 1. Extracted features and abbreviations (bold) used in the following

Feature	**Description**
Timing	
Input **R**ate	Number of keystrokes per second.
Input **R**ate **V**ariance	Variance of seconds per keystroke.
Think **T**ime	Time from dictation of word to first input letter of student.
Time **f**or **E**rror	Time from last correct input letter to erroneous input letter.
Time **t**o **N**otice **E**rror	Time from error input letter to first corrective action.
Off **T**ime	Longest time period between two subsequent letter inputs.
Input & Error Behavior	
Help **C**alls	Number of help calls (repeating the dictation).
Finished **C**orrectly	True if all errors are corrected when enter key is pressed.
Same **P**osition **E**rror	True if multiple errors occur at one letter position of a word.
Repetition **E**rror	State of previous input of the same word (three states: *Correct / Erroneous / Not Observed).*
Error **F**requency	Relative entropy [10] from observed to expected error distribution (given by the student model [12]) over last five inputs. Positive values are obtained from larger errors numbers, negative values from smaller ones.
Controller Induced	
Time **t**o **R**epetition	Time from erroneous input to respective word repetition.
Letters **t**o **R**epetition	Number of entered letters from erroneous input to respective word repetition.

$$t(x_i) = f(i) + p(x_i), \tag{1}$$

with independent additive normal $p(x_i) \sim \mathcal{N}(0, \sigma^2)$. The transformation $t(\cdot)$ of the original feature x_i consists of scaling and outlier detection. The separation of long-term variation $f(i)$ depends on the temporal input position i in the student input history. The finally obtained additive terms $p(x_i)$ are referred to as processed feature. Table 2 lists the employed processor modules. Whereas scaling and outlier detection operate point-wise on the individual words, regression subtraction is time- and user-dependent. The selection of processing steps and corresponding coefficients for each feature are the result of a downhill simplex optimization of the differential entropy (with fixed variance) [13,11], resulting in a distribution of $p(x_i)$ with maximal normality. Figure 1 illustrates the processing of the Time for Error (TfE) feature. The low-pass and variance filters, listed in Table 2, allow for a separation of low frequency components from rapid fluctuations of the processed features and are tested in the feature selection step.

Feature Selection and Model Building. The relation between processed features $p(x_i)$ and error repetition γ_r is estimated via LASSO logistic regression [11] with 10-fold cross-validation for different filter and filter parameters. The regression parameters are denoted by b_i. Figure 2 illustrates the comparison between Error Repetition Probability (ERP) predictions obtained from unprocessed and processed features. The model based on processed features exhibits a better BIC score ($-6\,369$) compared to unprocessed regression ($-6\,742$). In the selected features (see Table 3), we identified three main effects influencing the knowledge state at the next repetition:

Table 2. Employed feature processing modules and abbreviations (bold)

Module	**Operation on feature** x	**Parameters**
Scaling		
Logarithmic	$\log(s + x)$	s
Exponential	$\exp(-\frac{a+x}{b})$	a, b
Splitting	$\mathrm{I}_{x>s}$	s
Outlier detection		
Deviation **C**ut	$\min(\mu + \sigma, \max(\mu - \sigma, x))$ $\mu = \text{mean}(x)$	σ
Regression subtraction		
Learn**ing C**urve	$x_i - f(i)$ $f(i) = a\exp(-bi) + c$	a, b, c
Filtering		
Low-**P**ass	$x_i = \sum_{j=0}^{n} x_{i-j} G(j, n)$ [1]	n
Variance	$x_i = \text{var}([x_{i-n}, ..., x_i])$	n

[1] $G(j, n)$ corresponds to the sampled Gaussian kernel $G(j, n) = \frac{1}{\sqrt{2\pi n}} e^{-\frac{j^2}{2n}}$.

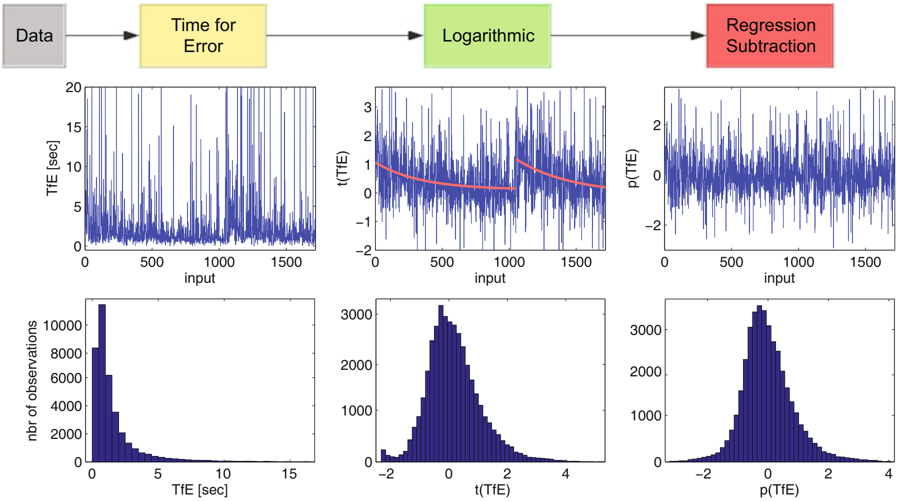


Fig. 1. Top line exemplifies the processing pipeline for the TfE feature. On the 2nd and 3rd row, signal and histogram plots show the processing steps for data recorded from two learners: extracted feature (left), transformation (center), and separation (right).

Focused state. indicates focused or distracted state of the student. In non-focused state more non-serious errors due to lapse of concentration occur, which are less likely to be committed again at the next repetition (lower ERP).

Receptive state. indicates the receptiveness of the student (receptive state or beyond attention span). Non-receptive state inhibits learning and causes a higher ERP.

Forgetting. the time (decay) and number of inputs (interference) between error and repetition induce forgetting of learned spelling and increase the ERP.

The parameters of the logistic regression indicate how features are related to the ERP. We inferred the affiliation of features to engagement states based on

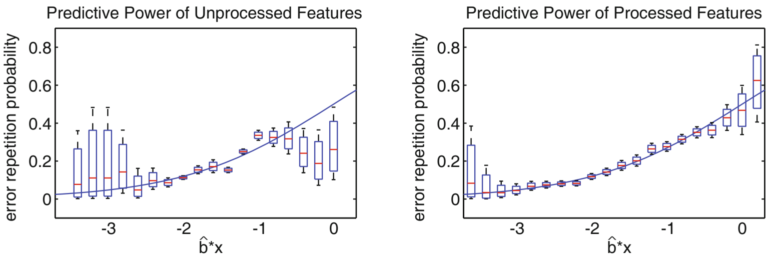


Fig. 2. ERP prediction (10-fold cross-validation) from unprocessed (left) and processed features (right). Predictions are plotted as blue curve and accompanied by mean (red stroke), 68% (box), and 95% confidence intervals (whisker) of the observed repetitions for bins containing at least 10 observations.

the relations extracted from the regression analysis and expert knowledge about desired input behavior. For example, the parameter $b = 0.06$ of EF demonstrates that a higher than expected error frequency is related to a lower ERP. This indicates that a student is non-focused and commits more but rather non-serious errors. On contrary, if a student does not finish an input correctly (FC = 0), the ERP increases ($b = -0.49$). This indicates that students, which are not correcting their spelling errors, are less likely to pick up the correct spelling.

In the following we investigate the mutual dependence of the two engagement states, which are considered as dynamic nodes. We compared three models: (1) based on a mutual independence assumption ($F \nleftrightarrow R$); (2) with dependence of focused state on receptivity ($F \leftarrow R$); (3) with dependence of receptivity on focused state ($F \rightarrow R$). The parameters of the DBN are estimated based on the expectation maximization (EM) algorithm implemented in Murphy's Bayes net toolbox [9]. The mutual dependence of the engagement states is inferred based on the estimated model evidence (BIC).

3 Results

Figure 3 presents the graphical model ($F \rightarrow R$) best representing the data with a BIC of $-718\,577$, compared to $-724\,111$ ($F \nleftrightarrow R$) and $-718\,654$ ($F \leftarrow R$). The relation between the Focused and Receptive state is illustrated by their joint probability distribution in Figure 4 (left). In a fully focused state, students are

Table 3. Optimal processing pipeline, estimated parameter b and significance for features selected by the LASSO logistic regression. Note that the exponential scaling inverts the orientation of a feature. The last two columns show the influence of the engagement states on the features modeled in the DBN: for binary nodes the probability p_1 of being *true*; for Gaussian nodes the estimated mean m of the distribution.

Feature	**Processing Pipeline**	**b**	**sig.**	**p_1[%]/m**	
Focused State				focused	non-f.
EF	Exp	0.06	2e-4	0.16	-0.34
IR	Log - DevC - LearnC - Var	-0.12	4e-6	-0.41	0.87
IRV	Log - DevC - LearnC	-0.22	2e-11	-0.36	0.78
REc		-0.28	8e-8	45%	32%
TfE	Log - DevC - LearnC - LowP	-0.50	1e-9	-0.13	0.28
Receptive State				receptive	non-r.
FC		-0.49	1e-7	95%	88%
HC	Split(zero/non-zero)	0.29	2e-4	4%	28%
OT	Log - DevC - LearnC - LowP	0.27	1e-9	-0.35	1.20
REe	LowP	0.20	1e-9	0.07	-0.24
TtNE	Exp - DevC - LearnC	-0.18	1e-5	0.11	-0.36
Forgetting					
TtR	Exp	-0.29	2e-8		
LtR	Log	0.34	1e-9		

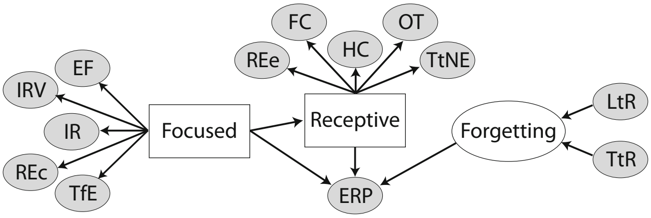


Fig. 3. The selected dynamic Bayesian net representation. Rectangle nodes denote dynamic states. Shaded nodes are observed.

never found completely non-receptive. In contrast, students can be distracted (non-focused) despite being in a receptive state.

The ERP conditioned on the two states is presented in Figure 4 (right). One can observe that the offset between top plane (forgetting) and bottom plane (no forgetting) is greater in the focused compared to the non-focused state. This underpins the assumption that in the non-focused state more non-serious errors are committed, of which the correct spelling is actually already known by the student. Therefore, the forgetting has a lower impact on their ERP. As expected, the non-receptive state generally causes a higher ERP. Again, this effect on learning is reduced for non-serious errors in the non-focused state. The estimated parameters of the conditional probability distributions for all the other observed nodes are presented in Table 3 (right).

The investigation of the age-dependence of engagement states shows that students below the median of 10.34 years exhibit a significantly ($p < 0.001$) higher probability of being classified as non-receptive (24.2%) and non-focused (32.5%) compared to those above the median (20.0% and 27.0%, respectively). This indicates that younger students tend to fall significantly more frequently into non-focused and non-receptive states.

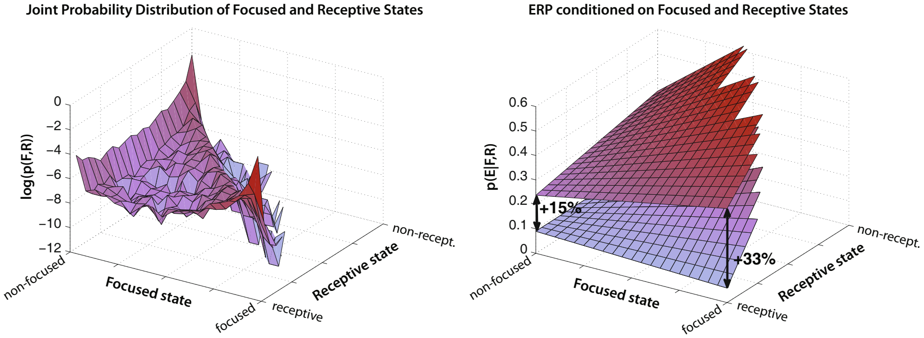


Fig. 4. Left: joint probability distribution of Focused and Receptive states. Right: ERP conditioned on engagement states for forgetting (top) and no forgetting (bottom plane). The ERP is plotted for all observed combinations of engagement states only.

4 Conclusion

We presented a model of engagement dynamics in spelling learning. We showed that domain knowledge can be systematically incorporated into data preprocessing to increase predictive power. In particular, the regression analysis demonstrates the advantages of feature processing for engagement modeling. Our approach enables the identification of the dynamic Bayesian network model directly from spelling software logs. The model jointly represents the influences of focused and receptive states on learning, as well as the decay of spelling knowledge due to forgetting. This core model can be extended with assessments of engagement of a different nature, such as sensor, camera or questionnaire data. This would allow to relate the identified states to the underlying fundamental affective dimensions (e.g., boredom, flow, confusion and frustration) of a student.

Acknowledgments. We would like to thank M. Kast and Kay H. Brodersen for helpful suggestions. This work was funded by the CTI-grant 8970.1.

References

1. Cooper, D.G., Muldner, K., Arroyo, I., Woolf, B.P., Burleson, W.: Ranking feature sets for emotion models used in classroom based intelligent tutoring systems. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 135–146. Springer, Heidelberg (2010)
2. Heray, A., Frasson, C.: Predicting Learner Answers Correctness through Brainwaves Assessment and Emotional Dimensions. In: AIED 2009, pp. 49–56 (2009)
3. Baker, R.S., Corbett, A.T., Koedinger, K.R.: Detecting student misuse of intelligent tutoring systems. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 531–540. Springer, Heidelberg (2005)
4. Johns, J., Woolf, B.: A Dynamic Mixture Model to Detect Student Motivation and Proficiency. In: UMAP 2006, pp. 163–168 (2006)
5. Busetto, A.G., Ong, C.S., Buhmann, J.M.: Optimized Expected Information Gain for Nonlinear Dynamical Systems. In: ICML 2009, pp. 97–104 (2009)
6. Arroyo, I., Woolf, B.: Inferring Learning and Attitudes from a Bayesian Network of Log File Data. In: AIED 2005, pp. 33–40 (2005)
7. Kort, B., Reilly, R., Picard, R.W.: An Affective Model of Interplay Between Emotions and Learning: Reengineering Educational Pedagogy - Building a Learning Companion. Advanced Learning Technologies, 43–46 (2001)
8. Gross, M., Vögeli, C.: A Multimedia Framework for Effective Language Training. Computer & Graphics 31, 761–777 (2007)
9. Kast, M., Meyer, M., Vögeli, C., Gross, M., Jäncke, L.: Computer-based Multisensory Learning in Children with Developmental Dyslexia. Restorative Neurology and Neuroscience 25(3-4), 355–369 (2007)
10. Kullback, S., Leibler, R.A.: On Information and Sufficiency. Annals of Mathematical Statistics 22(1), 79–86 (1951)
11. Bishop, C.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
12. Baschera, G.M., Gross, M.: Poisson-Based Inference for Perturbation Models in Adaptive Spelling Training. International Journal of AIED 20(1) (2010) (in press)
13. Nelder, J.A., Mead, R.: A Simplex Method for Function Minimization. Computer Journal 7, 308–313 (1965)
14. Murphy, K.: The Bayes Net Toolbox for Matlab. Computing Science and Statistics 33 (2001)

Assessment of Learners' Attention While Overcoming Errors and Obstacles: An Empirical Study

Lotfi Derbali, Pierre Chalfoun, and Claude Frasson

Département d'informatique et de recherche opérationnelle
Université de Montréal, 2920 Chemin de la Tour, Montréal, Canada

Abstract. This study investigated learners' attention during interaction with a serious game. We used Keller's ARCS theoretical model and physiological sensors (heart rate, skin conductance, and electroencephalogram) to record learners' reactions throughout the game. This paper focused on assessing learners' attention in situations relevant to learning, namely overcoming errors and obstacles. Statistical analysis has been used for the investigation of relationships between theoretical and empirical variables. Results from non-parametric tests and linear regression supported the hypothesis that physiological patterns and their evolution are suitable tools to directly and reliably assess learners' attention. Intelligent learning systems can greatly benefit from using these results to enhance and adapt their interventions.

Keywords: Learners' attention, assessment, serious game, physiological sensors, EEG, regression model.

1 Introduction

The increased use of Computer-Based Education over the last decades encourages the investigation of new and engaging learning environments. Currently, serious games are used to train or educate learners while giving them an enjoyable experience. They have been considered as the next wave of technology-mediated learning. Several studies have assessed their potential as learning tools [1-3]. They have concluded that the integration of games into learning systems have enhanced the desired learning outcomes. Amory and colleagues have identified game elements that learners found interesting or useful within different game types such as in-game rules, immersive graphical environment and interactivity just to name a few [4]. Beside these distinctive design elements that seem necessary to stimulate learners' motivation, other researchers however have reported that consequences of different game elements, such as risks, errors, and obstacles, seem to be more relevantly correlated with learners' motivation and attention [5]. Indeed, computer games typically put traps and obstacles in the way of the player thus requiring more attention while overcoming them in order to properly progress through the rest of the game. In contrast to the significant amount of research effort in the area of serious games, less has been done however regarding the assessment of learners' attention while overcoming errors and obstacles. These situations are therefore the specific area of interaction that our research focuses on to assess learners' attention.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 39–46, 2011.

Traditional assessment usually involves measuring performance, time spent, and response time as main indicators of learners' attention. However, researchers have recently used psychological motivational models, physical sensors, and a combination of both to assess complex learners' states such as motivation, attention and engagement during serious game play [2, 6, 7]. Nevertheless, further studies are required to possibly identify learners' physiological evolution while overcoming errors and obstacles. Consequently, a learning system can customize its learning process by adapting learning strategies in order to respond intelligently to learners' needs, objectives and interests. The present paper aims at highlighting some of the relevant physiological patterns that occur in learners and correlating them with learners' attention while overcoming errors and obstacles in a serious game. We have therefore carried out an empirical study to assess learners' attention using Keller's ARCS psychological model combined with physiological recordings, namely heart rate (HR), skin conductance (SC) and brainwaves (EEG). We ask in this paper the two following research questions: can we identify relevant physiological manifestations in learners' attention while overcoming errors and obstacles? If so, can we reliably predict learners' attention by establishing a reliable AI model?

The organization of this paper is as follows: in the next section, we present previous work related to our research. In the third section, we explain our empirical approach in assessing learners' attention. In the fourth section, we detail our experimental methodology. In the fifth section, we present the obtained results and discuss them, in the last section, as well as present future work.

2 Related Work

Assessing learners' states is of particular importance in establishing proper strategies and understanding the processes that might explain differences between learners' knowledge acquisition. Unlike human tutors, intelligent systems cannot exclusively rely on observational cues, such as posture and gesture, to infer emotional and cognitive states, such as motivation and engagement. Several studies have been therefore proposed to intelligently identify these states through the use of physical sensors. One of those studies used biometric sensors (HR, SC, electromyography and respiration) and facial expression analysis to develop a probabilistic model of detecting students' affective states within an educational game [8]. Another study used four different sensors (camera, mouse, chair, and wrist) in a multimedia adaptive tutoring system to recognize students' affective states and embedded emotional support [9].

In the particular case of learners' attention, performance and response time have been generally used as assessment metrics. Recent studies have reported significant results in assessing learners' attention using others cues. Qu and colleagues for example used a Bayesian model to combine evidence from the learner's eye gaze and interface actions to infer the learner's focus of attention [10]. Kuo-An and Chia-Hao applied fuzzy logic analysis of students facial images when participating in class to prevent erroneous judgments and help tutors deal with students attentiveness [11]. Another multimodal approach by Peters and colleagues investigated a user attention model by establishing three core components (gaze detection, neurophysiological detection, and a user attention representation module) for human-agent interaction [12]. The authors proposed to establish patterns of behavior and attention allocation useful for endowing autonomous agents with more natural interaction capabilities. Finally, in video games context, commercial helmet-embedded sensors combining multiple channels such as EEG and facial EMG, have been designed to recognize game relevant player states, such as engagement, attention, and boredom [13].

It is clear that combining physical sensors and theoretical model is best for addressing learners' attention during a specific activity or context. However, this combination has been rarely done. For example, [14] used a self-report questionnaire (Keller's ARCS model) and a portable EEG to examine attention and motivation in a virtual world. We also aim in this work to examine the attention state but in a completely different goal and perspective. Indeed, in contrast to Rebolledo-Mendez and colleagues' work, we have chosen to assess learners' attention while overcoming errors and obstacles during serious game play. We have also chosen to combine the ARCS model and different physiological sensors (HR, SC, and EEG) as our assessment metrics. We chose serious games for they constitute a powerful learning environment to support attention and motivation [2]. They can accelerate learning and support the development of various skills, such as cognitive thinking and problem solving skills [15]. In fact, many studies have been increasingly trying to define specific features of games that enhance learning [1, 2, 16]. They have stated that these environments increase attention state through the use of traps and obstacles to allow learners for instance to take risks and overcome obstacles. We are interested in assessing this specific state in learners and the next section will present the method used to assess learners' attention.

3 Assessment of Learners' Attention

The key issue in this paper is related to the assessment of learners' attention in serious games environment. The *Attention* category of the ARCS model of motivation [17] has been chosen to theoretically assess learners' attention. Indeed, Keller's model is of particular interest in our study since it separately considers the attention dimension and it has been used in learning, training and games [18]. Even the use of a theoretical model may offer some insight into the learners' attention directly from the learners but it remains insufficient. Several objective measures, however, are not dependent on a learners' perception and generally include independent measures such as performance, time spent in a game, response time, and physiological reactions. In our empirical assessment approach, we decided to assess learners' attention by using non-invasive physiological sensors (SC and HR). These sensors are typically used to study human affective states [19]. Furthermore, we decided to add another interesting and important sensor: EEG. Indeed, brainwave patterns have long been known to give valuable insight into the human cognitive process and mental state[20] .

This paper also explores the intricate relationship between the *Attention* category in the ARCS model and its corresponding EEG fingerprint expressed in the form of a ratio known as the attention ratio (Theta/Low-Beta) [21]. Indeed, according to the authors, a negative correlation exists between the attention ratio and learners' attention. A high attention ratio is usually correlated with excessive Theta and consequently inattentive state. Conversely, a low attention ratio is normally correlated with excessive Low-Beta brainwave activity reflecting attentive state in adults. In addition, it is common knowledge throughout the neuro-scientific community that investigations of cerebral activity limited to one area of the brain may offer misleading information regarding complex states such as attention. We have therefore investigated different cerebral areas to study simultaneous brainwave changes. The idea is to analyze, in a joint venture, both physiological and cerebral signals to determine, or at least estimate, their correlations with learners' attention while overcoming errors and obstacles during serious game play. To that end, prediction models will be constructed using theoretical and empirical data. A detailed description of all these possibilities is given in the experiment and results sections.

4 Experiment

The participants were invited to play the serious game called FoodForce from the World Food Program of the United Nations intended to educate players about the problem of world hunger. FoodForce is comprised of multiple arcade-type missions, each intended at raising players' awareness towards specific problems regarding world-wide food routing and aid. FoodForce also presents players' objectives in a short instructional video before the beginning of each mission. A virtual tutor also accompanies the player throughout each mission by offering various tips and lessons relative to the obstacles and goals at hand. Following the signature of a written informed consent form, each participant was placed in front of the computer monitor to play the game. A baseline was also computed before the beginning of the game. A pre-test and post-test were also administered to compare learners' performance regarding the knowledge presented in the serious game.

The missions we are interested in investigating in this paper are missions 3 and 5. Mission 3 instructs players to drop 10 food packets from an airplane to an alley on the ground. Before dropping a packet, a player has 5 seconds to calculate the speed and strength of the wind before releasing the food ideally as close to the center of the lane as possible. *Errors* in this mission are reflected through the obtained final score. Furthermore, the tutor intervened and gives an immediate feedback after each drop. Mission 5 is concerned with driving food trucks in dangerous territories and get through *obstacles* such as quickly replacing flat tires and managing diplomatically through intimidation attempts by angry locals. Players loose one truck of food for each failed attempt to successfully overcome an obstacle in this mission.

The motivational measurement instrument called Instructional Materials Motivation Survey IMMS [17] was used following each mission to assess learners' motivational state. SC and HR sensors were attached to the fingers of participants' non-dominant hands, leaving the other free for the experimental task. An EEG cap was also conveniently fitted on learners' heads and each sensor spot slightly filled with a proprietary saline solution. EEG was recorded by using a cap with a linked-mastoid reference. The sensors were placed on three selected areas (F3, C3 and Pz) according to the international 10-20 system. The EEG was sampled at a rate of 256 Hz. A Power Spectral Density (PSD) was computed to divide the EEG raw signal into the two following frequencies: Theta (4-8 Hz) and Low-Beta (12-20 Hz) in order to compute the attention ratio (Theta/Low-Beta) as described above. To reduce artifacts, participants were asked to minimize eye blinks and muscle movements during recording. A normalization technique (min-max) was applied to all physiological data.

We computed an index representing players' physiological evolution throughout the mission with regards to each signal signification. This index, called Percent of Time (PoT), represents the amount of time, in percent, that learners' signal amplitude is lower (or higher) than a specific threshold. The threshold considered for each signal is the group's signal average for each mission. The PoT index is a key metric enabling us to sum-up learners' entire signal evolution for a mission. For SC and HR, the PoT index will be computed for values *above* the threshold since we are looking for positive evolutions when playing a serious game. Conversely, for EEG attention ratios, a PoT index was calculated when learner's attention was *below* the threshold as explained previously in section 3. Fig. 1 illustrates a learner's EEG attention ratio evolution during 20 seconds one mission. The computed PoT for the selected 5 second window in this figure would be 40% (2 values below divided by 5 values) and 80% for the entire 20 seconds (16 values below divided by 20 values overall).

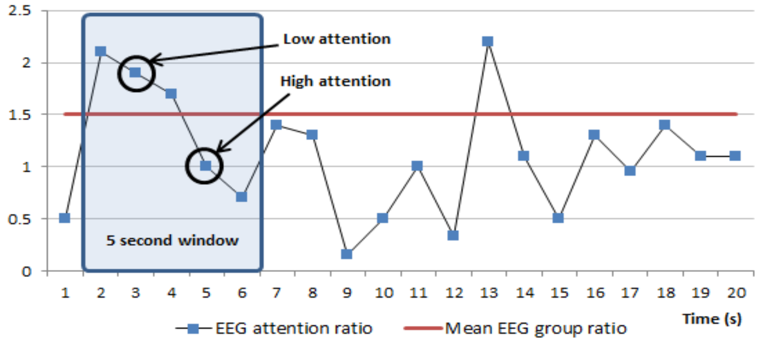


Fig. 1. Learner's attention ratio evolution

Thirty three volunteers (11 female) took part in the study in return of a fixed compensation. Participant's mean age was 26.7 ± 4.1 years. Four participants (2 female) were excluded from the EEG analysis because of technical problems at the time of recording. The next section will detail the experimental results and findings.

5 Results

Our statistical study relied on non-parametric statistical tools because our sample population is small (29 participants) and no justifiable assumptions could be made with regards to the normal distribution of the data. Hence, Wilcoxon signed ranks test and Spearman's rho ranks test have been used. Furthermore, reported significant p-values were all computed at the .05 significance level (95% confidence).

First, we report significant positive change regarding learners' knowledge acquisition. Indeed, we administered pre-tests and post-tests questionnaires pertaining to the knowledge taught in the serious game and compared results using the Wilcoxon signed ranks test ($Z = 4.65$, $p < 0.001$).

Second, we report results of correlation run on data of missions 3 and 5. Analysis of mission 3 showed that a significant relationship between reported attention and three physiological sensors (*PoT-F3 index: spearman's rho=.34, n=29, p<.001; PoT-SC index: spearman's rho=.536, n=29, p<.01; PoT-C3 index: spearman's rho=.532, n=29, p<.01*). Similar results have been found for reported attention regarding mission 5, *except* for the PoT-F3 index (*PoT-C3 index: spearman's rho=.62, n=29, p<.01; PoT-SC index: spearman's rho=.503, n=29, p<.01*). These results positively answer our first research question (can we identify relevant physiological manifestations in learners' attention while overcoming errors and obstacles?). Indeed, learners' attention while overcoming errors and obstacles can reliably be monitored and related to changes in skin conductance and F3 and C3 EEG sensors.

Third, in order to answer our second research question (can we reliably predict learners' attention by establishing a reliable AI model?), we ran linear regressions to predict learners' reported attention during each mission. Our prediction models used all computed PoT indexes and learners' mission final score as predictor variables (PoT-SC, PoT-HR, PoT-F3, PoT-C, PoT-Pz and Score) and the stepwise method for variable selection. Table 1 reports the results of multiple linear regressions.

Table 1. Results of regression models

Regression model	**F**	**Sig.**	**Adjusted R^2**	**Significant predictors**		
Mission 3	$F_{2,26}$=18.304	.000 (*)	.553	PoT-F3:	Beta=.560	p=.000
				PoT-SC:	Beta=.437	p=.002
Mission 5	$F_{1,27}$=28.409	.000 (*)	.495	PoT-C3:	Beta=.716	p=.000

(*) Significance at the 0.05 level

In our prediction models, EEG attention ratios are significant predictors for attention for the duration of both missions. These results seem to show the relevance and importance of adding the EEG in assessing learners' attention evolution, even more so when attention cannot be clearly established by the use of HR and SC alone. Furthermore, our AI model is sensitive to the type of mission as well as the time window for assessment. Indeed, a described earlier in section 4, mission 3 and mission 5 involve different skills from a learner that are represented by changes in F3 and C3 respectively [22]. During mission 3, while trying to avoid errors and mistakes as much as possible, learners will tend to rely mostly on the frontal cortex (F3) because it is known to be strongly implicated in taking quick decisions under pressure. Conversely, during mission 5, while trying to overcome obstacles, a more "generalized" problem-solving approach is used and thus the central region of the brain (C3) seems to be the most solicited. An example of this situation is illustrated in fig. 2.

This figure presents PoT index evolutions of 3 learners (TOP, BOTTOM and LEARNER 17) in 2 distinct moments: the beginning (*Start*) and the end (*End*) of the mission. The blue filled bar (Top) represents a learner whose reported attention is highest for both missions. Conversely, the brown horizontal sprites (Bottom) represent a learner whose reported attention is lowest. Learner 17 (the gray diagonal sprites) has reported a very low attention in mission 3 but a very high attention in mission 5. We can see by the results that the predictors found in the model for mission 3 (PoT-F3 and PoT-SC) can distinguish between learners with high versus low attention. Learner 17 has the same trends (PoT-F3 and PoT-SC) as the bottom learner. Conversely, the predictor found in the model for mission 5 (PoT-C3) is the one to look at in order to separate learners' attention. Again we can clearly see that learner 17's PoT-C3 trend is almost the same at the top learner.

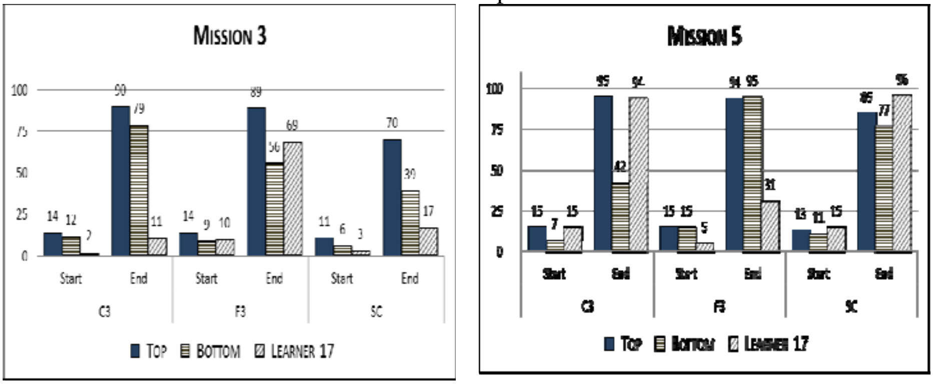


Fig. 2. PoT index evolution for missions 3 and 5

6 Conclusion and Future Work

In this paper, we have assessed learners' attention while overcoming errors and obstacles in a serious game using the ARCS theoretical model as well as three objective physiological measures: HR, SC and EEG. Results have shown that learners' attention was correlated with specific physiological manifestations, especially observable in the evolution of the EEG PoT indexes (C3 and F3). We have also built significant regression models that have shown to be valuable tools in predicting learners' attention using physiological patterns' evolution for each mission.

The obtained results are very encouraging to their future integration in an adaptive real-time attention detection prototype for an intelligent learning system. This integration will positively contribute to learning because reliable real-time objective assessment of learners' attention is now possible, since we can rely on this assessment as a substitute for self-reports that can disrupt a learning session. Furthermore, it is possible to enrich an intelligent system to properly adapt its interventions during a specific activity or context based on task type. However, one possible limitation of this study is the dependence of all categories of the ARCS model. In further work, we plan to address a complementary study in order to highlight other distinctive, or even common, physiological patterns related to other ARCS categories (relevance, confidence, and satisfaction) and the overall motivational state of the learner.

Acknowledgments

We thank the CRSNG, the FQRSC and the Tunisian Government for their support.

References

1. Garris, R., Ahlers, R., Driskell, J.E.: Games, motivation, and learning: a research and practice model. Simulation & Gaming 33, 441–467 (2002)
2. Prensky, M.: Digital Game-Based Learning. McGraw Hill, New York (2001)

3. 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)
4. Amory, A., Naicker, K., VIncent, J., Adams, C.: The use of computer games as an educational tool: identification of appropriate game types and game elements. British Journal of Educational Technology 30, 311–321 (1999)
5. Dondlinger, M.J.: Educational video game design: A review of the literature. Journal of Applied Educational Technology 4, 21–31 (2007)
6. Derbali, L., Frasson, C.: Prediction of Players Motivational States Using Electrophysiological Measures during Serious Game Play. In: IEEE International Conference on Advanced Learning Technologies, Sousse, Tunisia, pp. 498–502 (2010)
7. Gunter, G.A., Kenny, R.F., Vick, E.H.: A case for a formal design paradigm for serious games. International Digital Media & Arts Association 3, 93–105 (2006)
8. Conati, C.: Probabilistic Assessment of User's Emotions in Educational Games. Applied Artificial Intelligence 16, 555–575 (2002)
9. Arroyo, I., Cooper, D.G., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion Sensors Go To School. In: International Conference on Artificial Intelligence in Education, Brighton, UK, pp. 17–24 (2009)
10. Qu, L., Wang, N., Johnson, W.L.: Using Learner Focus of Attention to Detect Learner Motivation Factors. User Modeling, 70–73 (2005)
11. Kuo-An, H., Chia-Hao, Y.: Attentiveness Assessment in Learning Based on Fuzzy Logic Analysis. Intelligent Systems Design and Applications, 142–146 (2008)
12. Peters, C., Asteriadis, S., Rebolledo-mendez, G.: Modelling user attention for human-agent interaction. In: 10th Workshop on Image Analysis for Multimedia Interactive Services, pp. 266–269 (2009)
13. Hudlicka, E.: Affective game engines: motivation and requirements. In: 4th International Conference on Foundations of Digital Games, Orlando, Florida, pp. 299–306 (2009)
14. Rebolledo-Mendez, G., de Freitas, S., Rojano-Caceres, R., Gaona-Garcia, A.R.: An empirical examination of the relation between attention and motivation in computer-based education: A modeling approach. In: International Florida Artificial Intelligence Research Society Conference, Florida, USA, pp. 74–79 (2010)
15. Freitas, S.d., Oliver, M.: How can exploratory learning with games and simulations within the curriculum be most effectively evaluated? Computers & Education 46, 249–264 (2006)
16. McNamara, D.S., Jackson, G.T., Graesser, A.C.: Intelligent tutoring and games (iTaG). In: Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, pp. 1–10. IOS Press, Brighton (2009)
17. Keller, J.M.: Development and use of the ARCS model of motivational design. Instructional Development 10, 2–10 (1987)
18. Keller, J.M.: Motivational design for learning and performance: The ARCS model approach. Springer, New York (2010)
19. Lin, T., Imamiya, A., Hu, W., Omata, M.: Display Characteristics Affect Users' Emotional Arousal in 3D Games. Universal Access in Ambient Intelligence Environments, 337–351 (2007)
20. Wilson, G.F., Fisher, F.: Cognitive task classification based upon topographic EEG data. Biological Psychology 40, 239–250 (1995)
21. Putman, P., van Peer, J., Maimari, I., van der Werff, S.: EEG theta/beta ratio in relation to fear-modulated response-inhibition, attentional control, and affective traits. Biological Psychology 83, 73–78 (2010)
22. Demos, J.N.: Getting started with neurofeedback. Norton & Compagny, New York (2005)

Lattice-Based Approach to Building Templates for Natural Language Understanding in Intelligent Tutoring Systems

Shrenik Devasani[1], Gregory Aist[1], Stephen B. Blessing[2], and Stephen Gilbert[1]

[1] VRAC, Iowa State University, 1620 Howe Hall, Ames, IA 50011, USA
[2] University of Tampa, 401 W. Kennedy Blvd., Tampa, FL 33606, USA
{shrenik,aist,gilbert}@iastate.edu, sblessing@ut.edu

Abstract. We describe a domain-independent authoring tool, ConceptGrid, that helps non-programmers develop intelligent tutoring systems (ITSs) that perform natural language processing. The approach involves the use of a lattice-style table-driven interface to build templates that describe a set of required concepts that are meant to be a part of a student's response to a question, and a set of incorrect concepts that reflect incorrect understanding by the student. The tool also helps provide customized just-in-time feedback based on the concepts present or absent in the student's response. This tool has been integrated and tested with a browser-based ITS authoring tool called xPST.

Keywords: natural language processing, intelligent tutoring system, authoring tool.

1 Introduction

Interpreting textual responses from students by an Intelligent Tutoring System (ITS) is essential if it can come close to matching the performance of a human tutor, even in domains such as Statistics and Physics, since the use of language makes the learning process more natural. Natural language has the advantage of being easy to use for the student, as opposed to learning new formalisms.

Over the past decade, studies have been conducted that confirm the importance of using language in both traditional learning environments and in intelligent tutoring systems. Chi et al. [1, 2] have showed that eliciting self-explanations enhances deeper learning and understanding of a coherent body of knowledge that generalizes better to new problems. Aleven et al. [3] conducted studies with the PACT Geometry Tutor in which students who provided explanations to solution steps showed greater understanding in the post-test, compared to students who did not provide explanations.

Many ITSs have successfully incorporated natural language processing. The CIRCSIM Tutor [4] is a language based ITS for medical students that uses word matching and finite state machines to process students' natural language input. Rus et al. [5] have described an approach of evaluating answers by modeling it as a textual entailment problem. Intelligent tutoring systems such as the AutoTutor [6] and Summary Street [7] use Latent Semantic Analysis (LSA) [8] to evaluate student

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 47–54, 2011.

answers, a technique that uses statistical computation and is based on the idea that the aggregate of all the word contexts in which a word appears determines the similarity of meaning of words to each other. The problem with LSA is that it does not encode word order and it cannot always recognize negation. Another problem with LSA is that it scores students' responses only based on how well it matches the ideal answer, and cannot point out what exactly is wrong with an incorrect response.

Though ITSs today use a variety of techniques to provide support for natural language understanding, user-programming of NLP in ITSs is not common with authoring toolkits. The various techniques described here do not give sufficient power to non-programmers as the NLP is left to expert developers or to machine learning algorithms, and the user is more likely to focus on tutoring strategies. Our approach addresses these issues.

2 The ConceptGrid Approach

ConceptGrid is intended to be used by tutor-authors with little or no programming experience. The most crucial aspect about developing an authoring tool that can be used by non-programmers is managing the trade-off between its ease of use and its expressive power. Keeping this in mind, ConceptGrid has been designed such that its ease of use and expressiveness lie between that of simple word matching approaches and complex approaches such as those that use complex machine learning algorithms.

The tutor-author develops the natural language understanding component for a tutor by breaking down the expected response to a question into specific concepts. The author then builds templates that describe a set of required concepts (that are meant to be a part of student's response to a question) and a set of incorrect concepts (that reflect incorrect understanding by the student). Every template is mapped to a single user-defined concept name. Since a student can describe a single concept in various forms, several templates can be used to describe different representations of a single concept, in order to recognize and provide feedback to a wider range of student responses (both correct and incorrect). Thus, there is a one-to-many relationship between concepts and templates.

A template consists of one or more atomic checktypes, or check functions, that evaluate a student's input. These particular atomic checktypes are based on well-known algorithms and distance measures. The word "atomic" refers to the fact that these checktypes can be applied to a single word only. The set of atomic checktypes have been described in Table 1.

Apart from these atomic checktypes, we have two more checktypes that help make the template more expressive: Any(n_1, n_2) and Not(n, 'direction', word_list). The checktype "Any" matches any sequence of words that is at least n_1 words long and at most n_2 words. It helps account for words that are not explicitly accounted for using the other checktypes. The "Not" checktype takes care of negation. It makes sure that the n words appearing to the left or right (specified by 'direction') of the word following the checktype do not match the words mentioned in "word_list".

Table 1. Atomic checktypes used in designing a template

Checktype	Description
Exact(word_list)	Returns true if a literal character-by-character word match with any of the words in word_list is found
Almost(word_list)	Returns true if a literal match, after ignoring vowels, with any of the words in word_list is found
Levenshtein(n, word_list)	Returns true if the least Levenshtein distance between a word in word_list and matched word is <= n
Hamming(n, word_list)	Returns true if the least Hamming distance between a word in word_list and matched word is <= n
Soundex(word_list)	Returns true if a Soundex match with any of the words in word_list is found
Synonym(word_list)	Returns true if an exact match with any of the words in word_list or its synonyms (from WordNet) is found
Stemmer(word_list)	Returns true if a literal match with the stem of the matched word, with any of the words in word_list is found (uses Porter Stemmer)

The checktypes Synonym and Stemmer can be nested within other atomic checktypes to make them more powerful. Levenshtein(Synonym('interface'),1), for example, captures the idea that any synonym of the word "interface" is fine, even if it has a spelling mistake.

When the student misses out on a subset of the required concepts, or mentions a subset of incorrect concepts, customized feedback can be given that points out the issue.

3 The ConceptGrid Interface

The web-based interface is designed to allow the tutor-author to create templates that describe both required and incorrect concepts, and mention the customized just-in-time feedback that needs to be given to the students.

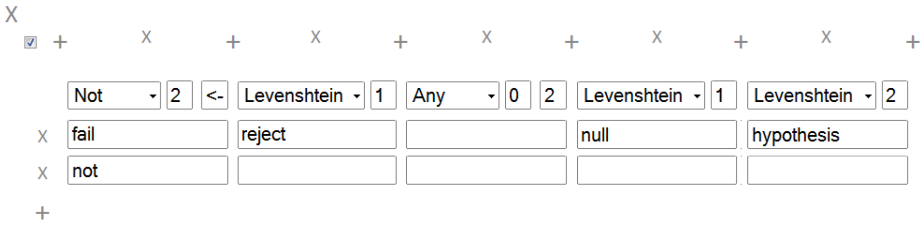
To simplify the process of constructing templates, we have a lattice-style table-driven interface for entering the template's checktypes and the corresponding parameters (Figure 1). A new template is created either by entering the dimensions of the table or by entering a sample response, from which a table is created and initialized. The table consists of a sequence of multi-level drop-down menus that represent the checktypes. The multiple levels help the author nest different checktypes. Each drop-down menu is associated with a specific number of textboxes that store the parameters associated with it. Each drop-down menu has several textboxes below it that store the contents of the parameter "word_list" associated with the corresponding checktype. The contingent approach of having the parameters dependent on the specific checktype provides a mild form of just-in-time authoring help. The user can navigate through the table just like a numerical spreadsheet and add or delete new rows and columns.

There are two sets of templates; the first describes required concepts and the second describes incorrect ones. Multiple templates can be mapped onto a single concept. Consider the following question in a statistics problem: “Based on your results, what do you conclude about the conditions of the music?” Let us assume that the correct answer to the question is "Reject the null hypothesis. There is a significant difference in memory recall between the rock music and no music conditions."

Some of the concepts that can be defined for the sample response mentioned above are described in Table 2.

Table 2. Examples of concepts. Conclusion-Correct and Conclusion-Incorrect look at the holistic response and the rest look at the sub-components of the response.

Concept Name	Description
Rejection-Correct	Matches responses that correctly mention whether the null hypothesis has to be rejected or not
Rejection-Incorrect	Matches responses that incorrectly mention whether the null hypothesis has to be rejected or not
Significance-Correct	Matches responses that correctly mention the significance of the result of the statistical test
Significance-Incorrect	Matches responses that incorrectly mention the significance of the result of the statistical test
Ind-Variable-Mention	Matches responses that explicitly mention the independent variable (e.g. type of music)
Dep-Variable-Mention	Matches responses that explicitly mention the dependent variable (e.g. memory recall)
Conclusion-Correct	Matches responses that have the correct conclusion of the statistical test
Conclusion-Incorrect	Matches responses that have the incorrect conclusion of the statistical test

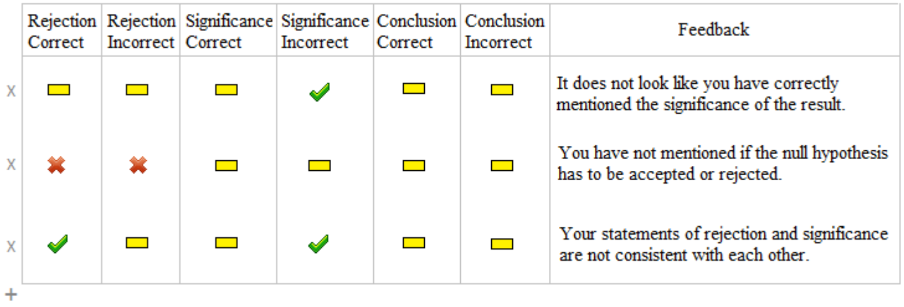


Not ▾ 2 <-	Levenshtein ▾ 1	Any ▾ 0 2	Levenshtein ▾ 1	Levenshtein ▾ 2
fail	reject		null	hypothesis
not				

Fig. 1. The lattice-style table-driven interface of ConceptGrid. The template represents the concept “Rejection-Correct”, described in Table 2.

The tutor-author then can design a ternary truth table called the Feedback Table (Figure 2) where he or she can enter the feedback that is to be given to the students, based on the truth values of the concepts: true – concept present (green check), false – concept absent (red X), or don’t care (yellow dash). The author enters the values of the truth table through tri-state checkboxes. Feedback can be entered for both the absence of required concepts and presence of incorrect ones.

The Feedback Table helps provide feedback in a simple manner for seemingly complicated issues, such as an inconsistent statement (the last row of the Feedback Table in Figure 2) in the example discussed.



	Rejection Correct	Rejection Incorrect	Significance Correct	Significance Incorrect	Conclusion Correct	Conclusion Incorrect	Feedback
x	▬	▬	▬	✓	▬	▬	It does not look like you have correctly mentioned the significance of the result.
x	✗	✗	▬	▬	▬	▬	You have not mentioned if the null hypothesis has to be accepted or rejected.
x	✓	▬	▬	✓	▬	▬	Your statements of rejection and significance are not consistent with each other.
+							

Fig. 2. Feedback Table

There is a provision to create user-defined variables that can be used while building checktypes or mentioning the feedback. This approach helps re-use templates for similar questions. The author can also enter a set of stop words that will be filtered out from the student's response prior to being processed.

Once the templates are designed and the feedback tables are filled, the author can test the templates with sample student responses. The output of the test mentions if the student's response has matched the required concepts. If a match is not found, then it displays the feedback associated with that response. It also displays the truth values of all the concepts defined by the author.

4 Algorithm and Implementation

The implicit sequencing in the lattice approach means that the resulting complex checktypes are finite parsers. That is, progress through the lattice corresponds to progress left-to-right in processing the input.

The templates are represented internally as and-or trees. The algorithm involves a combination of recursion and memoization to efficiently process the input. Since the algorithm might need to backtrack many times, memoization helps speed up the processing by having function calls avoid repeating the calculation of results for previously processed inputs.

Our tool has been integrated with the Extensible Problem Specific Tutor (xPST) - an open source authoring tool that is intended to enable non-programmers to create ITSs on existing websites and software [9]. Though xPST is a text-based authoring tool, its syntax is not very-code like. ConceptGrid has been customized to generate "code" that is compatible with xPST's syntax, based on the author's templates and Feedback Table, which can be then be inserted into any xPST file.

5 Results: The xSTAT Project

The research question for this paper is whether ConceptGrid could enable an instructor to create a tutor that would score students' free response answers as accurately as he or she manually did. At this point, the question is purely a feasibility issue: can it be done with the ConceptGrid tool? We tested this issue as a part of the xSTAT project at University of Tampa, dedicated to developing an intelligent homework helper for statistics students [10].

For the xSTAT effort, six authors (3 instructors and 3 undergraduates) created multiple tutors each for college level statistics problems. The problems contained real-world scenarios with actual data, followed up by several questions for the student to answer. Each of the problems had a question at the end that asked students to enter the conclusion of the statistics test. To assess these problems, 6 were chosen out of the total pool of 74 and given to students as homework problems. All problems were solved on-line using a standard web browser. Half of the students received feedback on their answers via the xPST intelligent tutor (i.e., answers were marked as either correct or incorrect, and hints and just-in-time messages were displayed), and half did not (i.e., these students simply filled out the web-based form). It is worth noting that these tutors were created without ConceptGrid, so that authors had to explicitly enter the "xPST code" that represents the templates without a graphical user interface. Also, in the absence of visualization through the Feedback Table, subsets of missing and incorrect concepts had to be explicitly mentioned. This non-lattice approach was not very usable by non-programmers. This difficulty motivated the creation of the ConceptGrid lattice approach, which is computationally equivalent and designed to be much more usable by non-programmers.

In all, 41 students solved a total of 233 instances of the six problems across the homework. We built a corpus after collecting all student responses to the end question (both those with tutoring and without). The corpus had 554 unique responses to this final conclusion question across the six homework problems. This corpus includes multiple incorrect responses by the same student to the same problem if they were in the tutored condition.

These responses were manually scored by an instructor and a teaching assistant based on the presence or absence of the concepts defined in Table 2. Then, a tutor-author attempted to use ConceptGrid to produce templates that would score the 554 responses similar to those manual scores. The result of that work contained a total of 10 templates common to all six problems, to cover all concepts, except “Ind-Variable-Mention” and “Dep-Variable-Mention”. The concepts “Ind-Variable-Mention” and “Dep-Variable-Mention” required a template each that was unique to each of the six problems. In all, there were 22 templates across all six problems. A template, on an average consisted of 4 checktypes.

Since the manner in which a template tries to match a student’s response – a sequence of words is comparable to the manner in which a regular expression matches a string, it might seem that the results have a lot of false negatives. But, since this approach tries to "understand" responses by looking for smaller concepts and key phrases with the help of checktypes rather than literal word matching, it is much more expressive. The results in Table 3, where we report the number of false positives, false negatives and the accuracy, the fraction of correct classifications, confirm this observation. The last column shows the values of Unweighted Kappa, which is a measure of the degree to which the human grader and ConceptGrid concur in their respective classifications.

Table 3. Results of the classification of 554 student responses using ConceptGrid

Concept	False Positives	False Negatives	Accuracy	Kappa
Rejection-Correct	1	34	0.9368	0.8657
Rejection-Incorrect	6	5	0.9801	0.9217
Significance-Correct	1	7	0.9856	0.9662
Significance-Incorrect	12	1	0.9765	0.8890
Ind-Variable-Mention	1	3	0.9928	0.9853
Dep-Variable-Mention	4	3	0.9874	0.9733
Conclusion-Correct	0	24	0.9567	0.8614
Conclusion-Incorrect	6	0	0.9892	0.9727

6 Conclusions and Future Work

We have described ConceptGrid, a tool that is intended to help non-programmers develop ITSs that perform natural language processing. It has been integrated into an ITS authoring tool called xPST. We tested it as a part of the xSTAT project and were able to approach the accuracy of human instructors in scoring student responses.

We would like to conduct an empirical evaluation study that helps demonstrate that the ConceptGrid tool, a part of xPST, is actually feasible for non-programmers to use on a variety of tasks, as we have done for xPST's core authoring tool [11]. The study will also help provide an insight into the time required by a tutor-author to develop templates for particular question types.

Currently, ConceptGrid does not support a dialogue between the student and tutor. It only evaluates student responses and gives just-in-time feedback. To support more extensive knowledge-construction dialogues, ConceptGrid responses would need to provide information required by the dialogue manager.

Our current approach is non-structural, i.e., it is focused on words and numerical analysis, rather than grammar and logic. The advantage with this approach is that it is simple for non-programmers to use, and is very effective in domains such as statistics where the student responses are expected to follow a general pattern. In addition, the ConceptGrid approach is domain-independent, one of its biggest advantages.

ConceptGrid could be extended to be structural as well, but that achievement might come at the cost of usability by non-programmers. To include structural matching, either the templates could nest by invoking other templates, or the atomic checktypes could include some checktypes that invoked structural matching. For nested concepts, we could define a concept and then use it within more complex concepts, in the following manner.

GreaterThan(X,Y) = X – "bigger" or "more"or "greater" – "than" – Y

WellFormedConclusion = GreaterThan("weight of the log", "weight of the twig")

This way, the framework can be extended to more powerful natural language processing using a similar approach to the processing that context-free grammars allow. Alternately, the set of ConceptGrid atomic checktypes could be extended to enable structurally-oriented checktypes that would match a nonterminal from a context-free grammar, such as an NP with "twig" as the head in a syntactically oriented grammar, or match the semantics of a section of the utterance.

Acknowledgments. This work is done with the support of the U.S. Air Force Office of Scientific Research.

References

1. Chi, M.T.H., de Leeuw, N., Chiu, M.H., LaVancher, C.: Eliciting Self-Explanations Improves Understanding. Cognitive Science 18, 439–477 (1994)
2. Chi, M.T.H., Bassok, M., Lewis, M.W., Reimann, P., Glaser, R.: Self-explanations: How Students Study and Use Examples in Learning to Solve Problems. Cognitive Science 13, 145–182 (1989)
3. Aleven, V., Koedinger, K., Cross, K.: Tutoring Answer Explanations Fosters Learning With Understanding. In: Proceedings of Artificial Intelligence in Education, AIED 1999, pp. 199–206 (1999)
4. Glass, M.: Processing Language Input in the CIRCSIM-Tutor Intelligent Tutoring System. In: Moore, J.D., et al. (eds.) Artificial Intelligence in Education, pp. 210–221 (2001)
5. Rus, V., Graesser, A.: Deeper Natural Language Processing for Evaluating Student Answers in Intelligent Tutoring Systems. American Association for Artificial Intelligence (2006)
6. Graesser, A., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N.: TRG.: Using Latent Semantic Analysis to Evaluate the Contributions of Students in AutoTutor. Interactive Learning Environemnts, 149–169 (2000)
7. Steinhart, D.: Summary Street: An Intelligent Tutoring System for Improving Student Writing Through the Use of Latent Semantic Analysis. Ph.D. dissertation. Dept. Psychology, University of Colorado, Boulder (2001)
8. Landauer, T.K., Foltz, P.W., Laham, D.: Introduction to Latent Semantic Analysis. Discourse Processes 25, 259–284 (1998)
9. 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)
10. Maass, J., Blessing, S.B.: xSTAT: An Intelligent Homework Helper for Students. Poster presented at the Georgia Undergraduate Research in Psychology Conference (2011)
11. Gilbert, S., Blessing, S.B., Kodavali, S.: The Extensible Problem-Specific Tutor (xPST): Evaluation of an API for Tutoring on Existing Interfaces. In: Dimitrova, V., et al. (eds.) Artificial Intelligence in Education, pp. 707–709. IOS Press, Brighton (2006)

Motivational Processes

Benedict du Boulay

Human Centred Technology Research Group, School of Informatics,
University of Sussex, Brighton, BN1 9QJ, UK
B.du-Boulay@sussex.ac.uk

Abstract. A motivationally intelligent tutor should determine the motivational state of the learner and also determine what caused that state. Only if the causation is taken into account can an efficient pedagogic strategy be selected to find an effective way to maintain or improve the learner's motivation. Thus we argue that motivation is more constructively thought of as a process involving causation rather than simply as a state. We describe methods by which this causality might be determined and suggest a range of pedagogic tactics that might be deployed as part of an overall pedagogic strategy.

Keywords: motivation, pedagogy, feelings, expectancies and values.

1 Introduction

Many scholars attest to the complex interplay between the cognitive and affective issues in learning [e.g. 1]. To the extent that tutors are concerned that learners stay engaged with the learning task, they are interested in how this interplay affects motivation: motivation is, after all, the impulse that drives the learner to exert effort in learning. The multiple bi-directional relations between the cognitive, the metacognitive, the affective, the meta-affective and motivation are complex [2]. Indeed analysis of the interplay of affect, self-assessment, competence, value judgements and effort show a wealth of interconnections which depend both on the individual history of the learner, but also on the current learning context [3]. In a similar vein, the expectancy-value theory of motivation maps links between the learner's cultural milieu, their expectations of success, the achievement choices that they make, their affective memories, their interpretations of their experience and many other factors [4]. While it may be helpful to identify the instantaneous motivational state of a student, the literature points to the idea that motivation is more productively viewed as a process that was operating long prior to the educational interaction, that continues to unfold during the learning, but that has causative antecedents in earlier learning experiences as well as consequences for future learning. There is a dynamic element to the way that the learner negotiates their perceptions of their learning experiences, their feelings and the impulse to exert or not to exert effort in learning. The process operates over different timescales. At the granularity of an individual lesson, or episode within a lesson, how the student reacts to success, failure, help or hint (say) will be driven by the kind of parameters identified above. Over longer timescales, the relationships between these driving

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 55–62, 2011.

parameters can themselves change, e.g. as the learner develops their capability as a self-regulated learner [5].

In order to help manage this complexity, we choose to label instantaneous motivational states in terms of their main characteristic affective component. Thus we can say, as a kind of shorthand, that a student is in the motivational state of being bored (say), if the main affective dimension of that state is boredom. Of course, how an individual student reacts to their motivational state of boredom depends on other factors. For example, some students may be spurred into finding adaptive ways (such as setting themselves more challenging work) or mal-adaptive ways (such as gaming the system) to reduce their boredom. Others may simply acquiesce to the boredom and disengage altogether. That different learners become bored for different reasons and then go on to deal with that boredom in different ways supports the idea that motivation should be regarded as a process that unfolds in an individual way.

Much effort has been devoted to developing methods to determine the instantaneous motivational and affective state of the student: from self-report, from facial expression, from posture, from skin conductance, from pressure on the mouse, from language, from behaviour and from other clues [see e.g. 6, 7-13]. In terms of developing a pedagogy to make use of this information, various positive and negative affective states (and cycles of states) have been identified as important in education including anxiety, boredom, confusion, delight, disappointment, enjoyment, flow and frustration [see e.g. 14, 15-18]. Whilst positive states are generally desirable, learning will often involve some negative episodes, especially of confusion or frustration when hard problems are encountered, or of anxiety when anticipating difficult issues ahead.

There is limited scope for making the best choice of pedagogic response based on an analysis of the current state of the learner only. So this paper explores ways that this analysis can be augmented in order to assist the tutor. The rest of the paper is divided into three sections. The first develops the notion of motivational processes through the idea of trajectories of states. The subsequent section looks at methods by which the tutor might gather information about the cause(s) of a particular motivational state. The following section then outlines various pedagogic tactics for dealing with the cause(s) of (negative) motivational states.

2 States, Trajectories and Motivational Processes

The underlying model of most tutoring systems is based on the idea that the student passes through a sequence of motivational states which have both a cognitive and an affective dimension. To an extent the trajectory of these states is determined by the content and the conduct of the academic work that is being undertaken. However the history of that person as a learner and influences outside the lesson can have a large effect, such as a row with friend before the lesson or a sequence of prior awkward interactions with that teacher. A second important determinant of motivation and thus of the trajectory of states is the effect of the many parameters identified earlier, such as self-assessments and value judgements. For example, a context-specific distinction is often drawn between mastery and performance orientated learners and the difference these orientations have for the learner's expectation of, and particularly interpretation of, error and setbacks [19]. So it is not just a matter of what the learner is trying to achieve, it is also a matter of the way that they see themselves as learners and the degree that that perception influences how much effort they are willing to put into the business of learning. Other traits and personality variables [see e.g. 20] also affect the unfolding trajectory of states.

Given the above it is perhaps best to think of the learner's motivation as a complex process that interacts with events during learning in ways that are sometimes quite hard to determine, even for observant human teachers. The tutoring system must thus act as a diagnostic tool in part attempting to determine the current state of the student, but also attempting to unpick the causality that might have led to the current state or the current behaviour [see e.g. 21]. To make matters harder, some learners are adept at masking their affective states, particularly when it comes to maintaining "face" in front of their peers. It is also the case that even human teachers find it difficult to decide on the affective state of their students, partly because of masking, and partly because academic affective reactions can be quite nuanced [22].

3 Gathering Data

We suggest that there are several ways to try to get a better sense of what drives a particular student. First, this would involve extending the scope of the logging of interactions, e.g. via learning diaries, as described by Zimmerman [5]. Such logs contain data about affect (whether gathered through self-report or by less intrusive methods) and these would be integrated with performance data. It is helpful to have a record that extends backwards over several sessions so that the tutor has the chance to detect repeated patterns of cognitive and affective interaction. The tutor would then also be able to refer back to both positive and negative episodes, their precursors, and their consequences as part of its tutorial strategy [23] in a manner not unlike ELM-ART working at the cognitive level [24]. Some steps towards this extension of the logging have been undertaken [25]. Van Zijl adapted a version of the EER Tutor [9] to employ diagrammatic self-report to record the learner's affective valence (i.e. whether they felt positive, negative or neutral). The system was augmented with motivational rules that used this data along with performance data to refer the undergraduate student to past successes.

A second way of understanding better what drives the learner is to engage with the learner *about* their experience of learning. While a full natural language dialogue about the learning domain is hard enough [see e.g. 26], interacting about the learner's experience of grappling with that domain is likely to be harder, though a menu-based interaction can be helpful [27], especially when tackled by pairs of students. Both at the outset of a lesson and again at the end, the pair can be asked questions about their expectations and values and how they anticipate their experience (or how it worked out in fact). Each learner could be responsible for making the entry on behalf of their peer. This might reduce gaming and lead to a discussion about how to interpret the menu options and whether the choice of answer was correct, see [28]. Even if the tutor ignored this input, there should be metacognitive and meta-affective benefits for the two learners in thinking about and articulating their expected and actual cognitive and affective reactions to the learning. This greater insight into their own motivational processes should help them then deal with any motivational inadequacies of the tutoring system itself [29].

4 Pedagogic Tactics

Useful detailed empirical work on identifying pedagogical tactics has been undertaken by observing skilled teachers, e.g. [30], or more specifically their responses to particular states, such as use of an "off-topic" comment in response to a student who is happy or confused [31]. However we argue that it is not enough simply to identify the current affective state of the student (e.g. frustrated) in order to determine an appropriate course of pedagogic action [32]. Pintrich [33] suggests that the motivational literature has explored two broad areas in addition to Feelings which drive motivational processes. These are associated with Expectancies and Values. Even taking a narrow view of these two areas leads to contrasting remedial tactics.

4.1 Expectancies

Various negative motivational states such as confusion, anxiety, frustration and boredom can be traced to negative expectations of either the experience of undertaking the learning task or its outcomes. So a student might be frustrated (say) because the work is too easy and their anticipation is that the remainder of the lesson is likely to lack challenge and interest. A sensible pedagogic response in this case might be to follow Keller's [34] advice and stimulate the learner's curiosity to increase their degree of engagement. A student might also be frustrated because the work appears too hard and they have little expectation of understanding it. Here the strategy might be to suggest easier work, if it is believed that the learner's sense of their own capability for the task in hand, their self-efficacy [35], is well-founded; or possibly to show by reference to their previous achievements that success is in fact likely, if their sense of their capability is too pessimistic. A learner may also be anxious that he or she will not be able to tackle a problem successfully, or that the work is too easy, or that there may be some public loss of "face". In dealing with this kind of issue, it is again helpful to try to determine whether the learner's expectations are accurate. This will need evidence from prior learning episodes to establish whether the learner has a tendency towards realism, optimism or pessimism (as one way to divide such judgements) in these matters [36]. Where there is a realistic fear of failure or other negative experience then steps can be taken to make the work easier, to scaffold it more densely or otherwise to reduce the chances of failure. Where the learner is pessimistic, the tutor can use the evidence already accumulated to reacquaint the learner with similar previous episodes that demonstrate past success. This could be augmented with changes to the task or to its scaffolding just as for realistic students. Where the learner is generally optimistic but is nevertheless in a negative motivational state, then more complex action may be needed involving exploration of exactly what the negative expectations are and why they have emerged in order to try to deal with them.

4.2 Values

Continuing with the issue of frustration, in the case of Values, a student might be frustrated because he or she has no interest in the lesson (irrespective of whether it is easy or difficult) and would rather be doing something else. Thus within the Values sphere, some negative motivational states can be traced to a mismatch between the values of the learner and the values associated with the learning task. Values include both the learner's goals as well as the value judgements he or she applies to different kinds of learning experience. In general terms there are three different ways to realign the values mismatch. The first is to ensure that the mismatch is not simply down to the learner's misunderstanding of what the values associated with the learning task actually are. If the learner has an accurate but only partial understanding of the nature of the task, it may be that this understanding can be augmented to align with his or values. For example, supposing someone finds themselves taking a mandatory statistics class for which they have little appetite, it may be possible to show how successful completion of the class will assist them in some area that they do value, but had not realised would be helped by a deeper understanding of statistics. The second way is to try change the learner's values themselves, and the third is to change the nature of the learning so that it aligns better to the learner.

It may be that the system is unable to determine the cause(s) of a particular negative motivational state. If that is the case there seem to be several possibilities. First the tutoring system could turn the issue over to the learner, and ask the learner to choose between the different remedial tactics available (as outlined above). Second it could rank the tactics in order of prior success for that learner, and if that data is not available, then in order of prior success for that class of learner. It could then try the most highly ranked tactic whilst monitoring the learner's reaction and move to the next most highly ranked tactic if things seem to be getting worse rather than better.

4.3 Meta Level Tutoring

In all cases the expected and actual trajectories of motivational state can be captured by the tutoring system for two purposes. First is their utility for potential use later when a similar situation occurs. The second is to open up the possibility of the tutoring system engaging in a meta-affective (discussion of the feelings experienced during learning) and meta-motivational tutoring (discussion of factors that impede or facilitate a learner becoming self-regulated [5]). By taking the learner back through an interaction and getting them to focus on the cognitive, affective and motivational trajectory he or she has traversed, there should be scope for developing the learner's insight into his or her strengths, weaknesses and strategies *as a learner* (i.e. developing the long-term motivational process mentioned earlier).

5 Conclusions

We have argued that a motivationally intelligent tutoring system should take account not just of the instantaneous motivational state of the learner, but also of the causative motivational processes that led to that state. Simply ascertaining that a student is bored or frustrated (say) is not enough on its own to determine what best to do next. We have suggested ways in which learner logs could be used to counteract tendencies towards inaccurate self-assessment and to develop the learner's meta-affective and meta-motivational insight. We have outlined some pedagogic tactics, dividing them into those operating the area of Expectancies and those in the area of Values.

Acknowledgments. I thank Martin van Zijl, Katerina Avramides, Alison Hull and the anonymous referees for their helpful suggestions on earlier drafts of this paper.

References

1. Forgas, J.P.: Affect and Cognition. Perspectives on Psychological Science 3, 94–101 (2008)
2. du Boulay, B., Avramides, K., Luckin, R., Martinez-Miron, E., Rebolledo-Mendez, G., Carr, A.: Towards Systems That Care: A Conceptual Framework based on Motivation, Metacognition and Affect. International Journal of Artificial Intelligence and Education 20(3), 197–229 (2010)
3. Boekaerts, M.: Understanding Students' Affective Processes in the Classroom. In: Schutz, P.A., Pekrun, R. (eds.) Emotion in Education, pp. 37–56. Acadmic Press, Burlington (2007)
4. Wigfield, A., Eccles, J.S.: Expectancy–Value Theory of Achievement Motivation. Contemporary Educational Psychology 25, 68–81 (2000)
5. Zimmerman, B.J.: Investigating Self-Regulation and Motivation: Historical Background, Methodological Developments, and Future Prospects. American Educational Research Journal 45, 166–183 (2008)
6. Arroyo, I., Cooper, D.G., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion Sensors Go to School. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Grasser, A. (eds.) Artificial Intelligence in Education. Building Learning Systems that Care: from Knowledge Representation to Affective Modelling, Vol. Frontiers in AI and Applications 200, pp. 17–24. IOS Press, Amsterdam (2009)
7. D'Mello, S., Graesser, A., Picard, R.W.: Toward an affect-sensitive AutoTutor. IEEE Intelligent Systems 22, 53–61 (2007)
8. Zeman, J., Klimes-Dougan, B., Cassano, M., Adrian, M.: Measurement Issues in Emotion Research With Children and Adolescents. Clinical Psychology: Science and Practice 14, 377–401 (2007)
9. Zakharov, K., Mitrovic, A., Johnston, L.: Towards Emotionally-Intelligent Pedagogical Agents. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. L. (eds.) ITS 2008. LNCS, vol. 5091, pp. 19–28. Springer, Heidelberg (2008)
10. Kleinsmith, A., De Silva, P.R., Bianchi-Berthouze, N.: Recognizing Emotion from Postures: Cross-Cultural Differences in User Modeling. In: Ardissono, L., Brna, P., Mitrovic, A. (eds.) UM 2005. LNAI, vol. 3538, pp. 50–59. Springer, Heidelberg (2005)
11. Conati, C., Chabbal, R., Maclaren, H.: A Study on Using Biometric Sensors for Monitoring User Emotions in Educational Games. In: Proceedings of the Workshop Assessing and Adapting to User Attitude and Affects: Why, When and How? 9th International Conference on User Modeling, UM 2003 (2003)
12. D'Mello, S.K., Craig, S.D., Witherspoon, A., McDaniel, B., Graesser, A.: Automatic detection of learner's affect from conversational cues. User Modeling and User-Adapted Interaction 18, 45–80 (2008)
13. Baker, R., Walonoski, J., Heffernan, N., Roll, I., Corbett, A., Koedinger, K.: Why Students Engage in "Gaming the System" Behaviours in Interactive Learning Environments. Journal of Interactive Learning Research 19, 185–224 (2008)
14. Baker, R.S.J.d., Rodrigo, M. M.T., Xolocotzin, U.E.: The Dynamics of Affective Transitions in Simulation Problem-Solving Environments. In: Paiva, A., Prada, R., Picard, R.W. (eds.) ACII 2007. LNCS, vol. 4738, pp. 666–677. Springer, Heidelberg (2007)

15. Muldner, K., Burleson, W., VanLehn, K.: "Yes!": Using Tutor and Sensor Data to Predict Moments of Delight during Instructional Activities. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 159–170. Springer, Heidelberg (2010)
16. Pekrun, R., Goetz, T., Titz, W., Perry, R.P.: Academic Emotions in Students' Self-Regulated Learning and Achievement: A Program of Qualitative and Quantitative Research. Educational Psychologist 37, 91–105 (2002)
17. Larsen, J.T., McGraw, A.P., Mellers, B.A., Cacioppo, J.T.: The Agony of Victory and Thrill of Defeat Mixed Emotional Reactions to Disappointing Wins and Relieving Losses. Psychological Science 15, 325–330 (2004)
18. Graesser, A., Chipman, P., King, B., McDaniel, B., D'Mello, S.: Emotions and Learning with AutoTutor. In: Luckin, R., Koedinger, K.R., Greer, J. (eds.) Proceeding of the 2007 Conference on Artificial Intelligence in Education: Building Technology Rich Learning Contexts that Work, vol. Frontiers in AI and Applications 158, pp. 569–571. IOS Press, Amsterdam (2007)
19. Dweck, C.S., Chiu, C.-y., Hong, Y.-y.: Implicit Theories and Their Role in Judgments and Reactions: A Word From Two Perspectives. Pscychological Inquiry 6(4), 267–285 (1995)
20. Conati, C., Zhou, X.: Modeling Students' Emotions from Cognitive Appraisal in Educational Games. In: Cerri, S.A., Guy, G., Paraguacu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 944–954. Springer, Heidelberg (2002)
21. Baker, R.S.J.d., D'Mello, S.K., Rodrigo, M.M.T., Graesser, A.C.: Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive–affective states during interactions with three different computer-based learning environments. International Journal of Human-Computer Studies 68, 223–241 (2010)
22. Balaam, M., Luckin, R., Good, J.: Supporting affective communication in the classroom with the Subtle Stone. International Journal of Learning Technology 4, 188–215 (2009)
23. Hull, A., du Boulay, B.: Scaffolding Motivation and Metacognition in Learning Programming. In: Dimitrova, V., Mizoguchi, R., du Boulay, B., Grasser, A. (eds.) Artificial Intelligence in Education. Building Learning Systems that Care: from Knowledge Representation to Affective Modelling, vol. Frontiers in AI and Applications 200, pp. 755–756. IOS Press, Amsterdam (2009)
24. Weber, G., Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-based Instruction. International Journal of Artificial Intelligence in Education 12, 351–384 (2001)
25. van Zijl, M.: Towards a Motivationally Intelligent Pedagogical Agent. Department of Computer Science and Software Engineering. University of Canterbury, Christchurch (2010)
26. Graesser, A.C., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: an intelligent tutoring system with mixed-initiative dialogue. IEEE Transactions on Education 48, 612–618 (2005)
27. del Soldato, T., du Boulay, B.: Implementation of Motivational Tactics in Tutoring Systems. International Journal of Artificial Intelligence in Education 6, 337–378 (1995)
28. Puntambekar, S., du Boulay, B.: Design of MIST – A System to Help Students Develop Metacognition. In: Murphy, P. (ed.) Learners, Learning & Assessment, pp. 245–257. Paul Chapman Publishing, London (1999)
29. Avramides, K., du Boulay, B.: Motivational Diagnosis in ITSs: Collaborative, Reflective Self-Report. In: Dimitrova, V., Nizoguchi, R., du Boulay, B., Graesser, A. (eds.) Artificial Intelligence in Education. Building Learning Systems that Care: From Knowledge Representation to Affective Modelling, vol. Frontiers in AI and Applications 200, pp. 587–589. IOS Press, Amsterdam (2009)
30. Lepper, M.R., Woolverton, M., Mumme, D.L., Gurtner, J.: Motivational techniques of expert human tutors: Lessons for the design of computer-based tutors. In: Lajoie, S., Derry, S. (eds.) Computers as Cognitive Tools, pp. 75–105. Lawrence Erlbaum Associates, Hillsdale (1993)

31. 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.L. (eds.) ITS 2008. LNCS, vol. 5091, pp. 50–59. Springer, Heidelberg (2008)
32. du Boulay, B.: Towards a Motivationally-Intelligent Pedagogy: How should an intelligent tutor respond to the unmotivated or the demotivated? In: Calvo, R.A., D'Mello, S.K. (eds.) Affective Prospecting, Explorations in the Learning Sciences. Instructional Systems and Performance Technologies, vol. 3. Springer, New York (2011)
33. Pintrich, P.: Motivation and Classroom Learning. Handbook of Psychology: Educational Psychology 7, 103–122 (2003)
34. Keller, J.M.: Motivational design of instruction. In: Reigluth, C.M. (ed.) Instructional design theories and models: An overview of their current status, pp. 386–434. Lawrence Erlbaum, Hillsdale (1983)
35. Bandura, A.: Self-efficacy: The exercise of control. Freeman, New York (1997)
36. Gama, C.: Metacognition in Interactive Learning Environments: The Reflection Assistant Model. In: Lester, J.C., Vicari, R.M., Paraguacu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 668–677. Springer, Heidelberg (2004)

Using Tutors to Improve Educational Games

Matthew W. Easterday, Vincent Aleven, Richard Scheines, and Sharon M. Carver

Human-Computer Interaction Institute, Carnegie Mellon University

Abstract. Educational games and tutors provide conflicting approaches to the assistance dilemma, yet there is little work that directly compares them. This study tested the effects of game-based and tutor-based assistance on learning and interest. The laboratory experiment randomly assigned 105 university students to two versions of the educational game Policy World designed to teach the skills of policy argument. The game version provided minimal feedback and imposed penalties during training while the tutor version provided additional step-level, knowledge-based feedback and required immediate error correction. The study measured students' success during training, their interest in the game, and posttest performance. Tutor students were better able to analyze policy problems and reported higher level of competence which in turn affected interest. This suggests that we can improve the efficacy and interest in educational games by applying tutor-based approaches to assistance.

Keywords: intelligent tutoring systems, educational games, policy argument, debate.

Educational games promise to make learning fun. Games typically provide less explicit assistance and harsher penalties than intelligent tutors, and perhaps as a result, more interesting choices. Do the mechanics that make games fun also promote learning? Or is lowered assistance the price we pay for increasing interest?

A review of educational game research shows a lack of empirical evaluation, especially the controlled comparisons between games and other approaches that would allow us to answer this question [1]. The explosion of AIED/ITS work on games in the last several years has produced scores of papers but has not radically altered the situation [2-4]. The great majority of work includes either no empirical evaluation or no control, and many of the remaining controlled experimental studies compare features that are important but not intrinsic to games or tutors [5-7].

To determine whether games offer a superior approach, we need to test whether their essential features, like the possibility of losing, the hidden or uncertain state created by opponents or random events, and the lack of external rewards, interfere with learning. We also need to test commonly used features like fantasy contexts [8]. Likewise we need to know whether the essential features of tutors such as how they provide step-level assistance interfere with interest. Of the few recent experimental studies in AIED/ITS, some have shown games to be inferior to or no better than didactic, non-intelligent instruction [9-10]. Another found no difference in learning, but a benefit in engagement for the game [11].

Tutors are inherently defined by how they provide assistance [12]. Thus, the greatest potential conflict between games and tutors is their different approaches to

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 63–71, 2011.

assistance. AIED/ITS research must determine whether games offer a superior solution to the assistance dilemma [13], or whether we simply transplant traditional tutor-based assistance into games without harming interest.

The purpose of this study was to compare the effects of *assistance* (using either a tutor or game-based approach) on learning and interest using an educational game called Policy World that teaches the skills of policy argument [14-15]. In the game-based version, the student received only a baseline level of assistance typically used in games including: situational feedback such as the game characters' dialogue, minimal-error flagging via the scoreboard, and penalties for making errors, such as restarting a level. In the tutor-based version, the student received additional knowledge-based feedback on every step and was required to immediately correct errors. In other words, the tutor always gave hints while the game let students die (fail and restart). Learning variables included students' learning of the search, comprehension, evaluation, diagram construction, synthesis, and decision skills taught by Policy World. Interest was measured by the Intrinsic Motivation Inventory [16].

These variables allow us to pose several competing hypotheses:

1. **Game hypothesis:** game-based assistance will increase learning and interest.
2. **Tutor hypothesis:** tutor-based assistance will increase learning and interest.
3. **Assistance tradeoff:** game-based assistance will increase interest, while tutor-based assistance will increase learning.

Intuitively, we might expect tutors to be more effective at increasing learning, because the principles upon which they are based have been derived from decades of empirical work [17] and because of the empirically demonstrated benefits of immediate, knowledge-based feedback with immediate error-correction [18]. On the other hand, situational feedback and delayed intelligent novice-feedback, similar to that offered by games, can be just as effective or even more effective at promoting learning as immediate, knowledge-based feedback [19-20], although their effects on interest are unclear. Intuitively, the game might be more fun because it gives the player more autonomy and the satisfaction of winning. On the other hand, excessive floundering is not fun, and the additional assistance offered by the tutor might be welcomed by a struggling student. These competing intuitions and tradeoffs form the core of the assistance dilemma especially as applied to educational games [13, 21].

Method

Population and Setting

105 university students were recruited through an on-line participant database and campus flyers. Students were compensated $20 for completing the on-line study, an additional $5 for passing posttest 1, and an additional $5 for passing posttest 2.

Intervention

Policy World. Policy World is an educational game designed for teaching the skills of policy argument. Students play the role of an analyst who must provide policy recommendations on topics like the drinking age, video game violence, carbon emissions, and national health care. The current version has 6 levels: a pretest, 3 training levels, and two posttests. Most levels include three broad activities: searching for policy information, analyzing that information, and debating policy recommendations with a computer opponent. During search, students use a fake Google interface to find newspaper reports containing causal claims. During analysis, students use causal diagramming tools to analyze causal claims. During debate, students make a policy recommendation, explain how the policy will affect a desired outcome, and provide evidence for their position by citing reports. The analysis tools were disabled on the pretest and posttest 2.

Baseline game assistance. The baseline assistance in Policy World consisted of minimal feedback and penalties. During analysis, red/gold scoreboard stars indicated whether the student passed an analysis stage. During debate, the judge character would comment on critical mistakes and give the student a *strike* (as in baseball). The dialogue provided a form of situational feedback while the stars and strike provided a form of minimal feedback. As in most games, this assistance provided neither explicit teaching feedback nor was it at the step level. The baseline penalty was lost progress. When the student made an error on a stage of analysis of a particular causal claim, they were sent back to the first analysis stage. When the student received too many debate strikes, they had to replay the whole level. Note that in this study, students were automatically promoted after testing levels and were given the option to be promoted past a training level after playing it twice.

Tutoring assistance. The tutor version of Policy World provided supplemental assistance *only* during training. This additional assistance included explicit step-level feedback and immediate error correction. During training of analysis and debate the tutor provided explicit error-specific and teaching feedback on each step. The tutor also required immediate error correction, thus overriding Policy World's penalties.

Design

The study used a two-group, between-subjects, randomized, controlled, experimental design that compared a *game* to a *tutor* version of Policy World.

Task, Training Feedback, and Measures

Each Policy World level consisted of two phases: *search and analysis* and *debate*. In the search and analysis phase, the student searched for evidence using a fake Google interface to find 3-7 newspaper-like reports, 3-5 paragraphs in length, based on articles from sources like the *New York Times* and *Frontline.* At any time during this phase, the student could select a report to analyze which required him to comprehend, evaluate, diagram, and synthesize the evidence about the causal claims in the report.

Comprehend. After selecting a report to analyze, the student attempted to highlight a causal claim in the text such as: *the Monitoring the Future survey shows that 21 minimum drinking age laws decrease underage consumption of alcohol.*

Evaluate. The student then used combo boxes to identify the evidence type (*experiment, observational study*, *case*, or *claim)* and strength of the causal claim. Strength was rated on a 10-point scale with the labels: *none*, *weakest*, *weak*, *decent*, *strong*, and *strongest.* The evaluation was considered correct if: (a) the evidence type was correctly specified, and (b) the strength rating roughly observed the following order taught during training: experiments > observational studies > cases > claims.

Diagram. The student next constructed a diagrammatic representation of the causal claim using boxes to represent variables and arrows to represent an *increasing, decreasing,* or *negligible* causal relationship between the two variables. The student also "linked" the causal claim in the report to the new diagram arrow which allowed him to reference that report during the debate by clicking on that arrow.

Synthesize. The student then *synthesized* his overall belief about the causal relationship between the two variables based on all the evidence linked to the arrows between those variables up to that point. The synthesis step required the student to specify which causal relationship between the two variables was best supported by the evidence, and his confidence in that relationship on a 100 point slider from *uncertain* to *certain.* During training, a synthesis attempt was considered valid if: (a) the student moved his belief in the direction of the evidence, assuming the student's description of the evidence was correct, and (b) the student's belief mirrored the overall evidence, assuming the student's description of the evidence was correct.

Analysis feedback. During training, analysis errors resulted in animated red stars. Game students received no explanation for the error and were forced to restart the analysis of the claim. Tutor students received explanations and got to try again.

After ending the analysis phase, students moved to debate phase.

Recommendation. In the first step of the debate, the judge asked the student to choose a policy recommendation from a list of policy options which included increasing or decreasing any of the possible variables or *doing nothing*. For example, the student might recommend that: *we should repeal the 21 age drinking limit.* If the student proposed a recommendation that defied common sense or any directives given at the start of the problem, for example: *decreasing people's genetic propensity to drink,* the judge overruled the recommendation and gave the student a strike.

Mechanism. If the student proposed any recommendation besides *doing nothing*, the judge then asked the student to provide a mechanism that explained how the recommendation affected the desired policy outcome. The student used a set of combo boxes representing variables and causal relations to construct a mechanism such as: *repealing the drinking limit will decrease binge drinking which will decrease drunk driving*. If the student constructed an incoherent mechanism, for example that did not include the policy outcome, the judge gave the student a strike.

Mechanism Attack. If the student recommended *doing nothing*, the opponent proposed an alternate recommendation and mechanism, such as: *repealing the drinking limit will decrease binge drinking which will decrease drunk driving.* The student then had to attack a causal relation in the opponent's mechanism with an alternate relation, like: *repealing the drinking limit will not decrease binge drinking.* If the student made an incoherent attack by agreeing with the opponent or attacking a claim not in the opponent's mechanism, the judge gave the student a strike.

Evidence. After explaining or attacking a mechanism, the judge asked the student to cite reports with causal claims supporting the student's causal claim. Ideally, the student consulted his diagram by checking and clicking the relevant arrow on the diagram and checking the reports linked to that arrow during analysis. If the student provided a mechanism, the opponent would attack up to three causal claims in that mechanism before the student won the debate. If the student attacked an opponent's mechanism, he only had to provide evidence for one attack, which would invalidate the opponent's entire causal chain. If the student provided irrelevant evidence, or weaker evidence than the opponent, the student received a strike.

Debate feedback. During training, all students received strikes for the gross errors described earlier. After 5 strikes, game students had to restart the level. Tutor students were given tutoring both for gross errors and any debate move inconsistent with their analysis. For example, a plausible recommendation inconsistent with the student's diagram would not receive a strike, but would receive tutoring. Citing sufficient, but not *all,* relevant evidence also initiated tutoring. Tutor students were then given Socratic tutoring on how to use the diagram and asked to try again.

Table 1. Posttest Measures

Measure	Description
Comprehend	# of claims correctly identified
Evaluate	# of correct evaluations of type and strength
Diagram	# of diagram elements linked to valid claims
Synthesize	# of times the synthesized relation and confidence shifted toward new perceived evidence and consistent the perceived evidence
Recommend	# of unique winning recs / (# unique winning recs + # losing recs)
Mechanism	# of unique winning mechanisms / (# unique winning mechanisms + # losing mechanisms)
Attack	# of unique winning attacks / (# unique winning attacks + # losing attacks)
Evidence	# unique winning ev. attempts / (# unique wining ev. attempts + # losing ev. attempts)
Training success	Average (# correct / (# attempts of each step on all training problems))
IMI	Intrinsic Motivation Inventory with sub-scales measuring: competence, effort, pressure, choice, value and interest [16]

During testing, students were allowed to construct arbitrarily incorrect diagrams, so we used the proxy measures of diagram and synthesis correctness.

Procedure

Students first took a *pretest* on either: junk food advertising and childhood obesity (13-15 causal statements), health care (8-9 causal statements), and cap and trade (9-10 causal statements). During the pretest, the analysis tools for comprehension, evaluation, construction, and synthesis were not available. All students were allowed to search for as many or as few reports as they liked before continuing to the debate.

Students were then randomly assigned to the *game* or *tutor* condition. Each group completed 3 *training* problems on video game violence (3 causal statements), the drinking age (12 statements), and the meth epidemic, (8 statements). During training, game students received only baseline feedback and penalties. Game students who failed a training level debate had to replay it once before being promoted to the next level. Tutor students received additional step-level, knowledge-based feedback and immediately corrected errors so they always successfully completed these levels.

Finally students played *posttest 1* (with analysis tools) and *posttest 2* (without analysis tools) levels counterbalanced with the pretest. The posttests provided neither the tutor nor the baseline analysis assistance. Students did not replay the posttests.

Results

Analysis 1: Who Learns More? We first examined success on the posttest 1 analysis steps. Table 2 shows that tutor students surpassed game students on every pre-debate analysis step. This suggests that adding step-level, knowledge-based feedback and immediate error correction increases learning in a game environment.

There was no significant difference between the two groups on any of the debate tasks on either posttest. By analogy to Algebra: tutor students did better constructing the equation but were still far from solving it.

Table 2. Comparison of Game and Tutor Groups on Posttest 1 Analysis

	Game		Tutor					
Measure	Mean	SD	Mean	SD	t	p	ll	ul
Comprehended	0.870	1.738	3.784	3.25	-9.816	6.18E-21 ***	-4.68	-3.1
Evaluated	0.704	1.449	2.549	2.48	-11.72	8.04E-28 ***	-5.03	-3.6
Diagramed	0.833	1.678	3.353	3.09	-5.148	2.00E-06 ***	-3.49	-1.5
Synthesized	1.056	1.937	5.078	4.42	-5.978	9.43E-08 ***	-5.37	-2.7

Analysis 2: Path model. We used path analysis to examine the causal relationships between assistance, training, interest, analysis, and debate. To search over all path models consistent with our background theories and that fit the data, we used the GES algorithm [22] implemented in Tetrad 4 to search for equivalence classes of un-confounded causal models consistent with the correlations in Table 3 and prior knowledge about the relationships between variables. This included the knowledge that: assistance was determined before any other factor, training was completed next, intrinsic motivation was measured before posttest 1, the student created a posttest 1 diagram before debating, and recommendations were provided before evidence.

Table 3. Correlations for Assistance, Diagramming, Debate, and Motivation

			Intrinsic Motivation Inventory										
	Assist	Train	Interest	Comp	Effort	Press	Choice	Value	Diag	Rec	Ev	M	SD
Asst	1											0.49	0.50
Train	.69***	1										0.58	0.19
Int	.05	.21*	1									3.82	1.34
Com	.44***	.50***	.57***	1								3.22	1.34
Eff	.04	.07	.13	-.04	1							5.16	1.06
Pres	-.14	-.22*	-.25***	-.37***	.34***	1						4.26	1.28
Cho	-.12	-.07	.44***	.36***	-.08	-.19*	1					3.58	1.08
Val	.12	.18.	.81***	.57***	.16	-.19*	.34***	1				4.37	1.35
Diag	.46***	.50***	.18	.42***	-.01	-.16.	-.02	.25***	1			2.07	2.77
Rec	-.01	-.01	.07	.18.	-.12	-.07	.06	.10	.26**	1		0.24	0.35
Ev	-.02	.07	.20***	.25***	-.02	-.13	.10	.19	.37***	.56***	1	0.22	0.37

*p<.05 **p<.01 ***p<.001

Figure 1 shows the model discovered by Tetrad which we consider highly plausible and shows an excellent fit to the data. A chi-squared test of the deviance of the path model from the observed values showed that we cannot reject this model at a significance level of .05, $\chi2$ (40, n = 105) = 40.31, p > .46. Larger p-values indicate better fit and values above .05 indicate that we cannot reject the model at a significance level of .05.

According to the path model, tutor students had a greater success rate during training (as in Analysis 1). Students with greater success during training were more likely to diagram on posttest 1. Students who diagrammed more were more likely to make winning recommendations and to provide winning evidence. Students who had more success in providing recommendations were more likely to succeed in providing winning evidence. Those who received more assistance and those who had greater training success were more likely to report feeling competent. Those reporting higher competence valued the activity more for learning about policy, which increased interest. Those who perceived more choice while playing the game felt more competent and were more interested in the game, however assistance did not affect choice. Interest was correlated with, but did not cause competence and task success.

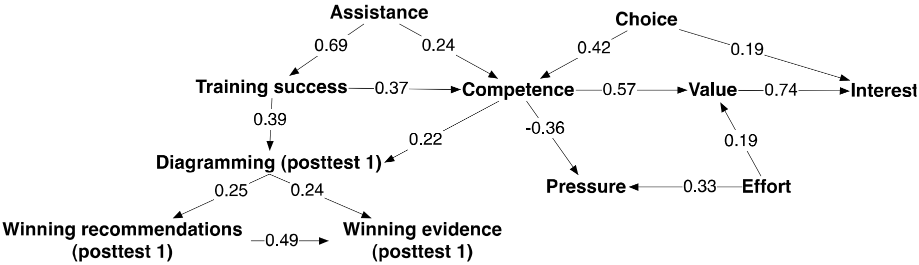


Fig. 1. A path model of the relations between the assistance, success on training, the amount of diagramming on posttest 1, posttest 1 debate performance, and intrinsic motivation

Discussion

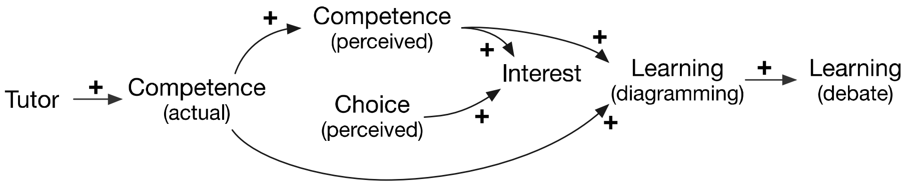


Fig. 2. Summary of results indicating support for mechanisms of the tutoring hypothesis

The results support the *tutoring hypothesis* that adding tutoring-based assistance to game environments increases both learning and interest (specifically competence). Figure 2 summarizes our view of the mechanisms that explain the patterns in the data we collected in this study. Adding tutoring to the game-like inquiry environment helped students succeed in training, which increased their ability to create diagrams on the posttest, which increased their ability to cite winning evidence during the policy debate. Adding tutoring also increased students' self-reported competence, which increased their interest in the game which did not affect learning. Choice *did* increase interest in the activity, however choice was not affected by the tutor. The results can be described intuitively: assistance increased competence, which is good for learning and interest. The mechanisms between assistance, learning, and interest described by these results provide consistent support for the use of tutors in games.

Acknowledgements. This work was supported in part by a graduate training grant awarded to Carnegie Mellon University by the U.S. Department of Education (# R305B040063), the Siebel Scholars Foundation, and the Pittsburgh Science of Learning Center, funded by the National Science Foundation (# SBE-0836012). The opinions expressed are the authors' and do not represent the views of the U.S. Department of Education, the Siebel Scholars Foundation, or the National Science Foundation.

References

[1] Hays, R.T.: The Effectiveness of Instructional Games: A Literature Review and Discussion (Tech Rep. 2005-004). Storming Media (2005)

[2] Aleven, V., Kay, J., Mostow, J. (eds.): ITS 2010. LNCS, vol. 6094 & 6095. Springer, Heidelberg (2010)

[3] Dimitrova, V., Mizoguchi, R., du Boulay, B., Graesser, A. (eds.): Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling. IOS Press, Amsterdam (2009)

[4] Lane, H.C., Ogan, A., Shute, V. (eds.): Proceedings of the Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, Brighton, UK (2009)

[5] Ogan, A., Aleven, V., Kim, J., Jones, C.: Intercultural Negotiation with Virtual Humans: The Effect of Social Goals on Gameplay and Learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 174–183. Springer, Heidelberg (2010)

[6] Rowe, J.P., Mott, B.W., McQuiggan, S.W., Robison, J.L., Lee, S., Lester, J.C.: Crystal island: A narrative-centered learning environment for eigth grade microbiology. In: Lane, H.C., Ogan, A., Shute, V. (eds.) Proceedings of the Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, Brighton, UK, pp. 11–20 (2009)

[7] Lane, H.C., Hays, M.J., Auerbach, D., Core, M.G.: Investigating the Relationship between Presence and Learning in a Serious Game. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 274–284. Springer, Heidelberg (2010)

[8] Cordova, D.I., Lepper, M.R.: Intrinsic motivation and the process of learning: Beneficial effects of contextualization, personalization, and choice. Journal of Educational Psychology 88(4), 715–730 (1996)

[9] McQuiggan, S.W., Rowe, J.P., Lee, S., Lester, J.C.: Story-Based Learning: The Impact of Narrative on Learning Experiences and Outcomes. In: Woolf, B., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 530–539. Springer, Heidelberg (2008)

[10] Lane, H.C., Schneider, M., Albrechtsen, J.S., Meissner, C.A.: Virtual Humans with Secrets: Learning to Detect Verbal Cues to Deception. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 144–154. Springer, Heidelberg (2010)

[11] Hallinen, N., Walker, E., Wylie, R., Ogan, A., Jones, C.: I was playing when I learned: A narrative game for French aspectual distinctions. In: Lane, H.C., Ogan, A., Shute, V. (eds.) Proceedings of the Workshop on Intelligent Educational Games at the 14th International Conference on Artificial Intelligence in Education, Brighton, UK, pp. 117–120 (2009)

[12] VanLehn, K.: The behavior of tutoring systems. International Journal of Artificial Intelligence in Education 16(3), 227–265 (2006)

[13] Koedinger, K.R., Aleven, V.: Exploring the assistance dilemma in experiments with cognitive tutors. Educational Psychology Review 19(3), 239–264 (2007)

[14] Easterday, M.W.: Policy world: A cognitive game for teaching deliberation. In: Pinkward, N., McLaren, B. (eds.) Educational Technologies for Teaching Argumentation Skills. Bentham Science Publishers, Oak Park (in press)

[15] Easterday, M.W., Aleven, V., Scheines, R., Carver, S.M.: Constructing causal diagrams to learn deliberation. International Journal of Artificial Intelligence in Education 19(4), 425–445 (2009)

[16] University of Rochester: Intrinsic motivation instrument, IMI (1994), `http://www.psych.rochester.edu/SDT/measures/IMI_description.php`

[17] 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, pp. 61–78. Cambridge University Press, Cambridge (2006)

[18] Corbett, A.T., Anderson, J.R.: Locus of feedback control in computer-based tutoring: Impact on learning rate, achievement and attitudes. In: Jacko, J., Sears, A., Beaudouin-Lafon, M., Jacob, R. (eds.) Proceedings of the ACM CHI 2001 Conference on Human Factors in Computing Systems, pp. 245–252. ACM Press, New York (2001)

[19] Nathan, M.J.: Knowledge and situational feedback in a learning environment for algebra story problem solving. Interactive Learning Environments 5(1), 135–159 (1998)

[20] Mathan, S.A., Koedinger, K.R.: Fostering the intelligent novice: Learning from errors with metacognitive tutoring. Educational Psychologist 40(4), 257–265 (2005)

[21] Aleven, V., Myers, E., Easterday, M., Ogan, A.: Toward a framework for the analysis and design of educational games. In: Biswas, G., Carr, D., Chee, Y.S., Hwang, W.Y. (eds.) Proceedings of the 3rd IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning, pp. 69–76. IEEE Computer Society, Los Alamitos (2010)

[22] Spirtes, P., Glymour, C., Scheines, R.: Causation, prediction, and search, 2nd edn. MIT Press, Cambridge (2000)

A Cognitive Tutoring Agent with Episodic and Causal Learning Capabilities

Usef Faghihi[1], Philippe Fournier-Viger[2], and Roger Nkambou[3]

[1] Dept. of Computer Sciences, University of Memphis
[2] Dept. of Computer Sciences, National Cheng-Kung University
[3] Dept. of Computer Sciences, Université du Québec à Montréal
{Usef.faghihi,philippe.fv}@gmail.com, nkambou@uqam.ca

Abstract. To mimic human tutor and provide optimal training, an intelligent tutoring agent should be able to continuously learn from its interactions with learners. Up to now, the learning capabilities of tutoring agents in educational systems have been generally very limited. In this paper, we address this issue with CELTS, a cognitive tutoring agent, whose architecture is inspired by the latest neuroscientific theories and unite several human learning capabilities such as episodic, emotional, procedural and causal learning.

Keywords: Cognitive agents, Episodic learning, Causal learning.

1 Introduction

An Intelligent Tutoring System (ITS) is an educational system that provides tailored assistance to learners without human intervention [1]. To be able to generate tailored assistance, an ITS typically relies on the "student model", a module dedicated to the evaluation of the learner. The student model can contain information about the domain knowledge that the learner is believed to possess and other information such as his/her perceived affective state. During a training session, an ITS will update the student model regularly to update its beliefs about the learner.

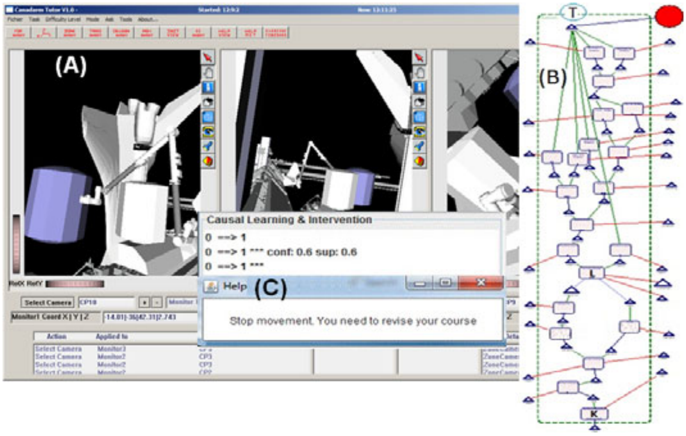
In this paper, we aim to broaden the adaptation capabilities of ITS by making the system learn from its interaction with learners. In this study, an ITS will adapt to learners mostly through interactions with them, just like human teachers. For this purpose, we propose Conscious Emotional Learning Tutoring System (CELTS) [2]. CELTS, in addition to its expert pre-defined "know how" knowledge, is an ITS equipped with human-like learning mechanisms. CELTS's learning mechanisms are based on the latest neuroscientific theories of cognition. It replicates three types of learning found in humans that are of benefit for tutoring tasks: episodic [3, 4], causal [5-8] and emotional learning [9-11]. The aforementioned learning and reasoning mechanisms give CELTS the capability of adapting its behavior according to previous experiences. They also help CELTS understand the cause of learner's mistakes during training sessions and allow it to assign emotional valences to the environment stimuli. Our emphasis in this study is on how CELTS can adapt its behavior to learners, and improve their learning.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 72–80, 2011.

In this paper, we first briefly explain the functioning of CELTS and its application domain. We then explain how we improved this agent by implementing the three aforementioned types of learning. We finally present an experimental evaluation of the agent's behavior with learners and report other experimentations to evaluate the algorithms.

2 CELTS

CELTS is a hybrid artificial intelligent tutor which is based on Baars' [12] theory of consciousness. It is integrated in CanadarmTutor [13], an ITS for learning to operate the Canadarm2 robotic arm installed on the international space station (ISS). CanadarmTutor is a simulation-based ITS offering a 3D reproduction of Canadarm2 on the space station and its control panel (cf. figure 1 (a)). Learning activities in CanadarmTutor mainly consists of operating Canadarm2 for performing various real-life tasks with the simulator. Operating Canadarm2 is a difficult task because astronauts have to follow a strict security protocol, the arm has seven-degrees of freedom and users only have a partial view of the environment through the cameras that they choose and adjust. CanadarmTutor integrates several research projects [13]. In this paper, we focus on CELTS, the component of CanadarmTutor that integrates all other components, that takes all the pedagogical decisions, generate dialogue and perform the high-level assessment of the learner.



```
0 ==> 1
0 ==> 1 *** conf: 0.6 sup: 0.6
0 ==> 1 ***
```

Stop movement. You need to revise your course

Fig. 1. (A) CanadarmTutor, (B) CELTS Behavior Network and (C) CELTS Feedback

CELTS performs through cognitive cycles. Cognitive cycles in CELTS start by perception and usually end by the execution of an action. CELTS uses its Behavior Network (BN) for action selection (Figure 1.B). The BN is implemented based on Maes' Behaviour Net [14]. It is a network of partial plans that analyses the context to decide what to do and which type of behavior to set off, and is represented as a graph (Figure 1.B). Given that CELTS is a tutor, an expert can define different solutions in the BN to help learners. Thus, BN's nodes are messages, hints, demonstration, etc.

(Figure 1.C) to assist learners while they manipulate Canadarm2 in the virtual environment (Figure 1.A). The learners' manipulations of the virtual world simulator, simulating Canadarm2 (Figure 1.A), constitute the interactions between them and CELTS. In particular, the virtual world simulator sends all manipulation data to CELTS, which, in turn, sends learners advices to improve their performance. Our team has now added different types of learning in CELTS based on neurobiology and neuropsychological theory [2].

In what follows, we briefly describe the Emotional, Episodic and Causal mechanisms added to CELTS.

2.1 Emotional Learning

The first type of learning is emotional learning [9-11]. In human being, emotions play a major role in learning, decision making and actions taken. However, emotions are an unclear concept that is not easily definable [15, 16]. Various definitions and very important responsibilities were given to emotions. Emotions allow us to adapt and accept new changes in our dynamic environment [4, 17].

In CELTS, we have added and Emotional Mechanism (EM) [2], which simulates the peripheral-central" theory of emotions. The peripheral-central approach takes into account both the short and long route information processing and reactions, as in humans: (i) short route is a quick but dumb (i.e., reflex-like) mechanism that prepares the agent to quickly act (pull away from or confidently approach a situation), and (ii) long route is the lasting modifications in workspace processing brought about by the variation in the valences assigned to all events as a result of the dumb specialist's processing [2]. Both the short and long routes perform in a parallel and complementary fashion in CELTS' architecture. The EM learns and at the same time contributes emotional valences (positive or negative) to the description of the situation. Valences for each type of stimuli are initially set by a random function, but are later adjusted automatically by CELTS so that stimuli are associated with positive/negative emotions when the learner shows a good/bad performance. The EM also contributes to the decisions made and the learning achieved by the system. In CELTS, Emotional learning assigns an emotional assessment to all environment's stimuli. In CELTS, the Emotional learning influences learning decision making [2]. For example, emotions play a very important role for remembering events from the episodic memory, as it is explained next.

2.2 Episodic Learning

The second type of learning is episodic learning, which consists of building an episodic memory (a memory of past events) to answer questions such as what, where and when [3, 4]. For a tutoring agent, episodic memory is crucial to know which interactions are successful with learners and which ones are not, and to use this information to improve its behavior. It also helps tutor to remember learners' mistakes.

To implement CELTS Episodic Learning mechanism (EPL), we used sequential pattern mining algorithms. EPL extracts frequently occurring events from its past experiences [2, 8]. In our context, CELTS learns during astronauts' training sessions for arm manipulation in the Canadarm2 simulator virtual world [13] (Figure 1.A). To construct CELTS' Episodic Memory, a trace of what occurred in the system is recorded in CELTS' different memories during consciousness broadcasts [2] as a sequence of events. Each event X= (*ti*, *Ai*) represents what happened during a cognitive cycle. The timestamp *ti* of an event indicates the cognitive cycle number, whereas the set of items *Ai* of an event contains an item that represents the coalition of information (e.g., collision risk with ISS) that were broadcast during the cognitive cycle. Each item can be associated with positive/negative emotions generated by EM. The memory consolidation process then periodically extracts frequent sub-sequences of events by using a custom sequential pattern mining algorithm (see Faghihi et al., 2010a for details). The resulting patterns constitute the episodic memory. CELTS use them to adapt its behavior by reusing "positive" patterns (carrying positive emotions) and avoiding "negative" patterns. For example, CELTS could reuse a sequence of tutoring interventions that successfully helped learners many times (bring positive emotions), while avoiding sequences that led to poor user learning.

2.3 Causal Learning

The third type of learning, causal learning, is to learn the causal relationships between events [5-7]. Human beings systematically construct their causal knowledge based on episodic memory [18-21].Given that episodic memory contains the memory of events and their outcomes, humans make inductive abstraction to construct causal relations between events. Thus, in humans, causal memory is influenced by the information retained by episodic memory. Inversely, new experiences are influenced by causal memory [18-21].

Up to now, few works have been done to incorporate causal learning in cognitive agents. Schoppek [22, 23] integrated one in ACT-R [24]. However, the model "*overestimates discrimination between old and new states*" and every assumption for the creation of a causal model must be detailed by a programmer. The casual learning incorporated in CLARION (Sun, 2006) requires someone to predefine information. Most of the researchers propose the use of Bayesian approach when it comes to causal learning, for instance Gopnik [25] use a Bayesian approach for the construction of knowledge. However, Bayesian approach needs experts to assign predefined values to variables, and this is often a very difficult and time-consuming task [26]. Bayesian networks have also been used in ITS but not for causal learning, for instance, it is used for the construction of beliefs about students in the student model (e.g. Pump Algebra Tutor, Andes, etc.; [1]). In the context of a tutoring agent like CELTS, the aforementioned limitation of Bayesian networks is a serious issue, because we wish that CELTS could learn and adapt its knowledge of causes automatically and without any human intervention. Another problem for Bayesian learning, crucial in the present context, is the risk of combinatory explosion in the case of large amounts of data. In the case of our agent, constant interaction with learners creates the large amount of data stored in CELTS modules. For this last reason, we believe that using data mining algorithms which is conceived for handling lot of data is more appropriate to implement a causal learning mechanism in CELTS.

To implement causal learning in CELTS we used a sequential rule mining algorithm [9, 33]. Each rule has the form X⇒Y, where X and Y are unordered sets of events. The interpretation of a rule is that if events from X occur, the events from Y are likely to follow. Two interesting measures are used for ranking rules (these measures are the most widely used in the rule mining literature): *support* and *confidence*. The support of a rule is defined as the number of sequences that contain the rule. The confidence of a rule is defined as the ratio between the number of sequences where the rule appears and the number of sequences containing its left part. This information can be interpreted as an estimate of the conditional probability P(Y | X) [27-29]. CELTS' Causal learning algorithm takes a database of event sequences (the episodic memory) and three parameters as inputs. The first parameter, *window_size,* defines the maximum time length in which a rule has to occur. In the context of CELTS, this parameter is very important as it allows CELTS to exclude rules between events that are separated by too much time. The *window_size* constraint is a global parameter that must be selected by an expert in CELTS. In our experiment for CELTS, we choose window_size = 20 cognitive cycles (which means about 5 seconds), because it seems to be a good value for finding causal relationships in our application domain. The second and third parameters are two thresholds: a minimum support and a minimum confidence threshold. CELTS' Causal learning algorithm then outputs the set of rules of the form X⇒Y that have a support and confidence no less than these thresholds, occurs within the maximum time length, and where $X \cap Y = \emptyset$. By extracting rules as explained above, respecting the temporal ordering of events we have demonstrated that CELTS is capable of inductive reasoning [8]. For a tutoring agent like CELTS, having a causal memory is an effective way of understanding the cause of learner's mistakes.

3 Experimental Evaluation

We evaluated CELTS from two angles. First, we performed an empirical evaluation with learners to evaluate: 1) the number of correct tutoring interventions, 2) the impact of these interventions on learners' performance, 3) the learners' satisfaction and 4) the correctness of the causal rules learned by CELTS. Second, we analyzed the performance of the data mining algorithms in CELTS and their scalability on larger random databases.

3.1 Evaluation with Learners

To determine the extent to which the three aforementioned learning mechanisms improved CELTS' performance, we asked eight users to test the new version of CELTS. This new version of CELTS is allows for its use with learning mechanisms (version A) and its use without learning mechanisms (version B). Learners were invited to manipulate Canadarm2 for approximately 1 hour, using both versions A and B of the system. The first four students (group A) used version A, and then version B. The second four learners (group B), first used version B and then version A. After its interactions with the users, CELTS categorized them into novices, intermediate and experts. During the experiments with version B, CELTS automatically learned more than 2400 rules from its interactions with learners. A few examples of rules are found below:

- 42% of the time, when the learner was not aware of distances and Canadarm2 was closed to the ISS, there was a collision risk:
  {Canadarm2_NearISS, Not_aware_of_distance}⇒ {Collision risk }
- 51% of the time, when a user forgot to adjust the camera, he/she later chose an incorrect joint of Canadarm2: { Forget_adjust_Camera}⇒{Bad_joint}

- 10% of the time, when the user moved Canadarm2 without adjusting the camera, he/she increase the risk of collisions:
  { Move_Canadarm2}, { Forget_adjust_Camera}, ⇒{ Collision risk }
- 10% of the users who manipulated Canadarm2 close to the space station, being aware of the distance and having reached the goal, were classified as experts :
  {Canadarm2_Near_ISS, Aware_of_distance, Goal_attained} ⇒ {Expert}.

Such rules are then used by CELTS as described in the Behavior Network (Figure 1.B) to adapt its behavior. To do so, during each cognitive cycle, CELTS checks which rules match with its current execution. If several rules match the current execution, the one having the most strength is used for prediction. The strength of a rule is defined as: *Strength(rule) = Confidence(rule) * Support(rule).*

To assure the quality of the rules found by CELTS, we asked a domain expert to evaluate them. Given that checking all rules one by one would be tedious, the expert examined 150 rules from the 2400+ recorded rules. Overall, the expert confirmed the correctness of about 85 % of the rules. Furthermore, from the found rules, many unexpected rules (e.g., correct) were discovered.

To evaluate to which extent the integration of the learning mechanisms impacted the performance of the learners, we measured four performance indicators during the usage of Version A and Version B of CanadarmTutor by group A and group B: (1) the percentage of questions that they answered correctly (2) the average time that they took to complete each exercise in minutes, (3)) the mean number of collision risks incurred by learners , and (4) the mean number of violations of the security protocols committed during the exercises. Figure 2 illustrates the results: (1) Group A correctly answered 50% of the questions, whereas Group B correctly answered 30% of the questions ; (2) Group A took an average of 1.45 minutes to complete an exercise whereas Group B took an average of 2.13 minutes, (3) Group A incurred an average of 3collision risks made by learners , whereas Group B incurred an average of 5 collision risks made by learners; (4) Group A had an average of 10 violated protocols whereas Group B had an average of 18 violated protocols. Although we have not used a very large number of learners in this experiment, from these results, we can see that the performance of the learners who used the new version of CELTS clearly improved.

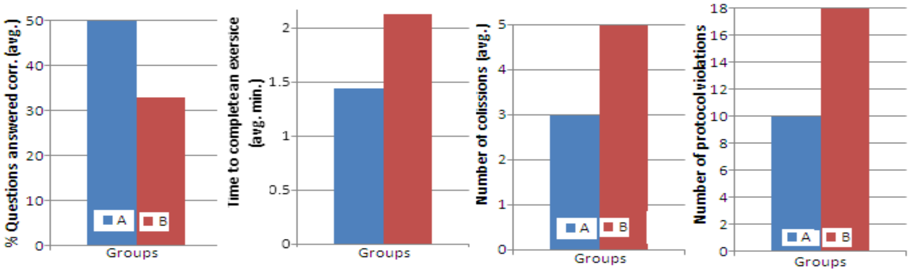


Fig. 2. Learners' Performance Comparison

Furthermore, we analyzed the correctness of the CELTS' hints and messages to the learners during tasks. To determine if an intervention was correct, we asked learners to rate each of CELTS' interventions as being appropriate or inappropriate. Because learners could incorrectly rate the tutor's interventions, we also observed the training sessions and verified the ratings given by each learner one by one subsequently. The results show that the average number of appropriate interventions was about 83 % using version A and 58 % using version B. This is also a considerable improvement in CELTS' performance over its previous version.

We also assessed the user satisfaction by performing a 10 minute post-experiment interview with each user. We asked each participant to tell us which version of CELTS they preferred, to explain why, and to tell us what should be improved. Users unanimously preferred the new version. Some comments given by users were: 1) the tutoring agent "exhibited a more intelligent and natural behavior"; 2) the "interactions were more varied"; 3) the "tutoring agent seems more flexible"; 4) "in general, gives a more appropriate feed-back". There were also several comments on how CELTS could be improved. In particular, many users expressed the need to give CELTS a larger knowledge base for generating dialogues. We plan to address this issue in future work.

3.2 Performance of the Data Mining Algorithms

In our previous experiments, we have also measured the execution time of our customized data mining algorithms used in CELTS to determine whether learners' performance was an issue. The execution time was always very good in our experiments. On average, it extracted causal rules and sequential patterns from approximately 500 sequences in less than 50 ms.

To test the scalability of the algorithms, we generated 20 000 sequences of events automatically. The sequences were generated by randomly answering the questions asked by CELTS during Canadarm2 manipulation. The Data mining algorithms were applied to the sequences in real time to extract useful information and the cause of the event (causal rules). The performance has been very good, the algorithms terminating in less than 100 seconds with the following parameters: a minimum support of 0.05 and a minimum confidence of 0.3. This demonstrates the scalability of the algorithms for much larger amounts of data than have been recorded in CELTS during our experiments with users. This performance is comparable to the performance of other large-scale frequent pattern mining algorithms in the literature which can often handle hundreds of thousands of sequences or transactions [30].

4 Conclusions

In this paper, we examined CELTS' performance by giving it the capabilities of learning from its interactions with learners so that it can better adapt to the learners. CELTS is equipped with three types of learning: a) emotional, b) episodic, and c) causal learning. In this study, both the new and the previous version of CELTS are integrated in the CanadarmTutor. We have evaluated the new version of CELTS in five ways. First, CELTS' performance was evaluated based on the number of correct interventions given to the learners during training sessions. Second, results showed that the new version has a considerable impact on learners' performance. Third, we evaluated the satisfaction of users. Fourth, a domain expert examined the correctness of the causal rules learned by CELTS. Fifth, experiments confirmed the performance and scalability of the data mining algorithms used in CELTS. In the future, we plan to further improve CELTS' algorithms, the pedagogical strategies used by CELTS and its dialogue generation module.

References

1. Woolf, B.P.: Building Intelligent Interactive Tutors. Student Centered Strategies for revolutionizing e-learning. Morgan Kaufmann, Massachusetts (2009)
2. Faghihi, U., poirier, P., Fournier-Viger, P., Nkambou, R.: Human-Like Learning in a Conscious Agent. Journal of Experimental & Theoretical Artificial Intelligence (2010) (in Press)
3. Tulving, E.: Precis of Elements of Episodic Memory. Behavioural and Brain Sciences 7, 223–268 (1984)
4. Purves, D., Brannon, E., Cabeza, R., Huettel, S.A., LaBar, K., Platt, M., Woldorff, M.: Principles of cognitive neuroscience. In: First, E. (ed.), Sunderland. Sinauer Associates, Massachusetts (2008)
5. Gopnik, A., Schulz, L. (eds.): Causal Learning: Psychology, Philosophy, and Computation. Oxford University Press, USA (2007)
6. Maldonado, A., Catena, A., Perales, J.C., Cándido, A.: Cognitive Biases in Human Causal Learning (2007)
7. Brauny, M., Rosenstiel, W., Schubert, K.-D.: Comparison of Bayesian Networks and Data Mining for Coverage Directed Verification. IEEE, Los Alamitos (2003)
8. Faghihi, U., Fournier-Viger, P., Nkambou, R., Poirier, P.: A Generic Causal Learning Model for Cognitive Agent. In: The Twenty Third International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems, IEA-AIE 2010 (2010)
9. LeDoux, J.E.: Emotion circuits in the brain. Annu. Rev. Neurosci. 2000 23, 155–184 (2000)
10. Morén, J.: Emotion and learning: a computational model of the amygdala. Lund University, Lund (2002)
11. Phelps, E.A.: Emotion and Cognition: Insights from studies of the human amygdala. Annual Review of Psychology 57, 27–53 (2006)
12. Baars, B.J.: In the Theater of Consciousness: The Workspace of the Mind. Oxford University Press, Oxford (1997)
13. Nkambou, R., Belghith, K., Kabanza, F.: An approach to intelligent training on a robotic simulator using an innovative path-planner. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 645–654. Springer, Heidelberg (2006)
14. Maes, P.: How to do the right thing. Connection Science 1, 291–323 (1989)
15. Thompson, R.F., Madigan, S.A.: Memory: The Key to Consciousness. Princeton University Press, Princeton (2007)
16. Alvarado, N., Adams, S., Burbeck, S.: The Role Of Emotion In An Architecture Of Mind. IBM Research (2002)
17. Damasio, A.R.: Looking for Spinoza: Joy, Sorrow and the Feeling Brain. Harcourt Inc., New York (2003)
18. Martin, C.B., Deutscher, M.: Remembering. Philosophical Review 75, 161–196 (1966)
19. Shoemaker, S.: Persons and their Pasts. American Philosophical Quarterly 7, 269–285 (1970)

20. Perner, J.: Memory and Theory of Mind. In: Tulving, E., Craik, F.I.M. (eds.) The Oxford Handbook of Memory, pp. 297–312. Oxford University Press, Oxford (2000)
21. Bernecker, S.: The Metaphysics of Memory. Springer, Berlin (2008)
22. Schoppek, W.: Stochastic Independence between Recognition and Completion of Spatial Patterns as a Function of Causal Interpretation. In: Proceedings of the 24th Annual Conference of the Cognitive Science Society (2002)
23. Schoppek, W.: Stochastic independence between recognition and completion of spatial patterns as a function of causal interpretation. In: Proceedings of the 24th Annual Conference of the Cognitive Science Society, pp. 804–809. Erlbaum, Mahwah (2002)
24. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., lebiere, C., Qin, Y.: An integrated theory of the mind. Psychological Review 111(4), 1036–1060 (2004)
25. Gopnik, A., Glymour, C., Sobel, D.M., Schulz, L.E., Kushnir, T., Danks, D.: A Theory of Causal Learning in Children: Causal Maps and Bayes Nets. Psychological Review 111(1) (2004)
26. Braun, M., Rosenstiel, W., Schubert, K.-D.: Comparison of Bayesian networks and data mining for coverage directed verification category simulation-based verification. In: Eighth IEEE International, High-Level Design Validation and Test Workshop, 2003, pp. 91–95 (2003)
27. Hipp, J., Güntzer, U., Nakhaeizadeh, G.: Data Mining of Association Rules and the Process of Knowledge Discovery in Databases. In: Industrial Conference on Data Mining, pp. 15–36 (2002)
28. Deogun, J.S., Jiang, L.: Prediction Mining – An Approach to Mining Association Rules for Prediction. In: Ślęzak, D., Yao, J., Peters, J.F., Ziarko, W.P., Hu, X. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3642, pp. 98–108. Springer, Heidelberg (2005)
29. Li, L., Deogun, J.S.: Discovering Partial Periodic Sequential Association Rules with Time Lag in Multiple Sequences for Prediction. In: Hacid, M.-S., Murray, N.V., Raś, Z.W., Tsumoto, S. (eds.) ISMIS 2005. LNCS (LNAI), vol. 3488, pp. 332–341. Springer, Heidelberg (2005)
30. Fournier-viger, P., Nkambou, R., Tseng, V.S.: RuleGrowth: Mining Sequential Rules Common to Several Sequences by Pattern-Growth. In: Proceedings of the 26th Symposium on Applied Computing (ACM SAC 2011), pp. 954–959 (2011)

When Does Disengagement Correlate with Learning in Spoken Dialog Computer Tutoring?

Kate Forbes-Riley and Diane Litman

Learning R&D Ctr, University of Pittsburgh, Pittsburgh, PA 15260
{forbesk,litman}@cs.pitt.edu

Abstract. We investigate whether an overall student disengagement label and six different labels of disengagement type are predictive of learning in a spoken dialog computer tutoring corpus. Our results show first that although students' percentage of overall disengaged turns negatively correlates with the amount they learn, the individual types of disengagement correlate differently with learning: some negatively correlate with learning, while others don't correlate with learning at all. Second, we show that these relationships change somewhat depending on student prerequisite knowledge level. Third, we show that using multiple disengagement types to predict learning improves predictive power. Overall, our results suggest that although adapting to disengagement should improve learning, maximizing learning requires different system interventions depending on disengagement type.

Keywords: types of disengagement, learning, correlations, spoken dialog computer tutors, manual annotation, natural language processing.

1 Introduction

The last decade has seen a significant increase in computer tutoring research aimed at improving student learning (and other performance metrics) by tailoring system responses to changing student affect and attitudes, over and above correctness. Student (dis)engagement behaviors have been of particular interest in this research, including displays of gaming, boredom, indifference, (lack of) interest, (low) motivation, curiosity, and flow (e.g. [7,3,9,11,12]). Correlational analyses of student (dis)engagement behaviors in tutoring system corpora have indicated that these behaviors are predictive of learning. For example, gaming [3,1] and boredom [9] have been associated with decreased learning during computer tutoring, while flow [9] and engagement [4] have been associated with increased learning. In addition, a number of automatic gaming detectors have been implemented and evaluated in computer tutors, with results indicating that gaming behaviors can be reliably detected in real-time using features of the tutoring interaction (cf. [3]). Moreover, controlled experiments using gaming-adaptive computer tutors - i.e., tutors enhanced with interventions that target student gaming - have shown that adapting to gaming can improve student learning [2,3] or other performance metrics (such as reducing gaming) [13,1].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 81–89, 2011.

Our own research builds on this prior work, with the larger goal of enhancing our spoken dialog computer tutor to automatically detect and respond to student disengagement over and above correctness and uncertainty,[1] and thereby improve learning and other metrics. However, in contrast to prior work, which has focused on detecting and adapting to only one or two disengagement behaviors (typically gaming), our goal is to detect and respond differently to a wider range of student disengagement, with the system interventions differing depending on the *type* of disengagement. Our work is also novel in that it focuses on spoken language-based disengagement displays. Working towards our end goal, in prior work we developed and evaluated an annotation scheme for manually labeling an overall measure of disengagement, as well as different types of disengagement, in our spoken dialog computer tutoring corpora (Section 2).

In this paper, we extend the results of others' prior work correlating disengagement behaviors and learning (Section 3). First, we show that although our overall measure of disengagement is predictive of decreased student learning in our spoken dialog computer tutoring corpus, different types of disengagement correlate *differently* with learning: some negatively correlate, while others don't correlate at all. Furthermore, the amount of prerequisite knowledge a student has changes these relationships somewhat. Finally, we show that using multiple disengagement types to predict learning improves predictive power. Importantly, our results suggest that while adapting to an overall measure of disengagement can improve student learning, maximally improving learning requires different system interventions depending on the type of disengagement.

2 Computer Tutoring Disengagement Data

Our research is performed on a corpus of spoken dialogs from a controlled experiment evaluating an uncertainty-adaptive version of our tutoring system, ITSPOKE (**I**ntelligent **T**utoring **SPOKE**n dialog system), which is a speech-enhanced and otherwise modified version of the Why2-Atlas qualitative physics tutor (cf. [6]). The experimental procedure was as follows: college students with no college-level physics (1) read a short physics text, (2) took a multiple choice pretest, (3) worked with ITSPOKE, (4) took a survey, and (5) took an isomorphic posttest. The resulting corpus contains 360 spoken dialogs (5 per student) from 72 students (6044 student turns). Figure 1 shows a corpus example.

Briefly, ITSPOKE tutors 5 physics problems (one per dialog), using a Tutor Question - Student Answer - Tutor Response format. After each tutor question, the student speech is sent to the Sphinx2 recognizer, which yields an automatic transcript. This answer's (in)correctness is then automatically classified based on this transcript, using the TuTalk semantic analyzer [8], and the answer's (un)certainty is automatically classified by inputting features of the speech signal, the automatic transcript, and the dialog context into a logistic regression

[1] As discussed further elsewhere, our current system already adapts to student uncertainty over and above correctness; our goal is thus to enhance this system to adapt to multiple affective states (disengagement and uncertainty) [7].

model. The appropriate tutor response is determined based on the answer's (in)correctness and (un)certainty and then sent to the Cepstral text-to-speech system, whose audio output is played through the student headphones and is also displayed on a web-based interface. See [6] for details.

Our disengagement annotation scheme is empirically derived from observations in our data but draws on prior work, including appraisal theory-based emotion models, which also distinguish emotional behaviors from their underlying causes (e.g.,[5])[2], as well as prior approaches to manually annotating disengagement or related states in tutoring corpora [9,11,12]). Our inter-annotator reliability evaluation on a corpus subset showed that our overall disengagement label (0.55 Kappa) and disengagement type labels (0.43 Kappa) can be annotated with moderate reliability on par with prior emotion annotation work [7]. For the current analysis, all student turns in the corpus were manually annotated as summarized below. See [7] for full details of the annotation scheme:

An **overall Disengagement label (DISE)** was used for all turns expressing moderate to strong disengagement in the tutoring process, i.e., answers given without much effort or without caring about correctness. Answers might also be accompanied by signs of inattention, boredom, or irritation. Clear examples include answers spoken quickly in leaden monotone or with sarcastic or playful tones, or with off-task sounds such as rhythmic tapping or electronics usage.[3]

One of the six **Disengagement Type labels** summarized below accompanied each DISE label. These labels represent the (inferred) underlying causes of disengagement *as well as* the behavior and context evidencing them. In particular, they distinguish different student reactions to the system's limited natural language processing abilities (NLP-Distracted/NLP-Gaming), different student perceptions of the tutoring material (Easy/Hard/Presentation), and a "catch-all" category for other student reactions as the session progresses (Done).

NLP-Distracted: Student became distracted and hyperarticulated[4] this answer because the system misunderstood an immediately prior answer due to its limited natural language processing capabilities.

Hard: Student lost interest because this tutor question was too hard (e.g., presupposes too much prior knowledge).

NLP-Gaming: Student didn't try to work out the answer to this tutor question; s/he instead deliberately gave a vague or incorrect answer or a guess to try and fool the system's limited natural language processing capabilities.

[2] Appraisal theories argue that one's appraisal of a situation causes emotion; i.e., emotions result from (and don't occur without) an evaluation of a context (e.g.,[5]).

[3] Affective systems research has found that total disengagement is rare in laboratory settings (e.g., [7,9]). As in that research, we thus equate the "disengagement" label with either no or low engagement. Since total disengagement is common in real-world unobserved human-computer interactions (e.g., deleting unsatisfactory software), it remains an open question as to how well laboratory-based findings generalize.

[4] That is, gives the answer with unnatural pitch, cadence, stress, or loudness in an attempt to make the computer better understand him/her. This label was renamed from "Language" in our prior work [7] for clarity.

Presentation: Student didn't pay attention to this tutor question because the presentation was too long or complex; his/her answer reflects unawareness of the fact that the tutor turn strongly hinted at the correct answer.

Easy: Student lost interest because this tutor question was too easy (e.g., a similar question was asked and answered earlier in the session).

Done: Student just wants the interaction to be over (typically later in the dialogs) - s/he is bored, tired, and/or not interested in continuing at this moment (or no other label fits).

This scheme should generalize to other learning environments, including analogs of NLP-Gaming and NLP-Distraction, since these two types represent two disengagement behaviors stemming from a system's inherent interaction processing inflexibility, which exists regardless of the communication medium.[5]

T$_9$: What's the numerical value of the man's acceleration? Please specify the units too.

S$_9$: The speed of the elevator. Meters per second. **(DISE: NLP-Gaming)**

. . .

T$_{15}$: What is the definition of Newton's Second Law?

S$_{15}$: I have no idea <*sigh*>. **(DISE: Hard)**

. . .

T$_{21}$: Based on our discussion, we conclude that the keys will remain in front of the man's face during the entire fall. [...] Would you like to do another problem?

S$_{21}$: No <*laugh*>. **(DISE: Done)**

Fig. 1. Corpus Example Illustrating Disengagement Annotation Scheme

Figure 1 illustrates the scheme. **S**$_9$ is labeled DISE with the NLP-Gaming Type because the student avoided giving a specific numerical value, offering instead a vague (and incorrect) answer. **S**$_{15}$ is labeled DISE with the Hard Type because the student gave up immediately and with irritation when too much prior knowledge was required. **S**$_{21}$ is labeled DISE with the Done Type because the student answered 'No' semi-jokingly in regards to continuing the experiment.

Note that our NLP-Gaming label represents a subset of the gaming behaviors addressed in prior work (Section 1), which focuses on hint abuse and systematic guessing.[6] ITSPOKE does not provide hints upon request, and the dialog is the only recorded behavior, thus all detectable gaming behavior is linguistic. Altogether our Disengagement types label a range of behaviors associated with disengagement, including off-task, bored, or low-motivated actions that don't attempt to exploit the system. Moreover, our labels capture the fact that these behaviors can be associated with different underlying causes. E.g., a student who disengages because a question is too hard may exhibit any of these behaviors.

[5] NLP-Distraction differs from the other types in that although students do lose the tutoring flow, this is not of their own (un)conscious volition.

[6] This prior work defines gaming as attempting to succeed by exploiting the system rather than learning the material and using that knowledge to answer correctly [3].

Note finally that this turn-level annotation scheme captures both fleeting disengagement states as well as long-term disengagement escalation across turns.

3 Prediction Results

To investigate whether our overall disengagement (DISE) and disengagement type labels are predictive of learning in our corpus, we computed the percentage of each label's occurrence for each student, and the partial Pearson's correlation between the percentage and posttest score, controlling for pretest to account for learning gain. Table 1 shows first the mean percentage (Mn%) and its standard deviation (sd) over all students, the Pearson's Correlation coefficient (R) and significance (p) with significant results bolded ($p \leq 0.05$), and the total number of occurrences (Tot) for each label in the entire dataset. These statistics are then provided for students with low and high pretest scores (see below). The last two rows show test scores for each group (Mn% and sd).

Table 1. Correlation Results between Disengagement or Disengagement Types and Learning in the ITSPOKE Corpus (N=72; Low Pretests N=40; High Pretests N=32)

	All Students			Low Pretests			High Pretests		
Measure	Mn%(sd)	R(p)	Tot	Mn%(sd)	R(p)	Tot	Mn%(sd)	R(p)	Tot
DISE	14.5(8.2)	**-.33(.01)**	886	16.2(8.3)	**-.37(.02)**	555	12.2(7.7)	-.26(.15)	331
NLPDistract	0.4(1.4)	-.03(.78)	28	0.6(1.8)	-.07(.68)	22	0.2(0.8)	.04(.81)	6
Hard	2.8(2.9)	**-.36(.01)**	172	3.6(3.3)	**-.35(.03)**	124	1.7(2.0)	**-.46(.01)**	48
NLPGame	3.0(3.0)	**-.34(.01)**	186	3.2(2.8)	**-.31(.05)**	108	2.9(3.2)	**-.39(.03)**	78
Easy	1.4(2.6)	.12(.33)	83	1.1(2.0)	-.02(.92)	36	1.8(3.2)	.30(.11)	47
Present	3.0(2.2)	**-.27(.02)**	182	3.6(2.1)	-.22(.17)	124	2.1(2.0)	**-.35(.05)**	58
Done	3.9(3.2)	-.08(.52)	235	4.2(3.2)	-.11(.53)	141	3.5(3.3)	-.04(.85)	94
Pretest	51.0(14.5)			40.5(7.8)			64.1(9.2)		
Posttest	73.1(13.8)			66.9(12.8)			80.8(10.9)		

Considering the results over all students, comparison of means shows that of the 14.5% overall disengaged turns on average per student, Done is the most frequent type of disengagement, followed by NLP-Gaming and Presentation, Hard, Easy, and NLP-Distracted. Since Done is defined as a "catch-all" category, it is not surprising that it is the most frequent; that it occurs only slightly more than three of the other types suggests that our six categories are sufficiently representative of the range of disengagement behaviors (and underlying causes) in our data. The high standard deviations suggest that the amount of overall DISE, and the disengagement types, are highly student-dependent.

The correlation results over all students show that overall DISE is significantly correlated with decreased learning. This supports prior work (Section 1) showing negative relationships between learning and boredom or gaming. Our results also show significant negative correlations between learning and the Hard, NLP-Gaming, and Presentation Types. This suggests that the negative DISE correlation is primarily due to these three types. Prior work suggests that gaming behaviors associated with poorer learning often occur when students lack the knowledge to answer the question [3,2].[7] Similarly, we hypothesize that students often exhibited linguistic (NLP) gaming in our corpus because the system's limited natural language processing abilities prevented them from eliciting information they needed to answer the question. Together, the results for the NLP-Gaming and Hard Types suggest that if not remediated, disengagement can negatively impact learning when it is caused by questions presupposing knowledge the student doesn't have. Relatedly, the negative Presentation correlation suggests that if not remediated, disengagement can also negatively impact learning when it is caused by the inflexibility of the system's half of the dialog.

There are no significant correlations over all students for the NLP-Distracted, Easy, or Done Types. This indicates that student disengagement during tutoring is not always negatively related to learning. In particular, although some students may get distracted and irritated by system misunderstandings, this (NLP-Distracted) is not associated with decreased learning. Of course, the NLP-Distracted Type was very rare in our corpus; more frequent occurrences may impact learning. In addition, although some students may temporarily lose interest when a tutor question is too easy, this (Easy) is not associated with decreased learning. This result supports prior work suggesting that disengagement behaviors in highly knowledgeable students may have little relation to learning, while the same behavior in students with low prerequisite knowledge is associated with poorer learning [3]. Of course, our subjects were all novices; a very high proportion of easy questions is more likely to be associated with poor learning. Interestingly, the lack of a negative Done correlation suggests that temporary losses of student interest that occur as the tutoring dialog or session nears its end (or for other unclear reasons) are also not related to poorer learning.

To further investigate how students' prerequisite knowledge level impacts the relationship between disengagement behavior and learning in our data, we split students into high (N=32) and low (N=40) groups based on their mean pretest score,[8] and then reran the correlations on each group individually.

Comparison of means in Table 1 shows similar relative frequencies of the types across both groups: Done, Presentation and NLP-Gaming occur most often, and NLP-Distracted least often. However, the relative frequencies of Hard and Easy differ depending on knowledge level. Comparing absolute frequencies, one-way ANOVAs showed that only DISE, Hard, and Presentation differed significantly across the two groups ($p<.05$), occurring more for low pretesters.

[7] Other suggested reasons for gaming in this prior work include a performance-based mentality (as opposed to learning-based) and low motivation to learn.

[8] We didn't use a median split because it placed the same score in both groups. A T-test showed the two groups represent different populations ($p<.001$). Also note that while a repeated test-measure ANOVA has indicated that all students learned during the tutoring ($F(1,69) = 225.688$, $p<0.001$) [6], a one-way ANOVA showed no difference in normalized learning gain between the high and low pretest groups.

Regarding the correlations, neither group patterned identically to the combined group. The low pretest group did not show the negative correlation between learning and Presentation, while the high pretest group did not show it for overall DISE. It may be that students with high prerequisite knowledge are most sensitive to the way the system presents the material, i.e., are more likely to disengage and stop learning if they have difficulty immediately understanding the presentation. Although not quite a trend, the positive Easy correlation appears to counterbalance the negative correlations in the high pretest group, perhaps explaining the lack of an overall DISE correlation. Interestingly, and in contrast to prior work, our results suggest that NLP-Gaming negatively impacts learning regardless of prerequisite knowledge. This may be because prior work focused on hint abuse and systematic guessing, which are gaming methods targeted at manipulating the system into giving the correct answer. In contrast, students don't know whether NLP-Gaming will result in the correct answer.

Finally, after examining how each disengagement metric predicts learning in isolation, we investigated their relative usefulness in a more complex learning model. We used stepwise linear regression to predict posttest, allowing the model to select its inputs from pretest and our seven disengagement metrics. The following model yielded the best significant training fit to our data (R^2=.49, p<.001). As shown, two disengagement types were incorporated along with pretest. The (standardized) feature weights indicate relative predictive power in accounting for posttest variance. As shown, the Hard Type (p<.01) is more predictive of decreased posttest than the Presentation Type (p=.03), but both work together to significantly increase the model's predictive power over pretest alone.

$$\textbf{Posttest} = .41^{*}\textbf{Pretest} - .28^{*}\textbf{\%Hard} - .21^{*}\textbf{\%Presentation}$$

4 Current Directions

We extended prior research by investigating how overall disengagement (DISE) and its subtypes relate to learning in spoken dialog computer tutoring. We showed that overall DISE negatively correlates with learning, as do the Hard, Presentation, and NLP-Gaming Types, but the NLP-Distracted, Easy and Done Types do not. We showed that prerequisite knowledge level impacts these relationships: only high pretesters exhibit the Presentation correlation, while only low pretesters exhibit the DISE correlation. We also showed that using both the Hard and Presentation Types to model learning improves predictive power.

These results are now impacting our next step: enhancing ITSPOKE to adapt to disengagement. They suggest that maximizing learning requires different adaptations depending on DISE type. Thus we are now using machine learning to automatically recognize DISE types, based on linguistic features (e.g., acoustic-prosodic, lexical and dialog) previously used to predict affect in speech (cf. [6]), and system-specific features (e.g., correctness, timing, knowledge level, and question difficulty) previously used to predict gaming (e.g., [3,2,13,4]).

Our adaptations assume that identifying the causes of affect can help determine how to best respond (cf. [5]). They build on our current results and on prior evaluations of gaming adaptations in computer tutors that involved preventing gaming (e.g., [13,4,10,1]), metacognitive feedback about better ways to learn [2,13,1], easier exercises focusing on the gamed material [3], and performance feedback reminding students of task value [2,13]. Our current results suggest that the Easy, NLP-Distracted and Done Types should receive minimal, non-invasive interventions; they don't impact learning (at least at current levels), thus their adaptation should aim to reduce disengagement without reducing learning (e.g., via metacognitive and performance feedback). Since the Hard, NLP-Gaming, and Presentation Types negatively correlate with learning and involve a lack of understanding of the tutor question, they require more substantial interventions (e.g., feedback to promote re-engagement and an easier version of the question).

Acknowledgments

This work is funded by National Science Foundation (NSF) award #0914615 and #0631930. We thank Art Ward and our anonymous reviewers for comments.

References

1. Aleven, V., McLaren, B., Roll, I., Koedinger, K.: Toward tutoring help seeking: Applying cognitive modeling to meta-cognitive skills. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 227–239. Springer, Heidelberg (2004)
2. Arroyo, I., Ferguson, K., Johns, J., Dragon, T., Merheranian, H., Fisher, D., Barto, A., Mahadevan, S., Woolf, B.: Repairing disengagement with non-invasive interventions. In: Proc. Artificial Intelligence in Education (AIED), pp. 195–202 (2007)
3. Baker, R.S., Corbett, A., Roll, I., Koedinger, K.: Developing a generalizable detector of when students game the system. User Modeling and User-Adapted Interaction (UMUAI) 18(3), 287–314 (2008)
4. Beck, J.: Engagement tracking: using response times to model student disengagement. In: Proceedings of the 12th International Conference on Artificial Intelligence in Education (AIED), Amsterdam, pp. 88–95 (2005)
5. 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)
6. Forbes-Riley, K., Litman, D.: Benefits and challenges of real-time uncertainty detection and adaptation in a spoken dialogue computer tutor. Speech Communication (2011) (in press)
7. Forbes-Riley, K., Litman, D.: Annotating disengagement for spoken dialogue computer tutoring. In: D'Mello, S., Calvo, R. (eds.) Affect and Learning Technologies (to appear, 2011)
8. Jordan, P., Hall, B., Ringenberg, M., Cui, Y., Rose, C.: Tools for authoring a dialogue agent that participates in learning studies. In: Proc. Artificial Intelligence in Education (2007)

9. 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)
10. Murray, R.C.: vanLehn, K.: Effects of dissuading unnecessary help requests while providing proactive help. In: Proc. of the International Conference on Artificial Intelligence in Education, pp. 887–889 (2005)
11. Porayska-Pomsta, K., Mavrikis, M., Pain, H.: Diagnosing and acting on student affect: the tutor's perspective. User Modeling and User-Adapted Interaction: The Journal of Personalization Research 18, 125–173 (2008)
12. de Vicente, A., Pain, H.: Informing the detection of the students' motivational state: An empirical study. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 933–943. Springer, Heidelberg (2002)
13. Walonoski, J., Heffernan, N.: Prevention 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. 722–724. Springer, Heidelberg (2006)

Peering Inside Peer Review with Bayesian Models

Ilya M. Goldin and Kevin D. Ashley

Intelligent Systems Program and Learning Research and Development Center
University of Pittsburgh, Pittsburgh, PA, USA 15260
{goldin,ashley}@pitt.edu

Abstract. Instructors and students would benefit more from computer-supported peer review, if instructors received information on how well students have understood the conceptual issues underlying the writing assignment. Our aim is to provide instructors with an evaluation of both the students and the criteria that students used to assess each other. Here we develop and evaluate several hierarchical Bayesian models relating instructor scores of student essays to peer scores based on two peer assessment rubrics. We examine model fit and show how pooling across students and different representations of rating criteria affect model fit and how they reveal information about student writing and assessment criteria. Finally, we suggest how our Bayesian models may be used by an instructor or an ITS.

Keywords: computer-supported peer review, evaluation of assessment criteria, Bayesian models.

1 Introduction

Increasingly, instructors are turning to peer review as a teaching aid. [1, 2] Peer review has important benefits beyond shifting some of the burden of assessment from the instructor to students. By giving and receiving feedback from peers, students may improve their own work, and practice a useful professional skill. If instructors state their criteria explicitly, this helps make assessment rigorous and objective. Students can focus on these criteria as they write their essays and evaluate peer work. By spending less time on assessment, instructors can help struggling students. It has been shown that combined evaluations from multiple reviewers estimate essay quality reliably, and that students may respond better to feedback from peers rather than the instructor. [3, 4] Perhaps, most importantly, peer review enables instructors to assign writing exercises they might not otherwise assign for lack of time to prepare in-depth critiques, especially in very large classes.

Instructors and students would benefit even more, if peer reviewers used assessment rubrics that were conducive to all of the potential benefits. Rubrics are the heart of assessment. If reviewers assess aspects of peer work that are wrong for the exercise, then feedback will not be beneficial to authors, and the reviewers will have wasted their time providing it. Generic review criteria such as “flow”, “logic”, and “insight” [6] may be appropriate for some writing exercises, but not all. Peer feedback may be solicited in a structured way on the issues raised in an assignment (i.e., with

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 90–97, 2011.

problem-specific support to reviewers) and on more general but still domain-relevant aspects of the writing (domain-relevant support). [8]

A peer review system that prompts reviewers to assess authors' understanding of specific conceptual issues may provide aggregate estimates of students' grasp of these issues in the peer-review exercise thanks to a statistical model. The estimates are based on the feedback that students give to each other when in peer review. Gathering independent perspectives of multiple peer reviewers increases reliability. Modeling students' exchange of feedback may provide an instructor with a more informed view concerning how well students have grasped conceptual issues in a writing exercise. The modeling may also evaluate the criteria that students used to assess each other. Problems with peer assessment criteria may be indicative of curricular issues (e.g., if material is not covered in an optimal sequence), or of student comprehension of the criteria. Ultimately, the modeling aims to make peer-review exercise transparent to the instructor, who can use the information to guide and modify his or her teaching and make appropriate midcourse adjustments to the curriculum.

We compare our models on two types of peer assessment criteria. The only inputs to the model are the instructor's and the peers' scores of student work. We begin by describing computer-supported peer review and some artifacts of the peer review process, and explain how they may be related to assessment. We then develop several hierarchical Bayesian models relating these artifacts, and evaluate the models. Finally, we discuss the lessons learned from this modeling and explain how a Bayesian model may be used by an instructor or an Intelligent Tutoring System. We leave the actual generation and evaluation of reports to instructors for future work.

2 Study, Methods, and Data Sets

We report a new analysis of the datasets described in [8]. All 58 participants were second or third year law students in a course on Intellectual Property law. Students were required to take an open-book, take-home midterm exam and to participate in the subsequent peer-review exercise. The exam comprised one essay-type question, which students had 3 days to answer. Answers were limited to no more than four 1.5-spaced typed pages. The question asked students "to provide advice concerning [a particular party's] rights and liabilities" given presented a fairly complex factual scenario. The instructor designed the facts of the problem to raise issues involving many of the legal claims and concepts (e.g., trade secret law, shop rights to inventions, right of publicity, passing off) that were discussed in the first part of the course. Each claim involves different legal interests and requirements and presents a different framework for viewing the problem. Students were expected to analyze the facts, identify the claims and issues raised, make arguments pro and con resolution of the issue in terms of the concepts, rules, and cases discussed in class, and make recommendations accordingly. Since the instructor was careful to include factual weaknesses as well as strengths for each claim, the problem was ill-defined; strong arguments could be made for and against each party's claims.

With peer review systems such as CPR [5] and SWoRD [6], (1) students in a class write essays on a topic assigned by the instructor, and (2) the system distributes the essays to a group of *N* student peers for review. (3) Using review criteria and forms prepared by the instructor, the peer reviewers assess the student authors' papers along the criteria and submit their feedback via the system. It is important that reviewers provide written justifications of their numeric ratings. [7] The authors (4) may indicate whether or not the feedback was helpful, and (5) revise their drafts. In our study, the participants completed steps (1) through (4) of this peer review process. Students were randomly assigned to one of the two conditions in a manner balanced with respect to their LSAT scores. Each student gave feedback to and received feedback from four others, who had to be in the same condition. We collected ratings according to Likert scales (7 points, grounded at 1,3,5,7). The conditions differed in the rating prompts used by the reviewers, either domain-relevant or problem-specific. The former dealt with legal writing skills (i.e., issue identification, argument development, justifying an overall conclusion, and writing quality). The latter addressed criteria concerning five legal claims or issues raised by the problem's facts. This yielded two datasets: domain-relevant and problem-specific.

3 Overview of Bayesian Data Analysis

Our several different statistical models representing the domain of peer review use an expert's scores of the students' essays as the response variable; the models differ in the explanatory variables they use and in the hierarchical structure. We ask if it is possible to approximate the instructor scores by using the artifacts of peer review. We consider whether the additional complexity required for sophisticated modeling is a worthwhile trade-off for the inferences supported by the models.

We use a statistical modeling technique called Bayesian data analysis. [9] Bayesian models can incorporate prior beliefs about the parameters; for example, aggregate peer ratings may be said to be normally distributed. By combining prior beliefs with data and with formulations of likelihood, a Bayesian model yields posterior estimates for the parameters of interest and describes each estimate in terms of a probability distribution rather than just a point value.

While Bayesian modeling has long been applied in educational research, as far as known, our use of it is a novel contribution to the study of peer assessment in education. From the perspective of statistical analysis, peer review is fairly complex. It involves repeated measures (multiple reviews of every paper), sparse data (any student reviews only a few papers), and hierarchy (authors may receive feedback according to multiple reviewing criteria). By using Bayesian data analysis, we can enter these relationships among the data into our model in a straightforward way, and we can compare different models based on our intuitions about model structure. Furthermore, a single Bayesian computation estimates all the quantities of interest at once, bringing to bear all the available data. This means that the different parameters help estimate each other according to the expression of likelihood we enter.

Given two models that fit the data equally well, one may prefer the simpler one (e.g., complex models can be prone to overfitting) or the more complex one (e.g., it may embody knowledge about domain structure). We compare models in each condition in terms of Deviance Information Criterion (DIC), a metric that rewards well-fitting models, and penalizes models for complexity. Model fit is defined as deviance, similar to generalized linear models. Model complexity involves the effective number of parameters in the model. This is computed at model "run time" as a function of how information is pooled across groups in a multilevel model, rather than at "compile time" from the mathematical model structure. Lower DIC is better. DIC values may be compared on one dataset but not across datasets.

4 Hierarchical Bayesian Models of Peer Review

In each model, we regress the instructor score on peer ratings. Our baseline model 5.1a uses the simplest representation that maps from peer ratings to an instructor's score; it averages all ratings an author receives. It ignores the distinct rating dimensions reviewers used, and treats students as independent. In model 5.1b, we do not treat students as independent, pooling model parameters so that what we know about students as a group helps us understand individual students, and vice versa. In model 5.2b, we represent the ratings dimensions separately rather than together. We also developed models (not described here) that include inbound back-review ratings as a predictor and seek out trustworthy reviewers by comparing reviewer opinions.

We centered the peer ratings about the mean (and centered separately within each rating dimension for model 5.2b). We ran each model separately for the students in the two datasets, because it would not be sensible to compute the contribution of problem-specific information for students in the domain-relevant condition and vice versa. We fit each model 3 times and examined whether the chains converged in their posterior estimates. Each fit was allowed 6000 iterations, with 1000 initial iterations discarded to avoid bias due to randomly determined starting values.

4.1 Model 5.1a: Contribution of Inbound Peer Ratings

Model 5.1a is a regression of the midterm scores as a function of the pupils' inbound peer ratings only. We treat students as randomly drawn from a single population, and we do not distinguish between rating dimensions.

The multiple ratings that each student receives are exchangeable with each other (i.e., not tied to particular reviewers), and constitute repeated measures of each student. We treat midterm and ratings as normally distributed.

The inbound peer ratings are taken as normally distributed and sufficiently described by each author's ratings mean and variance. In such a model, the means and variances are hyperparameters and estimated simultaneously with other parameters during MCMC sampling. This yields both point estimates and posterior distributions with credible intervals indicating the model's certainty in the parameter estimate.

Formally, the model is as follows. The per-pupil instructor score Y_p is distributed normally, with a mean that is the per-pupil knowledge estimate μ_p and overall variance estimate σ^2.

$$Y_p \sim N(\mu_p, \sigma^2)$$

We fit a per-pupil intercept α_p. We also compute $\mu_p^{[IPR]}$, the mean of inbound peer ratings for pupil p ignoring criteria distinctions, and we give this a weight β.

$$\mu_p = \alpha_p + \beta * \mu_p^{[IPR]}$$

Finally, we say that a pupil's inbound peer ratings are distributed normally according to the pupil's individual mean $\mu_p^{[IPR]}$ and individual ratings variance $\sigma_{p[IPR]}^2$.

$$IPR_p \sim N(\mu_p^{[IPR]}, \sigma_{p[IPR]}^2)$$

The prior distribution for $\mu_p^{[IPR]}$ was said to be uninformative, normally distributed with a mean of 0 and a variance of 1000.

This "no pooling" regression model does not share information across pupils. [9] Each pupil is described via individual intercept α_p, between-pupils variance σ^2, individual mean peer rating $\mu_p^{[IPR]}$ and individual ratings variance $\sigma_{p[IPR]}^2$. In other models, below, we consider that information could be pooled across students, and what we learn about one student could help describe a different student.

Model 5.1a is a plausible first attempt to establish if the ratings that peers give each other approximate instructor assessment. It asks if the cumulative opinion of the reviewers (i.e., the mean inbound peer rating) corresponds to an instructor's grade, and if the peer reviewers tend to agree (i.e., measuring the ratings' variance). Additionally, it incorporates normal prior distributions for the response and the ratings. Whether or not this baseline differs from the alternative models, its evaluation should still be helpful in understanding peer review.

4.2 Model 5.1b: Contribution of Information Pooling

In model 5.1b, we use information learned about one student to inform our understanding of other students. This is accomplished in two ways. First, we stipulate that all individual intercepts α_p are not independent, but drawn from a common distribution. Each student's information is then used to estimate this distribution's hyperparameters μ_α and σ_α^2, and the distribution in turn constrains the estimation of the individual students' intercepts.

$$\alpha_p \sim N(\mu_\alpha, \sigma_\alpha^2)$$

Second, in a similar fashion, we constrain the estimation of individual students' inbound peer rating means $\mu_p^{[IPR]}$ via hyperparameters $\mu_{[IPR]}$ and $\sigma_{[IPR]}^2$.

$$\mu_p^{[IPR]} \sim N(\mu_{[IPR]}, \sigma_{[IPR]}^2)$$

All hyperparameters were given uninformative prior distributions.

4.3 Model 5.2b: Contribution of Rating Dimensions

Models 5.1a and 5.1b treat all inbound peer ratings as though they correspond to one rating dimension, no matter that they were elicited via different prompting questions. Model 5.2b represents the distinct dimensions of the peer ratings.

To incorporate information on dimensions, we say that each observed inbound peer rating is normally distributed with mean $\mu_{ip}^{[IPR]}$ that is equal to the average of the ratings received by author p for rating dimension i, and with a variance $\sigma_{i[IPR]}^2$ for dimension i that is shared across all pupils. (There are n=4 rating dimensions in the domain-relevant condition, and n=5 in the problem-specific condition.)

$$Y_p \sim N(\mu_p, \sigma^2)$$

$$\mu_p = \alpha + \Sigma_1^n \beta_i * \mu_{ip}^{[IPR]}$$

$$IPR_{ip} \sim N(\mu_{ip}^{[IPR]}, \sigma_{i[IPR]}^2)$$

Within each dimension, we pool individual pupils' means of inbound peer ratings $\mu_{ip}^{[IPR]}$ by stipulating a common distribution across students. These have uninformative prior distributions, normal with a mean of 0 and a variance of 1000.

Model 5.2b estimates a coefficient β_i for each rating dimension i. A partially pooled α_p in this model had problems with convergence, so we substituted a single completely pooled intercept α. Distinguishing the ratings by dimension leads to fewer observed ratings per pupil, per dimension. For example, rather than 20 peer ratings per pupil in the problem-specific condition (5 rating dimensions times 4 reviewers), there are ratings from 4 reviewers per dimension. This precludes estimation of individual per-dimension variances; instead, we estimate per-dimension variance parameters $\sigma_{i[IPR]}^2$ pooled across all students.

5 Results and Discussion

There are two key findings. First, partial pooling (model 5.1b) can improve significantly on the baseline (5.1a) for both domain-relevant and problem-specific datasets. (Table 1) Second, distinguishing the different rating criteria (5.2b) improves the fit for the problem-specific dataset, but actually hurts the fit for domain-relevant.[1]

Table 1. Model fit (DIC) for domain-relevant and problem-specific datasets

Model	domain-relevant DIC	problem-specific DIC
5.1a	1416	2423
5.1b	1305	2237
5.2b	1347	1732

In all three models, the intercepts α_p (5.1a, 5.1b) and α (5.2b) were estimated to be close to the mean of the instructor-assigned midterm scores. With ratings centered, the intercept represents the predicted midterm score for a student whose inbound peer ratings averaged to zero. Pooling made intercept estimates an order of magnitude tighter, e.g., for the problem-specific dataset, to ±0.8 points with 95% confidence on the instructor's scoring scale. Pooling for $\mu_p^{[IPR]}$ also allowed model 5.1b to share information across students, but the effect was less pronounced given that 5.1a already had tight intervals on these parameters. We find that partial pooling can be an effective technique for these models.

[1] We have seen DIC scores vary ±15 points over a dataset given different random starting points. Despite high autocorrelation for some parameters, Rhat values for all parameters in all models were below 1.2, i.e., the chains converged in their estimates. The chains mixed well, suggesting that samplers did not get stuck. Thus, we conclude that the results are stable.

Table 2. β coefficients, domain-relevant (DR) and problem-specific (PS), * marks significance

Model	β	β_1	β_2	β_3	β_4	β_5
5.1a-DR	-0.14					
5.1a-PS	-0.07					
5.1b-DR	0.99*					
5.1b-PS	1.46*					
5.2b-DR		1.48*	-0.50	0.56	-1.01	
5.2b-PS		2.10	0.86*	0.03	0.32*	0.40

The β coefficients could be said to represent the importance of the averaged inbound peer ratings (per-dimension or collapsing dimensions) to estimating the instructor's score. Under 5.1a, the credible intervals for β included zero, implying that peer ratings were not significant predictors of instructor scores for that model. With model 5.1b, estimates of β with 95% confidence did show that average peer ratings predicted instructor scores, emphasizing the value of pooling.

Model 5.2b showed that distinct rating dimensions helped to fit the data using problem-specific criteria but not using domain-relevant criteria. This can be seen from the overall fit (Table 1); additionally the β coefficients show that two of five the problem-specific dimensions helped to estimate the midterm score (a third, β_5, was marginally helpful), versus just one of four domain-relevant ones. Further, the problem-specific β estimates are all positive, suggesting that each dimension adds linearly to the intercept. Some domain-relevant β estimates have negative signs, as if high performance on those dimensions corresponds to a drop in the midterm score, which is counterintuitive. These problems for domain-relevant criteria echo the high pairwise correlation between $\mu_{ip}^{[IPR]}$ for all $\binom{4}{2} = 6$ criteria pairs; problem-specific support had correlation for only 2 of 10 pairs, as reported earlier. [8] High collinearity may cause instability and interactions among β_i coefficients for the domain-relevant rating criteria (without hurting overall model fit). The β_i for the problem-specific ratings may be intuitively interpreted as indicating that criteria differ in their impact on approximating instructor scores.

6 Conclusion

Parameter estimates from these models are likely to provide an ITS or an instructor with actionable information on individual pupils, the whole class, and the assessment rubric itself. Some may even suggest changes in curriculum or assessment. For example, $\mu_{ip}^{[IPR]}$ estimate a student's proficiency with regard to the criteria. Distributions of $\mu_{ip}^{[IPR]}$ can alert an instructor if the criteria differ in difficulty, or if they are poorly anchored. Pairwise correlations among each pupil's $\mu_{ip}^{[IPR]}$ may suggest which criteria are redundant. Inconsistent signs among β_i may hint that the reviewers' rubric differed from the instructor's grading scheme, while consistent β_i show how the criteria differ in their impact on approximating instructor scores. For instance, in the domain-relevant dataset, the $\mu_{ip}^{[IPR]}$ intercorrelation and the signs of β_i

suggest that the instructor should clarify the criteria and concepts to students and revise the criteria for future peer review. All these estimates accommodate missing peer ratings because they "borrow strength" from other students' ratings, and because they are posterior distributions with intervals that speak to the estimates' credibility.

In some cases, peer assessment is an important perspective on a student's work in its own right; in others, its relevance may depend on how well it approximates assessment by an instructor or other expert. Either way, consumers of peer assessment information, whether instructors or tutoring systems, require precise estimates of the key parameters in peer assessment. They also need to know whether or not the estimates are credible. The Bayesian models we have described fill that role.

The old software developers' adage "garbage in, garbage out" applies to peer assessment criteria. Criteria are not all equally useful, clear, or functional. The models we have developed and the results they report are only as good as the criteria. The good news is that peer review provides a built-in facility for evaluating the criteria, which can help instructors to refine them and to communicate them to pupils.

References

1. Strijbos, J., Sluijsmans, D.: Unravelling Peer Assessment. Special Issue of Learning and Instruction 20(4) (2010)
2. Goldin, I.M., Brusilovsky, P., Schunn, C., Ashley, K.D., Hsiao, I. (eds.): Workshop on Computer-Supported Peer Review in Education, 10th International Conference on Intelligent Tutoring Systems, Pittsburgh, PA (2010)
3. Falchikov, N., Goldfinch, J.: Student peer assessment in higher education: a meta-analysis comparing peer and teacher marks. Rev. of Ed. Research 70, 287–322 (2000)
4. Cho, K., Chung, T.R., King, W.R., Schunn, C.: Peer-based computer-supported knowledge refinement: an empirical investigation. Commun. ACM 51, 83–88 (2008)
5. Russell, A.: Calibrated Peer Review: A writing and critical thinking instructional tool. Invention and Impact: Building Excellence in Undergraduate Science, Technology, Engineering and Mathematics (STEM) Education. American Association for the Advancement of Science (2004)
6. Cho, K., Schunn, C.D.: Scaffolded writing and rewriting in the discipline: A web-based reciprocal peer review system. Computers and Education 48 (2007)
7. Wooley, R., Was, C.A., Schunn, C.D., Dalton, D.W.: The effects of feedback elaboration on the giver of feedback, pp. 2375–2380. Cognitive Science Society, Washington, DC (2008)
8. Goldin, I.M., Ashley, K.D.: Eliciting informative feedback in peer review: Importance of problem-specific scaffolding. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 95–104. Springer, Heidelberg (2010)
9. Gelman, A., Hill, J.: Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press, Cambridge (2006)

Modeling Confusion: Facial Expression, Task, and Discourse in Task-Oriented Tutorial Dialogue

Joseph F. Grafsgaard[1], Kristy Elizabeth Boyer[1],
Robert Phillips[1,2], and James C. Lester[1]

[1] Department of Computer Science, North Carolina State University
Raleigh, North Carolina, USA
[2] Applied Research Associates, Inc.
Raleigh, North Carolina, USA
{jfgrafsg,keboyer,rphilli,lester}@ncsu.edu

Abstract. Recent years have seen a growing recognition of the importance of affect in learning. Efforts are being undertaken to enable intelligent tutoring systems to recognize and respond to learner emotion, but the field has not yet seen the emergence of a fully contextualized model of learner affect. This paper reports on a study of learner affect through an analysis of facial expression in human task-oriented tutorial dialogue. It extends prior work through in-depth analyses of a highly informative facial action unit and its interdependencies with dialogue utterances and task structure. The results demonstrate some ways in which learner facial expressions are dependent on both dialogue and task context. The findings also hold design implications for affect recognition and tutorial strategy selection within tutorial dialogue systems.

Keywords: Affect, tutorial dialogue, tutorial strategies.

1 Introduction

Recent years have seen a growing recognition of the role that affective computing can play in providing students with highly adaptive and effective learning experiences [1,2]. These investigations highlight the importance of affect in tutorial interactions and have contributed to an emerging understanding of learner emotions [2-7]. To date, a number of systems have incorporated affect, recognizing and responding to it in pedagogically beneficial ways [8-10]. However, the field has not yet seen the emergence of a contextualized model of affect that explains when learners are likely to experience particular emotions and what the impacts of affective states are on learning outcomes.

This paper presents a novel approach to analyzing student emotion, as evidenced by facial expressions, during computer-mediated human task-oriented tutorial dialogues. In particular, we focus on all occurrences of a specific facial *action unit* [11] that has been shown to correlate with confusion in learning [12,13], as well as with anger, fear, and mental effort in other settings [14,15]. Concentrating on this single, highly relevant facial action unit reveals important interdependencies between facial expression, dialogue, and task structure. We discuss ways in which tutorial

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 98–105, 2011.

dialogue systems can leverage these contextual models of student affect to inform such behaviors as question asking and adaptive delivery of feedback.

2 Related Work

Research on emotion during learning within the AI in Education community has focused on predictive models of student affect [9,10,13,16,17], affective adaptations within intelligent tutoring systems [1,6,18], and understanding student affect during tutoring sessions [2-5,7]. Prior studies on understanding student affect during learning have aimed to identify the presence and characteristics of student emotions and transitions between them. Confusion and flow have been observed to positively effect learning gains, while boredom has a negative impact [3]. A state of stuck may be an important negative parallel to the state of flow [18]. Learners may transition in particular ways among the emotions of boredom, confusion, curiosity, delight, eureka, flow, and frustration, as shown in several studies [2-4,6].

Facial expressions provide a natural window onto student affect. Automated tracking of facial features and head movement has been shown to predict self-reported frustration [10], as well as confidence, interest, and excitement [9,19]. Studies of facial expression in learning contexts found that learner emotions are discernible through facial features [5,20] and that facial and discourse features diagnose confusion more accurately than gross body language [21]. Particular facial configurations have been found to correlate with learner emotions, and facial *action unit 4* (AU4), the Brow Lowerer, has been most strongly correlated with confusion [12,13].

The current work focuses on the affective state of confusion as evidenced by AU4 and extends previous work by applying a focused manual facial annotation approach to tutoring sessions in their entirety. This paper contributes to the body of empirical results on facial expressions of emotion by examining how the context of dialogue and learning task are associated with student displays of a highly relevant facial action unit, AU4.

3 Corpus and Facial Action Analysis

A corpus of human-human tutorial dialogue was collected during a tutorial dialogue study. Students solved an introductory computer programming problem and carried on computer-mediated textual dialogue with a human tutor. The original corpus consists of 48 dialogues and was previously annotated with dialogue acts and subtask structure [22]. Facial recordings of students were collected using built-in webcams. The tutors were not shown the student facial videos. Video quality was ranked based on how completely each student's face was visible within the frame, and the fourteen highest quality videos were used in this analysis. They have a total running time of eleven hours 55 minutes and include dialogues with three female subjects and eleven male subjects.

The facial videos were annotated manually using the Facial Action Coding System (FACS), which enumerates the possible movements of the face through a set of facial action units [11]. The FACS coders viewed videos from start to finish, pausing at observed instances of AU4 activation (Figure 2). Facial movements were encoded as events with a start frame and an end frame. A certified FACS coder [14] annotated all fourteen videos. A second certified FACS coder annotated six videos. After the tagging was complete, the sessions were discretized into one-second intervals. Cohen's kappa for inter-coder agreement on AU4 across all one-second intervals was κ=0.86, which indicates very good reliability. Excerpts from the fully annotated corpus are shown in Figure 1. Displays of AU4 were noted during a total of 53 minutes of the approximately 12 hours of video, with high variance across individual students (*min*=0 seconds; *max*=33 minutes).

Excerpt 1		
14:07:03	Tutor:	ok, so that's closer [Lukewarm Fdbk]
14:07:23	Tutor:	but you are currently saying, i want the value at position i to be the same as the value at position i + 1 [Statement]
	Student:	BuggyTaskAction
14:07:43	Tutor:	instead of wanting the value at position i to be one more than the current value at position i [Statement]
Excerpt 2		
17:44:41	Tutor:	okay, good so far [PositiveFdbk]
17:44:47	Tutor:	except there's a typo in that loop condition [NegContentFdbk]
	Student:	CorrectTaskAction
17:45:08	Tutor:	now that we have n, how can we change the loop condition for c? [AssessingQuestion]
	Student:	FacialExpression: AU4
Excerpt 3		
15:43:26	Tutor:	well you have one error, it's underlined in red [NegativeContentFdbk]
	Student:	FacialExpression: AU4, CorrectTaskAction
15:43:35	Tutor:	yup [PositiveFdbk]
	Student:	FacialExpression: AU4, IncompleteTaskAction
15:44:01	Tutor:	so far so good, let's fix the return statement and then we should probably check if the first two problems work by running it [Lukewarm Fdbk]

Fig. 1. Tutoring session excerpts

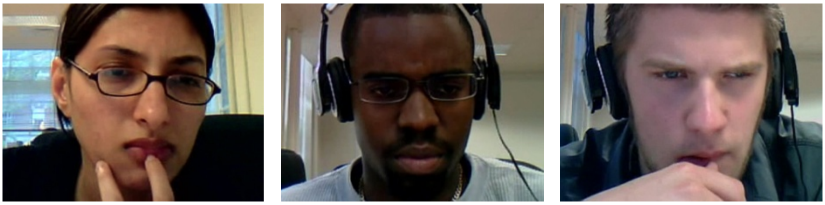


Fig. 2. Student displays of facial action unit 4 (AU4, Brow Lowerer)

The annotated facial action data were merged with the previously annotated dialogue acts and task actions to form a chronological record of task actions, dialogue, and student displays of AU4 that were then used to empirically explore dependencies between events. Table 1 displays the relative frequencies for student task action tags that occurred at the same time as AU4. Statistically significant differences are in bold.[1] Students were significantly less likely to display AU4 while engaging in on-track, INCOMPLETE task actions. Students were also more likely to display AU4 during a BUGGY or CORRECT task action, and less likely during DISPREFERRED task actions (which technically meet the problem specifications but circumvent the pedagogical goals of the task), though these differences were not statistically significant.

Table 2 displays the analogous relative frequencies of tutor dialogue acts across all sessions compared with the relative frequencies of only those dialogue acts that were followed by a student display of AU4 within ten seconds. The results indicate that students were significantly less likely to display AU4 immediately following tutor EXTRA-DOMAIN moves, LUKEWARM FEEDBACK, and QUESTIONS.

Table 1. Student AU4 during task actions[2]

Student Task Action	**Relative Freq. of Task Action *(stdev)*[3]**	**Rel. Freq. of Task Action With Student AU4 Present *(stdev)***	***p*-value (paired *t*-test, *N*=13)**
BUGGY	0.578 *(0.156)*	0.602 *(0.333)*	0.7808
DISPREFERRED	0.057 *(0.106)*	0 *(0.001)*	0.0773
INCOMPLETE (ON-TRACK)	**0.154 *(0.143)***	**0.082 *(0.151)***	**0.0076**
CORRECT	0.809 *(0.183)*	0.856 *(0.176)*	0.2943

Table 2. Student AU4 following tutor dialogue acts

Tutor Dialogue Act	**Relative Freq. of Tutor Act (*stdev*)**	**Rel. Freq. Of Tutor Act With Student AU4 w/in 10 Sec. (*stdev*)**	***p*-value (paired *t*-test, *N*=11)**
ASSESSING QUESTION	0.097 *(0.075)*	0.177 *(0.233)*	0.2510
EXTRA DOMAIN	**0.055 *(0.057)***	**0.009 *(0.020)***	**0.0227**
GROUNDING	0.063 *(0.081)*	0.020 *(0.052)*	0.2007
LUKEWARM CONTENT FDBK	0.031 *(0.025)*	0.012 *(0.028)*	0.0680
LUKEWARM FDBK	**0.023 *(0.021)***	**0 *(0)***	**0.0047**
NEGATIVE CONTENT FDBK	0.094 *(0.053)*	0.153 *(0.191)*	0.3117
NEGATIVE FDBK	0.016 *(0.013)*	0.006 *(0.014)*	0.0819
POSITIVE CONTENT FDBK	0.032 *(0.030)*	0.051 *(0.107)*	0.55
POSITIVE FDBK	0.150 *(0.069)*	0.162 *(0.317)*	0.9040
QUESTION	**0.049 *(0.060)***	**0.004 *(0.012)***	**0.0363**
STATEMENT	0.391 *(0.119)*	0.406 *(0.254)*	0.8221

[1] Because of the limited sample size and the goal of highlighting trends that warrant future study, a statistical correction for multiple tests was not applied.

[2] Sample sizes *N* reflect only students who displayed AU4 during task action segments (Table 1) or within ten seconds of any dialogue act (Table 2). Else the corresponding probability could not be calculated.

[3] Task action segments may contain multiple tags and therefore do not sum to one.

4 Discussion

These results indicate that student expressions of AU4 are dependent on both the dialogue and task context. This action unit is highly relevant for tutoring because of prior findings that it is correlated with confusion, negative emotions, and mental effort [12-15]. A contextual understanding of this action unit during learning may hold a number of important insights for developing affective tutoring systems.

4.1 Interpretation

After tutor EXTRA-DOMAIN dialogue acts, students were significantly less likely to display AU4, which is consistent with an understanding of EXTRA-DOMAIN moves as conversational and unrelated to the learning task. Students were also less likely to display AU4 following tutor LUKEWARM FEEDBACK, a finding that may at first seem counterintuitive. However, as demonstrated by Excerpt 1 of Figure 1, these tutors often used LUKEWARM FEEDBACK to encourage students. Finally, students were less likely to display AU4 immediately following a tutor QUESTION. This finding may also seem counterintuitive given the expectation that question answering may induce confusion, or at least require mental effort, on the part of the student. However, the lack of this facial expression following tutor questions is consistent with a prior observation that the non-expert tutors in this corpus rarely posed deep reasoning questions, but instead tended to ask shallow questions that could be answered quickly [23]. We hypothesize that when working with expert tutors, the statistical relationship between tutor questions and student expressions of AU4 may be reversed.

Some other trends warrant discussion although they did not display statistically significant relationships. For example, tutor ASSESSING QUESTION dialogue moves were more likely to be followed by student AU4 (Figure 1, Excerpt 2). Such questions ask students to reflect on what they already know. For novice students, being asked directly about their knowledge may have produced genuine confusion as they worked to reconcile their emerging knowledge of specific target concepts with their pre-existing knowledge. A similar phenomenon may explain why students were more likely to display AU4 after NEGATIVE CONTENT FEEDBACK (Figure 1, Excerpt 3). Out of all types of feedback, this type may be most likely to place students into cognitive disequilibrium [7].

A statistically significant dependence also emerged between student INCOMPLETE, on-track task actions and AU4. Students were less likely to display AU4 while engaged in these task actions. This finding is likely related to the cognitive-affective state of flow, in which the student is actively focused and making progress on the learning task [24].

4.2 Design Implications

These findings have important implications for the design of intelligent tutoring systems in two dimensions: affect recognition and tutorial strategy refinement. First, affect recognition involves inferring the student's emotional state based on a variety of predictors. *A priori* knowledge that a particular emotional state is more or less likely given the context of the dialogue or task may narrow the state space under consideration by an affect recognition model, potentially increasing efficiency and accuracy. Second, understanding which student emotions are likely to follow particular tutor moves or problem-solving events can help an ITS select cognitive strategies or affective interventions that are likely to guide students toward affective states conducive to learning.

The results presented here suggest particular ways in which ITSs may leverage knowledge of student affect to provide highly adaptive, affect-informed feedback. For example, the type of question the system poses may directly impact whether the student displays confusion-related facial expressions. Shallow questions are unlikely to produce a cognitive-affective state of confusion, while deep reasoning and assessment questions are more likely to do so. Additionally, when providing feedback on student errors, indirect approaches such as LUKEWARM FEEDBACK may not be sufficient to help novice students become aware of their mistakes or misconceptions. NEGATIVE CONTENT FEEDBACK, in which student errors are explicitly pointed out and a hint is given, appears more likely to accomplish this. Finally, the low probability of observing AU4 during student INCOMPLETE, on-track work emphasizes the importance of sensitivity during possible times of student flow, when a system may choose not to interrupt.

4.3 Limitations

The study has two primary limitations. First, the number of tutoring sessions is small due to the time-intensive manual tagging approach, which for each coder required up to ten hours per hour of video.[4] While manual annotation is time-intensive, it nevertheless serves as a valuable part of achieving complete coverage of tutoring sessions and establishing a foundation on which highly reliable automated techniques can be built. A second limitation lies in the structure of the tutorial dialogue itself, namely, that student utterances are approximately half as numerous as tutor utterances. With a larger number of student utterances, a correlational analysis analogous to that reported in Table 2 could reveal patterns of dependence between student utterances and AU4.

5 Conclusion

Affect plays a central role in learning, and developing a clear understanding of learner emotions can lead to improved affect recognition and adaptation by intelligent tutoring systems. In particular, understanding the interdependencies between facial expression, dialogue, and task structure may hold important insights for designing affective tutoring systems. The work reported here has examined student facial expression, in particular AU4 (Brow Lowerer), during computer-mediated human task-oriented tutorial dialogue. The findings demonstrate that the occurrence of this confusion-related facial expression is dependent on both dialogue and task context. The results indicate that students are less likely to display AU4 immediately following tutor questions, lukewarm feedback, and extra-domain dialogue acts, as well as during incomplete, on-track task actions. Leveraging knowledge of these

[4] This annotation approach considers only a subset of FACS action units. It is significantly faster than full FACS coding, which requires up to sixty hours per hour of video.

patterns can help tutoring systems better recognize student affect and select strategies or interventions that encourage desirable affective states.

This work constitutes a first step toward a comprehensive catalogue of fine-grained facial configurations during learning and their relationships with the tutoring context. Employing a fine-grained approach that focuses on a single facial action unit highlights several important directions for future work. First, facial action coding is a domain-independent approach that can be used to compare the occurrence of student emotions across tutoring corpora. Second, promising work on automatic facial action tagging indicates that in the near future, this type of fine-grained investigation will no longer require manual annotation [25]. Finally, the Core Affect framework [26] provides a promising model by which comprehensive facial annotations and contextual features may be utilized to identify emotions without prior semantic assumptions. Together, these lines of investigation will contribute to the design of the next generation of affectively aware tutorial dialogue systems.

Acknowledgements

This work is supported in part by the NC State University Department of Computer Science along with the National Science Foundation through Grants IIS-0812291, DRL-1007962 and the STARS Alliance Grant CNS-0739216. Any opinions, findings, conclusions, or recommendations expressed in this report are those of the participants, and do not necessarily represent the official views, opinions, or policy of the National Science Foundation.

References

1. Woolf, B.P., Burleson, W., Arroyo, I., Dragon, T., Cooper, D.G., Picard, R.W.: Affect-Aware Tutors: Recognizing and Responding to Student Affect. International Journal of Learning Technology 4, 129–164 (2009)
2. D'Mello, S.K., Lehman, B., Person, N.: Monitoring Affect States During Effortful Problem Solving Activities. Int. J. Artif. Intell. Educ. 20 (2010)
3. Baker, R.S.J.d., D'Mello, S.K., Rodrigo, M.M.T., Graesser, A.C.: Better to Be Frustrated than Bored: The Incidence, Persistence, and Impact of Learners' Cognitive-Affective States during Interactions with Three Different Computer-Based Learning Environments. International Journal of Human-Computer Studies 68, 223–241 (2010)
4. Lehman, B., D'Mello, S., Person, N.: The intricate dance between cognition and emotion during expert tutoring. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 433–442. Springer, Heidelberg (2010)
5. Afzal, S., Robinson, P.: Natural Affect Data - Collection and Annotation in a Learning Context. In: Proceedings of the International Conference on Affective Computing and Intelligent Interaction, pp. 1–7 (2009)
6. Robison, J.L., McQuiggan, S.W., Lester, J.C.: Evaluating the Consequences of Affective Feedback in Intelligent Tutoring Systems. In: Proceedings of the International Conference on Affective Computing and Intelligent Interaction, pp. 37–42 (2009)
7. Graesser, A.C., Olde, B.A.: How Does One Know Whether a Person Understands a Device? The Quality of the Questions the Person Asks When the Device Breaks Down. Journal of Educational Psychology 95, 524–536 (2003)

8. D'Mello, S.K., Picard, R.W., Graesser, A.C.: Toward an Affect-Sensitive AutoTutor. IEEE Intelligent Systems 22, 53–61 (2007)
9. Cooper, D.G., Muldner, K., Arroyo, I., Woolf, B.P., Burleson, W.: Ranking Feature Sets for Emotion Models used in Classroom Based Intelligent Tutoring Systems. User Modeling, Adaptation, and Personalization, 135–146 (2010)
10. Kapoor, A., Burleson, W., Picard, R.W.: Automatic Prediction of Frustration. International Journal of Human-Computer Studies 65, 724–736 (2007)
11. Ekman, P., Friesen, W.V., Hager, J.C.: Facial Action Coding System. A Human Face, Salt Lake City, USA (2002)
12. Craig, S.D., D'Mello, S.K., Witherspoon, A., Graesser, A.: Emote Aloud During Learning with AutoTutor: Applying the Facial Action Coding System to Cognitive-Affective States During Learning. Cognition & Emotion 22, 777–788 (2008)
13. McDaniel, B.T., D'Mello, S.K., King, B.G., Chipman, P., Tapp, K., Graesser, A.C.: Facial Features for Affective State Detection in Learning Environments. In: Proceedings of the 29th Annual Meeting of the Cognitive Science Society, pp. 467–472 (2007)
14. Ekman, P., Friesen, W.V., Hager, J.C.: Facial Action Coding System: Investigator's Guide. A Human Face, Salt Lake City, USA (2002)
15. Cohn, J.F., Zlochower, A.J., Lien, J., Kanade, T.: Automated Face Analysis by Feature Point Tracking Has High Concurrent Validity with Manual FACS Coding. Psychophysiology 36, 35–43 (1999)
16. Conati, C., Maclaren, H.: Empirically Building and Evaluating a Probabilistic Model of User Affect. User Modeling and User-Adapted Interaction 19, 267–303 (2009)
17. McQuiggan, S.W., Lee, S., Lester, J.C.: Early Prediction of Student Frustration. In: Proceedings of the Second International Conference on Affective Computing and Intelligent Interactions, pp. 698–709 (2007)
18. Burleson, W.: Affective Learning Companions: Strategies for Empathetic Agents with Real-Time Multimodal Affective Sensing to Foster Meta-Cognitive and Meta-Affective Approaches to Learning, Motivation, and Perseverance. MIT Ph.D. thesis (2006)
19. Kaliouby, R., Robinson, P.: The Emotional Hearing Aid: An Assistive Tool for Children with Asperger Syndrome. Universal Access in the Information Society 4, 121–134 (2005)
20. Afzal, S., Robinson, P.: Modelling Affect in Learning Environments - Motivation and Methods. In: Proceedings of the International Conference on Advanced Learning Technologies (2010)
21. D'Mello, S.K., Graesser, A.C.: Multimodal Semi-Automated Affect Detection from Conversational Cues, Gross Body Language, and Facial Features. User Modeling and User-Adapted Interaction 20, 147–187 (2010)
22. Boyer, K.E., Phillips, R., Ingram, A., Ha, E.Y., Wallis, M. D., Vouk, M. A., Lester, J. C.: Characterizing the effectiveness of tutorial dialogue with hidden markov models. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 55–64. Springer, Heidelberg (2010)
23. Boyer, K.E., Lahti, W.J., Phillips, R., Wallis, M.D., Vouk, M.A., Lester, J.C.: An Empirically-Derived Question Taxonomy for Task-Oriented Tutorial Dialogue. In: Proceedings of the Second Workshop on Question Generation, pp. 9–16 (2009)
24. Csikszentmihalyi, M.: Flow: The Psychology of Optimal Experience. Harper-Row, NY (1990)
25. Calvo, R.A., D'Mello, S.K.: Affect Detection: An Interdisciplinary Review of Models, Methods, and Their Applications. IEEE Transactions on Affective Computing 1, 18–37 (2010)
26. Russell, J.A.: Core Affect and the Psychological Construction of Emotion. Psychological Review 110, 145–172 (2003)

Extending a Teachable Agent with a Social Conversation Module – Effects on Student Experiences and Learning

Agneta Gulz[1], Magnus Haake[2], and Annika Silvervarg[1]

[1] Department of Computer Science, Linköping University, Sweden
[2] Department of Design Sciences, Lund University, Sweden

Abstract. The paper discusses the addition of off-task socially oriented conversational abilities to an existing “teachable agent” (TA) in an educational game in mathematics. The purpose of this extension is to affect constructs known to promote learning, such as self-efficacy and engagement as well as enhance students’ experiences of interacting with the game. A comparison of students that played the game with the off-task interaction to those who played without it, shows trends that indicate that students who played the game with off-task interaction had a more positive experience of the game, and that they also learnt more, as reflected in the learning outcomes of their TAs.

Keywords: Educational game in mathematics, conversational pedagogical agent, teachable agent, off-task interaction, socially oriented conversation.

1 Introduction

By conversational pedagogical agents, CPA:s, we refer to computer generated characters in a pedagogical context that engage in spoken or written conversation with students. Some CPAs may use non-verbal conversational channels, such as gestures and facial expressions, but this paper limits itself to conversation in written language. Various research groups have developed CPAs for different domains like physics, mathematics, foreign languages, programming and many others, and several evaluative studies have shown that CPAs can be effective as tutors [e.g. 1, 2]. There is a large variety in how CPA:s are designed and what specific pedagogical strategies they exploit, but all engage, in some way or other, in *task-oriented conversation* such as: elaborating on students’ answers, asking questions regarding the domain or task, correcting misconceptions, asking students to elaborate on their examples, providing hints and directions. In other words, they engage in conversation that clearly pertains to the learning material and tasks in question.

During the past decade some researchers have developed CPAs that in addition to carrying out task-oriented conversation engage in *relation oriented or socially oriented conversation* with students, i.e. conversation with no (apparent) relation to the learning tasks. Examples are: reassuring or cheering up a student, carrying on small-talk, engaging in mutual self-disclosure. Relational or social behaviours can also be realized via non-verbal communication, but this is outside the scope of this paper.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 106–114, 2011.

Reasons for adding a capability for socially oriented conversation in a CPA include: i) increased overall engagement and receptivity [3] ii) improved recall of the learning material through emotional engagement [4] in particular because social experiences activate the reward circuitry of the brain, which helps cement newly learned associations [5], iii) promotion of trust and rapport-building [6] and finally that students may feel more at ease with a learning task or topic [7]. For a more extensive presentation of these and other reasons, see [8, 9]. This paper describes a teachable agent that has been extended with a social conversation module. We discuss the justification for the extension and present an empirical study that evaluates the effects of the extension in terms of student experiences and learning. However, first we present selected examples of other CPAs capable of socially oriented conversation.

2 Previous Work

CPAs with a capability for socially oriented conversation broadly belong to two different categories. CPAs in the first category exhibit *on-task sociability*, that is they will and cannot digress into other topics than those that pertain to the learning task and domain. However, in connection with task-oriented conversation, they exhibit social behaviour such as displaying encouragement, assurance, agreement, and praise. One example is the *cooperative co-learner* [10] that in addition to on-task conversation in the domain of English language idioms, compliments and shows concern and encouragement when the difficulty level of the questions increase or when the student fails on a question (e.g. ”You’ll get the next one”). As another example consider the *Low social* and *High Social* agents [11] in a system for supporting collaborative design learning regarding thermodynamics. Student pairs can chat with each other as well as with the tutor CPA, where the percentage of social turns by the CPA (showing solidarity with a student who has difficulties, agreeing or showing tension release) is varied from 0% for the *No Social* agent, to 15% for the *Low social* and to 30% for the *High social.* As a third example [12] developed a model of socially intelligent tutorial dialogue on the basis of politeness theory. The *polite tutor agent* provided tutorial feedback to promote learner face and mitigate face threat, whereas the standard tutor agent provided direct feedback that disregarded learner face.

The second category of socially oriented CPAs contains those that exhibit *off-task sociability.* These are able to go outside of the task(s) and domain(s) and engage in conversation that involves small-talk-like topics, self-disclosure, personal narratives, etc. Although the work by T. Bickmore does not deal with pedagogical applications per se, it is central in this context. In [6] he coined the term *relational agent*, an agent designed to develop and maintain long-term, socio-emotional relations with users, and he has conducted a large number of studies that compare relation-oriented and strictly task-oriented agents and explore various off-domain sociability features. The value of autobiographical stories in agents is investigated in [13], with reference to Jakobson’s [14] *phatic* function of dialogue: to keep the communication channel open so that primary functional messages can be conveyed. The authors [13] propose that autobiographical storytelling by an agent is a central means for maintaining user engagement in an intervention over time – which can be crucial for educational applications. Yet they also point at the importance that the stories that an agent tells are *truly* engaging.

Kumar et al. [2] compared two software versions for letting student pairs engage in collaborative mathematics learning via a chat. Both versions contained *cognitive support agents*, but one also contained *social dialogue agents*, designed to show personal interest in the students by asking them to reveal their personal preferences about extra-curricular domains. The preferences were used as input when the math problems were constructed, with the intention that the social dialogue should give students the impression that the agent takes personal interest in them. The addition of the social dialogue agents turned out to have a strong positive effect on the attitude that students displayed towards agents and a slight positive effect on learning outcomes.

Mehlman et al. [15] present work on a learning game for expressing conceptual knowledge through qualitative reasoning models. A set of CPAs are included, and among them a *quizmaster agent*, that besides asking questions and giving feedback makes small talk utterances and humorous distractions unrelated to the quiz domain. This is modelled on how quizmasters in famous television shows countervail participants' stress and provide a more enjoyable form of competition.

3 A TA Based Game Extended with Social Off-task Conversation

Our game [16, 17] is a mathematics game that trains basic arithmetic skills with a focus on grounding base-ten concepts in spatial representations. It employs a board-game design with a variety of sub-games. When a student has learnt to play one particular board game, she can teach it to her *Teachable Agent* (TA) [18]. In the *observation mode* the TA "watches" the student play and picks up on game rules and on the student's responses to multiple-choice questions, such as "Why did you choose this card?" The student then chooses one answer from the listed potential explanations (but only one correct answer), including a "don't know" option. Proper (or improper) choices of cards and answers promote corresponding skills in the TA throughout the game. In the *try-and-be-guided mode*, the agent is allowed to propose cards. The student either accepts the agent's suggestion or rejects it and exchanges the agent's card for another one. In the latter case the agent asks, via the multiple-choice-format, why the student thinks her card was a better choice. For more information on the AI in the system we refer to [16, 17], which also describe the underlying pedagogical model of a master and an apprentice, which differs from the more common teacher-student model in TA-systems.

In other words, the basic TA-system contains a simple form of on-task conversation, via a multiple-choice format. A simple form of on-task sociability is involved as well, for instance the TA may praise the student when she earns points in the game.[1] For the study presented here, the game architecture was extended with a module where the student can engage in conversation with the TA, writing freely by means of the keyboard (in contrast to the multiple-choice format in the on-task conversation) and bring up basically any topic in a chat-like manner. We refer to this chat-like conversation as *off-task* conversation and distinguish within it between *on-domain* conversation and *off-domain* conversation – the former referring to chat conversation related to school, math and the math game (but notably not in the sense that the TA

[1] Yet a TA, which is merely a student of the learner, cannot coach with respect to whether answers to questions are wrong or right.

provides the student with information to play the games better or understand the math content better), and the latter to any other topics. The off-task conversation is implemented as a mixed-initiative dialogue strategy, which allows both the agent and the user to direct the dialogue by introducing new topics and posing questions. The agent keeps a history of the topics in the dialogue, both the current and previous sessions. *On-task* and *off-task* conversation have very different formats, but are still designed as two interrelated and complementary activities. The interconnecting factor is the persona of the agent, which integrates task and domain knowledge with off-domain knowledge (e.g. the agent is a 11-year old that goes to school and is learning math in the game, but also has interests such as music and film).

3.1 Aims and Relations to Other Systems

The off-task conversation is in the first place a means to enrich the game and its motivational qualities for a novel age group of 12-14 year old users. Informal pre-studies revealed that these users required more variation than younger students who became very engaged by the game in its basic form [19]. Bickmore's work and arguments on how social conversation with agents may be a means to maintain engagement in an intervention over time, was a main source of inspiration. Our aim is accordingly to enhance students' experience and increase their inclination to want to continue to use the game over time. A further aim is to exploit the off-task conversation for pedagogical interventions such as influencing students math self-efficacy and attitudes toward math. It is worth to point out that our work, like the work by others related above, approach off-task conversation in terms of its *pedagogical power* – not in terms of being pedagogically detrimental in taking attention from the learning task [e.g. 20]. We return to this in the discussion.

Enhancement of students' experience of the game can be achieved in various ways. For some individuals it is a question of variability in order to countervail boredom. For some it is a question of making the learning domain of mathematics more appealing and making students less tense or nervous (cf. Mehlman et al. [15] above). This in turn relates to the potential for more dedicated pedagogical interventions, cf. Kim et al. [7] on affecting students math self-efficacy and detracting "math anxious" students from perceived inabilities to confront mathematical learning material. For such interventions to work, trust in the agent is crucial. Bickmore [6] has shown that small talk and conversational storytelling can contribute to build such trust.

For our system we have taken inspiration from all of the above mentioned research. Nevertheless, our system is unique in involving a *Teachable agent* capable of off-task, off-domain, sociability. Compared to other pedagogical agents, a TA offers advantages as well as challenges when it comes to developing the agent's off-domain sociability. A TA sits at the very core of an educational software by instantiating the software pedagogy, i.e. learning by teaching. A TA is "someone" who has to learn from the student, and this means that there is an immediate, even if rudimentary, social relation between the student and her TA. Studies have shown that such a social relation develops between students and TAs also on the basis of strict on-task interaction and conversation alone [19, 21]. In other words there is a pedagogically integral and unquestionable sociability of a basic kind to start from and no risk that a pedagogical (teachable) agent is but a misguided social software garniture. While this is an advantage, there is a corresponding challenge in how to develop an adequate off-task conversation that extends and refines the rudimentary sociability of the (same) TA. In an on-task-conversation the relation between student and TA is quite straightforward, with the TA the one who learns and the student the one who teaches. But in an off-task-conversation one may for various reasons strive for a more equivalent peer-relation and more mutual learning. For details on how we approach this challenge of designing a peer, while yet retaining some of the protégée-effect [21], see [8].

The present study focused on the potential of the chat to increase engagement by comparing groups of students who used the original game with those who used the extended version, i.e. with and without chat. Apart from students' experiences of the game, we studied their perception of the TA's role, their self-efficacy (i.e. beliefs about their competency in playing the math game), and their learning accomplishments. We also studied possible differences between low- and high-achievers.

4 Method

4.1 Participants and Procedure

38 female and 42 male 12-14 year olds from three classes in a Swedish school participated in the study. The students were assigned a value (low, middle or high) for math achievement by their teacher, where 18 were classified as low, 39 as middle and 23 as high. Each class was divided into two groups with an even distribution according to gender and math achievement. All students got to play the math game during three lessons. The NoC group used the game without the chat module. The WithC group used the game with the off-task module, and after every two game sessions a "break" was offered. During the first three breaks the students had to chat with the agent until the break ended after three minutes, and the chat was closed. For the breaks thereafter the students were offered a choice between chatting with the agent or continuing to play, and when chatting there was always a choice to end the chat before the break was over. The students in the NoC group groups spent on average a total of 105 minutes with the game and the students in the WithC group 120 minutes, in order to make the time spent on the math game sessions equal for both groups. After the third lesson each student filled out a questionnaire.

It should be pointed out that all students, regardless of condition, did get breaks in the sense of cognitive rest and change of activity. Training one's agent involves an intellectual effort and working on math content, whereas letting one's TA play against the computer only requires passive viewing. For an observer it was obvious that students did made use of the latter as a kind of "break".

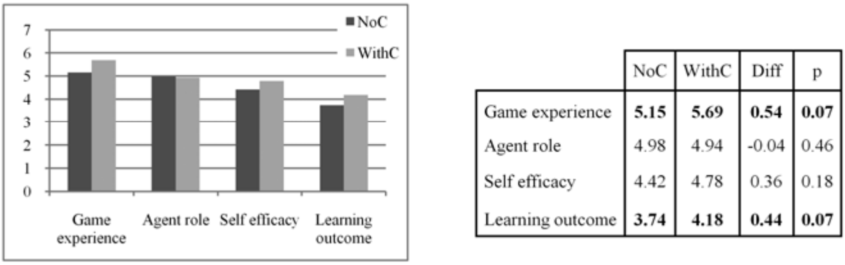
4.2 Instruments and Measurements

To evaluate the effect of the social chat on learning and experience of the game, a combination of data from questionnaires and computer-generated logs were used. The students filled out a questionnaire with 18 statements scaled from 1 (Strongly disagree) to 7 (Strongly agree). The questionnaire included the areas: i) game experience, e.g. if interesting, challenging, easy to concentrate, ii) experience of the role of the TA in facilitating learning and increasing enjoyment in the game, iii) self-efficacy beliefs regarding the game play and one's role as teacher. Statements for i) and ii) were developed based on [22], the self-efficacy measurement according to guidelines from [23]. Log data was used to see how well each student had taught her agent, as a measure of the student's own learning. For the WithC group the logs were also used to gather data regarding students' inclination to chat when given the choice.

5 Results and Analysis

Since not all students could participate in all three lessons or fill in the questionnaire, the final analysis included 29 females and 32 males. A comparison of the results on the questionnaire and the knowledge level of the trained agent for the NoC- and WithC-groups is presented in Figure 1. Items were clustered and an average score calculated for the game experience, the perceived importance of the agent's role in the system, and self-efficacy beliefs. The students' learning outcome was calculated based on the agent's final knowledge level in relation to how many times the student had played and trained the agent.

Figure 1 shows that students in the WithC-group tended to have a more positive game experience (diff=0.54, p=0.07), but there was no difference in the perceived role of the TA in the game, and marginal differences in self-efficacy beliefs. Also students in the WithC-group tended to reach better result in terms of how well they taught their TA (diff=0,3, p=0.07).



	NoC	WithC	Diff	p
Game experience	**5.15**	**5.69**	**0.54**	**0.07**
Agent role	4.98	4.94	-0.04	0.46
Self efficacy	4.42	4.78	0.36	0.18
Learning outcome	**3.74**	**4.18**	**0.44**	**0.07**

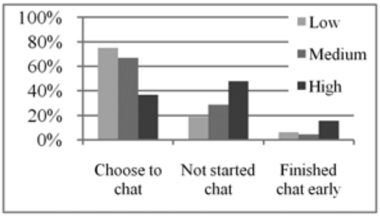
Fig. 1. The table and diagram shows the difference between the NoC- and WithC groups regarding: game experience, the agent's role in the game, self efficacy, and the learning outcome in terms of how well they did train their agent

Table 1 presents the results separated in sub-groups with respect to students' achievements in mathematics. For the low-achievers we see no differences between the WithC and NoC conditions. However, for the medium and high achieving students the experience of the game is considerably more positive for the WithC condition (diff=0.71, p=0.04 and diff=0.91, p=0.09). High-achievers in the WithC condition also rate their self efficacy beliefs significantly higher (diff=0.93, p=0.04) and have a superior learning outcome (diff=7.65, p=0.06).

Table 1. Questionnaire ratings and learning outcomes for low, medium and high achievers

	Low achieving				Medium achieving				High achieving			
	NoC	WithC	Diff	p	NoC	WithC	Diff	p	NoC	WithC	Diff	p
Game experience	4.92	3.58	-1.35	0.12	**5.24**	**5.95**	**0.71**	**0.04**	**5.12**	**6.03**	**0.91**	**0.09**
Agent role	4.58	3.31	-1.27	0.21	4.89	4.77	-0.12	0.41	5.34	5.71	0.37	0.3
Self efficacy	4.05	2.5	-1.55	0.11	4.48	4.85	0.37	0.25	**4.52**	**5.44**	**0.93**	**0.04**
Learning outcome	3.06	3.06	0.003	0.5	3.87	4.09	0.22	0.29	**3.91**	**4.68**	**0.77**	**0.06**

For the WithC group we further analysed the chat behaviour for the different subgroups. As shown in Fig. 2, there is a clear pattern where low and medium achievers choose to chat to a much higher extent than high achievers. Comments from students during the lessons indicate that at least some high achievers are quite task oriented and focus at the task at hand, i.e. to teach the agent, and so choose not to chat.



	Total	Low	Medium	High
Choose to chat	56%	75%	67%	36%
Not started chat	35%	19%	29%	48%
Finished chat early	9%	6%	4%	16%

Fig. 2. Table and diagram showing the difference in how low-, medium- and high achievers choose to chat when given the choice

6 Discussion

The primary result of the study is the indication that an added off-task conversation module i) can improve students' game experience and ii) is not necessarily a disadvantage in terms of learning accomplishment, but can to the contrary improve learning. This adds further support to our and others' approaches to the introduction of socially oriented off-task conversation as an integral learning element – in contrast to approaches where off-task behaviour is regarded to divert attention from learning thereby reducing the pedagogical efficiency (e.g. [20]). We hold both kinds of approaches valid, but advocate more nuances in the term "off-task behavior/conversation" in pedagogical contexts, and specifically for digital learning environments. The unit of learning is, we hold, a crucial parameter. For software meant to be used during a set, limited time and in relation to clear learning objectives, it may indeed be relevant to find means to control and even minimize off-task, and also be relatively easy to determine whether a behaviour indeed is unrelated to the curriculum in question. But in relation to a longer term learning context, another kind of balancing must be considered. Off-task behaviour can be essential for the development of a relation between agent and student, which can be central for reaching certain learning goals in longer term. The teachable agent based game discussed in this paper is this kind of longer term learning environment, and it is in view of this that we regard the off-conversation or chat module promising. However, for *low achievers* – in contrast to middle and high achievers – the chat was not associated with a more positive game experience nor by increased self-efficacy, which prompts further research. One possible, yet speculative, question is whether the addition of linguistic elements and written language to the game is troublesome for low-achievers. This is something that we will need to look further into.

References

1. Graesser, A., Chipman, P., Haynes, B., Olney, A.: AutoTutor: An intelligent tutoring system with mixed-initiative dialog. IEEE Trans. in Education 48, 612–618 (2005)
2. Kumar, R., Gweon, G., Joshi, M., Cui, Y., Rose, C.P.: Supporting students working together on math with social dialogue. In: Proc. the SLaTE Workshop on Speech and Language Technology in Education, pp. 96–99 (2007)
3. Cooper, B., Baynham, M.: Rites of passage: embedding meaningful language, literacy and numeracy skills in skilled trades courses through significant and transforming relationships. National Research and Development Centre for Adult Literacy and Numeracy (2005)
4. Hamann, S.: Cognitive and neural mechanisms of emotional memory. Trends in Cognitive Sciences 5(9), 394–400 (2001)
5. Chen, J., Shohamy, J., Ross, V., Reeves, B., Wagner, A.: The impact of social belief on the neurophysiology of learning and memory. Society for Neuroscience, San Francisco (2009)
6. Bickmore, T.: Relational Agents: Effecting Change through Human-Computer Relationships. PhD Thesis, Media Arts & Sciences, Massachusetts Institute of Technology (2003)
7. Kim, Y., Wei, Q., Xu, B., Ko, Y., Ilieva, V.: MathGirls: Increasing girls' positive attitudes and self-efficacy through pedagogical agents. In: Proc. AIED 2007, pp. 119–126 (2007)
8. Gulz, A., Haake, M., Silvervarg, A., Sjödén, B., Veletsianos, G.: Building a social conversational pedagogical agent – design challenges and methodological approaches. In: Perez-Marin, D., Pascual-Nieto, I. (eds.) Conversational Agents and Natural Language Interaction: Techniques and Effective Practices. IGI Global (2011)
9. Silvervarg, A., Gulz, A., Sjödén, B.: Design for off-task interaction – Rethinking pedagogy in technology enhanced learning. In: 10th IEEE Int. Conf. Adv. Learning Technologies (2010)
10. Maldonado, H., Lee, J., Brave, S., Nass, C., Nakajima, H., Yamada, R., Iwamura, K., Morishima, Y.: We learn better together. In: Proc. CSCL 2005, pp. 408–417 (2005)
11. Ai, H., Kumar, R., Nguyen, D., Nagasunder, A., Rosé, C.P.: Exploring the effectiveness of social capabilities and goal alignment in computer supported collaborative learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 134–143. Springer, Heidelberg (2010)
12. Wang, N., Johnson, W.L., Mayer, R.E., Rizzo, P., Shaw, E., Collins, H.: The politeness effect. Int. J. Human Computer Studies 66, 96–112 (2008)
13. Bickmore, T., Schulman, D., Yin, L.: Engagement vs. Deceit: Virtual humans with human autobiographies. In: Ruttkay, Z., Kipp, M., Nijholt, A., Vilhjálmsson, H.H. (eds.) IVA 2009. LNCS, vol. 5773, pp. 6–19. Springer, Heidelberg (2009)
14. Jakobson, R.: Linguistics and Poetics. In: Sebeok, T.A. (ed.) Style in Language, pp. 130–144. MIT Press, Cambridge (1960)
15. Mehlmann, G., Häring, M., Bühling, R., Wißner, M., André, E.: Multiple agent roles in an adaptive virtual classroom environment. In: Proc. IVA 2010, pp. 250–256 (2010)

16. Pareto, L., Haake, M., Lindström, P., Sjödén, B., Gulz, A.: A Teachable Agent Based Game Affording Collaboration and Competition (under revision)
17. Pareto, L., Schwartz, D., Svensson, L.: Learning by guiding a teachable agent to play an educational game. In: Proc. AIED 2009, pp. 662–664 (2009)
18. Biswas, G., Katzlberger, T., Bransford, J., Schwartz, D.: Extending intelligent learning environments with TA:s to enhance learning. In: Proc. AIED 2001, pp. 389–397 (2001)
19. Lindström, P., Haake, M., Sjödén, B., Gulz, A.: Matching and mismatching between the pedagogical design principles of a math game and the actual practices of play. J. Computer Assisted Learning 27, 90–102 (2011)
20. Rowe, J., McQuiggan, S., Robison, J., Lester, J.: Off-task behavior in narrative-centered Learning environments. In: Proc. AIED 2009, pp. 99–106 (2009)
21. Chase, C., Chin, D., Oppezzo, M., Schwartz, D.: Teachable agents and the protégé effect. J. Science Education and Technology 18(4), 334–352 (2009)
22. Anderson, L.W., Bourke, S.F.: Assessing affective characteristics in the schools, 2nd edn. Lawrence Erlbaum Associates, Mahwah (2000)
23. Bandura, A., Schunk, D.H.: Cultivating competence, self-efficacy, and intrinsic interest through proximal self-motivation. J. Personality and Social Psychology 41, 586–598 (1981)

Text Categorization for Assessing Multiple Documents Integration, or John Henry Visits a Data Mine

Peter Hastings[1,*], Simon Hughes[1], Joe Magliano[2], Susan Goldman[3], and Kim Lawless[3]

[1] DePaul University
[2] Northern Illinois University
[3] University of Illinois Chicago

Abstract. A critical need for students in the digital age is to learn how to gather, analyze, evaluate, and synthesize complex and sometimes contradictory information across multiple sources and contexts. Yet reading is most often taught with single sources. In this paper, we explore techniques for analyzing student essays to give feedback to teachers on how well their students deal with multiple texts. We compare the performance of a simple regular expression matcher to Latent Semantic Analysis and to Support Vector Machines, a machine learning approach.

Keywords: Natural Language Processing, Machine Learning, Corpus Analysis.

1 Introduction

In the digital age, literacy requires the reader more than ever before to be able to gather, analyze, evaluate, and synthesize complex and sometimes contradictory information across multiple sources and contexts [1]. Unfortunately, reading is typically taught and assessed using a single source text and rarely addresses comprehension and learning across multiple sources [2]. To improve this situation, teachers and students must be provided with educational curricula and tools that feature multiple-text comprehension and provide examples of tasks, texts and student performance in different subject matter areas [3–7]. In [2], we described the development of a formative assessment tool for characterizing a student's ability to comprehend and synthesize multiple texts. The goal of the current study is to develop and test techniques for providing automated assessment of the student essays.

As described in [2], 247 middle school students were given three texts describing different factors that led to the population boom in Chicago during the

[*] The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305F100007 to University of Illinois at Chicago. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 115–122, 2011.

mid-1800s. Each text focused on a different factor that either pushed people from their homes to Chicago (e.g., poor economic opportunities in rural areas), pulled people to Chicago (e.g., increase number of low-skilled jobs, jobs in the railroad industry), or the development of an infrastructure that supported a population increase (e.g., development of railroad and shipping industries). In this paper, these source texts are referred to as the "Better life", "Industry", and "Transportation" texts respectively. Students were told to read the texts and use the content to write an essay explaining why Chicago became a big city.

A critical component of the formative assessment tool is a theoretically-driven, ideal representation of how the texts could be used to answer the question of why Chicago became a big city, called a *documents model* [8, 9]. Created by discourse experts, the documents model is a graph which depicts the cause-and-effect relationships within the set of source texts, as well as the specific details that support these relationships. For example, code CL1 represents the most general level of the pull factors that brought people to Chicago. Code SCL1.1 represents an underlying cause, e.g. "businesses grew." Code SCL1.2 represents the effect of that cause, e.g. "jobs were created." Code ESCL1.2 is a specific example of job creation in meat processing industries. There are 37 codes in the documents model representing the concepts and relationships of the three texts.

In this paper, we describe our efforts to automatically identify the overlap between the student texts and the original source texts. We start by describing the corpus of student texts. Then we present a simple text classification method in which a human expert creates regular expressions to identify student sentences which correspond to a particular documents model category. In section 4, we evaluate Latent Semantic Analysis for classifying the student texts. Then we describe a machine learning approach to the classification problem, and finish with a comparison of the approaches.

2 The Corpus

As described above, our classification task is to determine how student essays relate to the original source texts. Our training data for the different methods was the set of student essays mentioned above that had been coded by human analysts. We worked with 459 student essays collected in 2008 and 2009, consisting of a total of 4076 sentences.

As reported in [2], each student sentence was given a (possibly empty) set of "text codes" that indicated which particular sentence(s) from the three sources it related to. Each sentence was also given a (possibly empty) set of documents model codes which indicated the related concepts from the documents model. For example, the student sentence, "Many people also worked in the meat processing by cutting the cattle and pigs" was coded with text code I16 for sentence 16 of the Industry text: "Butchers cut the cattle and pigs into the meat that people bought in grocery stores." It was coded with documents model code ESCL1.2, described above.

The annotated texts were translated into XML to facilitate the creation of multiple views of the text, for example, sorting by source category, or documents model concept. The sentences were preprocessed by removing punctuation and stop words (using the CLEF english stopword list available from http://members.unine.ch/jacques.savoy/clef/englishST.txt) and eliminating words which only occurred in one document. We did not use stemming. All words were upcased.

3 Pattern Matching

Our initial approach to classifying student texts used a tried and true approach: pattern matching with regular expressions. In the spirit of [10], we thought that human ingenuity, combined with a simple technique and a quick and convenient method for refining results might be fruitful. For this analysis, we wanted to determine how well we could identify which student sentences were associated with the codes in the documents model (DM). We created a web-based tool which displayed all the student sentences, sorted by DM code. For each code, it allowed the user to create a regular expression using terms and wildcards. For example, the pattern: `(meat (processing | packaging) * (industry | industries | factories))` matches any sentence that includes the word “meat” followed by “processing” or “packaging” followed by any number of other words and then “industry”, “industries”, or “factories”. The user can submit the set of patterns and receive almost instantaneous feedback about the performance of those patterns in classifying the student sentences in accordance with the human coding.

The concept nodes in the documents model (DM) are arranged hierarchically. The nodes at the top of the hierarchy represent the most general statements about the assigned topic, and therefore can be expressed in great variety of ways. Lower level nodes represent more specific information, which is more likely to be expressed with predictable content words, so we focused our efforts on developing patterns to match these lower level nodes (14 of the 37 total). Table 1 presents the performance of the patterns (and the aggregate) in terms of Recall *(true positives / (true positives + false negatives))*, Precision *(true positives / (true positives + false positives))*, and F_1 score *(2*Precision*Recall/(Precision+Recall))*.

Overall, the performance of this set of patterns was at least respectable and in some cases, very good. Some of the patterns were very simple. For SCL2.2, the pattern was simply a disjunction of the terms, “families”, “family”, or “feed”, and

Table 1. Matching documents model codes with regular expressions

DM code	Rec.	Pre.	F_1	DM code	Rec.	Pre.	F_1	DM code	Rec.	Pre.	F_1
ESCL1.1	0.78	0.61	0.68	ESCL2.4	0.78	0.94	0.85	SCL2.1	0.75	0.63	0.68
ESCL1.2	0.73	0.69	0.71	ESCL3.1	0.74	0.56	0.64	SCL2.2	0.92	0.56	0.70
ESCL1.3	0.69	0.80	0.74	ESCL3.2	0.66	0.25	0.36	SCL3.1	0.72	0.74	0.73
ESCL2.1	0.84	0.94	0.89	SCL1.1	0.76	0.86	0.81	SCL3.2	0.62	0.27	0.38
ESCL2.3	0.83	0.93	0.88	SCL1.2	0.78	0.30	0.43	Aggregate	0.76	0.78	0.77

it achieved a very high Recall value. Its Precision was moderate, however, because a significant number of sentences associated with other codes also included these terms. This highlights the difficulty of the "semantic overlap problem". In the case of "hand-built" mechanisms like this one, the problem is especially difficult because there is no way to know if a particular pattern is optimal or how close to optimal it is. For this reason, and to allow a broader coverage of the classification space, we explored automatic methods of classification using Latent Semantic Analysis and Machine Learning.

4 Latent Semantic Analysis

Latent Semantic Analysis (LSA) has been used in a wide range of cognitive modeling and educational tasks [11]. It uses singular value decomposition to create a vector-based representation of the words and documents in the training corpus, and can then compare documents with the cosine measure. Because the nodes in the documents model are conceptual and not textual, we used LSA to compare the sentences of the student essays with the original source sentences (the text model, or TM). This can be used as a proxy for the conceptual analysis, because the documents model includes a mapping from the text model codes to the documents model codes.

We used LSA from `http://lsa.colorado.edu` with the "General Reading up to 1st year college (300 factors)" space to calculate the cosine similarity between each student sentence and each sentence in the three source documents that the students read. If the cosine was greater than a threshold, we assigned the relevant TM code to the student sentence. As with the coder annotations, this allowed multiple TM codes per student sentence. Because the threshold must be empirically derived, we used a range of cosine thresholds (0.4, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, and 0.8). The results are shown in Table 2. The trade-off between Recall and Precision can be clearly seen across the different threshold values. The best result, using F_1 which gives Recall and Precision equal weight, was achieved with a cosine threshold of 0.70.

Table 2. Evaluation of LSA with different cosine thresholds

Threshold	0.40	0.50	0.55	0.60	0.65	0.70	0.75	0.80
Recall	0.70	0.63	0.58	0.53	0.48	0.43	0.38	0.34
Precision	0.14	0.24	0.31	0.41	0.53	0.66	0.75	0.80
F_1	0.23	0.35	0.40	0.46	0.50	0.52	0.50	0.48

5 Machine Learning

In a machine learning approach to text classification, some set of features of the texts are used to induce a classifier that should correctly categorize as many of the texts as possible. The most obvious features of a document are the words within it. One popular learning method for this type of classification task is Support Vector Machines (SVMs) [12, 13]. In this section we describe other applications of text classification techniques in educational contexts and then present our approach and evaluations of it.

5.1 Related Work

Although there have been a great many applications of machine learning in text classification for information retrieval, there have been relatively few within an educational context, and most of them have been aimed at inferring dialog acts, for example [14]. More similar approaches to ours include Larkey's [15] comparison of k-nearest neighbor, naïve Bayes and linear regression classifiers in assigning grades to student essays. Sathiyamurthy and Geetha [16] built a text classification system which used part-of-speech tagging to align e-learning documents according to an ACM domain ontology, allowing the documents to be classified according to Bloom's taxonomy [17]. Yilmazel et al [18] used an SVM algorithm to perform text categorization for automatically aligning curricular documents with state and federal science benchmarks.

5.2 SVMs for Text Classification

A typical task for text classification is learning to categorize news articles by topic. In [12], for example, SVMs were trained to identify the topic of 800,000 news stories from Reuters at three different levels of granularity. Two important differences between that study and ours are the size of the individual documents and the size of the training set. Because we would ideally like to give teachers information about which concepts from the documents model are included in the student essays, we are most interested in classifying individual sentences (as opposed to paragraph-length or longer documents). As mentioned above, our entire corpus consists of approximately 4000 sentences, two orders of magnitude less than Medlock used.

To create the training data for the SVMs, we separated the student essays into sentences (= documents) and preprocessed them as described above (removing stop words, etc.). Then we computed normalized *tfidf* vectors for each document following [13]. Each document vector had a weight for each of the terms in it. The weight for a term was computed as the number of times it occurs in the document divided by the log of the number of documents it occurs in. Then each vector is normalized to have length = 1 to allow comparison of documents with differing numbers of terms.

We used 10-fold cross-validation along with svm_multiclass [13] and trained the classifiers to categorize the sentences into the 37 documents model categories. The results from the best performing model are shown in Table 3. For comparison, the 14 DM codes which were also included in the pattern matching evaluation are shown in italics. The penultimate entry is the aggregate across all categories. The row labelled "Aggr 14" shows the aggregate results across the 14 codes from the pattern matching evaluation.

Table 3. SVM performance for DM codes

DM code	Rec.	Pre.	F_1	DM code	Rec.	Pre.	F_1	DM code	Rec.	Pre.	F_1
A	0.59	0.42	0.49	ESCL3	0.00	0.00	0.00	RC3	0.05	0.03	0.04
CL1	0.20	0.31	0.24	*ESCL3.1*	0.63	0.44	0.52	RC3.1	0.10	0.25	0.14
CL2	0.49	0.40	0.44	*ESCL3.2*	0.58	0.47	0.52	RC3.2	0.00	0.00	0.00
CL3	0.44	0.44	0.44	IRC1	0.02	0.10	0.03	RC3.3	0.08	0.19	0.11
ESCL1	0.59	0.45	0.51	IREN1	0.00	0.00	0.00	RE1	0.00	0.00	0.00
ESCL1.1	0.80	0.51	0.62	IREN2	0.00	0.00	0.00	*SCL1.1*	0.29	0.33	0.31
ESCL1.2	0.85	0.52	0.65	RC1+2	0.00	0.00	0.00	*SCL1.2*	0.46	0.46	0.46
ESCL1.3	0.87	0.65	0.74	RC1.1	0.08	0.26	0.12	*SCL2.1*	0.20	0.28	0.23
ESCL2	0.04	0.14	0.06	RC1.2	0.00	0.00	0.00	*SCL2.2*	0.24	0.27	0.25
ESCL2.1	0.63	0.43	0.51	RC2.1	0.06	0.14	0.08	*SCL3.1*	0.16	0.27	0.20
ESCL2.2	0.60	0.51	0.55	RC2.2	0.07	0.20	0.10	*SCL3.2*	0.02	0.25	0.04
ESCL2.3	0.72	0.49	0.58	RC2.3	0.06	0.17	0.09	Aggregate	0.42	0.42	0.42
ESCL2.4	0.70	0.49	0.58	RC2.3A	0.01	0.17	0.02	*Aggr 14*	0.54	0.45	0.49

Of the codes that were matched with the regular expression approach, the SVM often achieved better Recall but worse Precision. As mentioned above, "casting a broader net" increases Recall, but reduces Precision. Overall, these results confirm our intuition that the more specific concepts would be the easier ones to match. The exception to this is the A code (for Assertion). This is a sort of "catch-all" category that indicates a factual statement made by the student which is not directly derived from any of the sources. Despite the breadth of this category, the SVM achieved respectable performance in identifying it. It must be mentioned, however, that the A code is the most frequent one in the corpus, assigned to almost 1300 sentences, 18% of the total of 7321 TM codes given by the human coders. This compares with an average of 191 sentences (2.6%) for the codes in the subset of 14 used in pattern matching. Thus, it is possible, and perhaps even likely, that the SVM's performance on those more specific categories suffered for the benefit of overall performance. This will be discussed further in the next section.

6 Discussion, Future Work, and Conclusions

As shown above, among the 14 DM codes that pattern matching was applied to, the SVM approach significantly outperformed the pattern matching approach in only one of the categories, ESCL3.2. For the rest, pattern matching was close or much better. When training the SVM, we noticed that with tighter margins between the learned set of support vectors and the training set (lower values of the C parameter), prediction of many of the semantic categories was good, except for the catch-all A category. Because it is the most frequent, that had a large effect on the overall performance. By increasing the margin, we were able to improve performance on A and overall, but with reduced performance on the categories which had fewer examples in the training sets. However, we also tried creating binary classifiers for each DM code (not reported here due to space limitations). This would at least partially address the concern about the relative frequencies of the categories. Each binary classifier only has to distinguish the members vs. non-members of one category. There is still an effect, however, of the small number of positive instances of the more specific categories relative to the entire training set. The binary classifiers that we created generally achieved good Recall but poor Precision.

If we use the entire set of categories, we can (almost) directly compare the three approaches, but pattern matching gets a much lower Recall score (0.18) due to the missing codes. In this comparison, $F_1(Patterns) = 0.29$, $F_1(LSA) = 0.52$, and $F_1(SVD) = 0.42$. Although LSA matched student sentences with TM codes instead of DM codes, the aggregate measure should provide a good idea of the overall performance. We suspect that LSA had an advantage over SVM because many of the student sentences were close paraphrases of the source sentences. We should be able to check this by inferring DM codes from TM codes. This will be done in future work.

One advantage that the pattern matching approach has over the others is that it can take the ordering of the words into account. This could be addressed in a machine learning context by using n-grams or term identification methods. In future work, we will also explore other variations of the machine learning methods, including different classification techniques and higher-level approaches like boosting. If pattern matching retains its advantage for particular codes, a hybrid approach can be developed.

In this paper, we explored three text classification methods, pattern matching, LSA, and SVMs. For identifying many of the specific semantic categories, pattern matching performance exceeded that of the automatic methods. Despite the limitations of the pattern matching approach — the difficulty of coming up with appropriate patterns for all of the categories and the impossibility of knowing what an optimal pattern is — we believe that such a simple technique can still be effective and useful in an educational context.

References

1. New London Group: A pedagogy of multiliteracies: Designing social futures. Harvard Educational Review 66, 60–92 (1996)
2. Goldman, S.R., Lawless, K.A., Gomez, K.W., Braasch, J.L.G., MacLeod, S., Manning, F.: Literacy in the digital world: Comprehending and learning from multiple sources. In: McKeown, M.G., Kucan, L. (eds.) Bringing Reading Researchers to Life, Guilford, NY, pp. 257–284 (2010)
3. Britt, M.A., Wiemer-Hastings, P., Larson, A., Perfetti, C.: Using intelligent feedback to improve sourcing and integration in students' essays. International Journal of Artificial Intelligence in Education 14, 359–374 (2004)
4. Britt, M.A., Kurby, C., Dandotkar, S., Wolfe, C.: I agreed with what? Memory for simple argument claims. Discourse Processes 45(1), 52–84 (2008)
5. Goldman, S.R., Bloome, D.M.: Learning to construct and integrate. In: Healy, A.F. (ed.) Experimental Cognitive Psychology and its Applications: Festshrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer, pp. 169–182. American Psychological Association, Washington, D.C (2005)

6. Wolfe, M.B., Goldman, S.R.: Relationships between adolescents' text processing and reasoning. Cognition & Instruction 23(4), 467–502 (2005)
7. VanSledright, B.: Confronting history's interpretive paradox while teaching fifth graders to investigate the past. American Educational Research Journal 39, 1089–1115 (2002)
8. Rouet, J.F.: The skills of document use. Erlbaum, Mahwah (2006)
9. Rouet, J.F., Britt, M.A.: Relevance processes in multiple document comprehension. In: McCrudden, M.T., Magliano, J.P., Schraw, G. (eds.) Text Relevance and Learning from Text. Information Age Publishing, Greenwich (in press)
10. Hobbs, J., Appelt, D., Tyson, M., Bear, J., Israel, D.: SRI International: Description of the FASTUS system used for MUC-4. In: Proceedings of the Fourth Message Understanding Conference. Morgan Kaufmann Publishers, Inc., San Mateo (1992)
11. Landauer, T., Dumais, S.: A solution to Plato's problem: The Latent Semantic Analysis theory of acquisition, induction, and representation of knowledge. Psychological Review 104, 211–240 (1997)
12. Medlock, B.: Investigating classification for natural language processing tasks. PhD thesis, University of Cambridge, Technical Report UCAM-CL-TR-721 (2007)
13. Joachims, T.: Learning to Classify Text Using Support Vector Machines. PhD thesis. Cornell University. Kluwer (2002)
14. Samuel, K., Carberry, S., Vijay-Shanker, K.: Computing dialogue acts from features with transformation-based learning. In: Papers from the 1998 AAAI Spring Symposium on Applying Machine Learning to Discourse Processing, pp. 90–97. AAAI Press, Menlo Park (1998) Number SS-98-01
15. Larkey, L.S.: Automatic essay grading using text categorization techniques. In: Proceedings of SIGIR 1998, pp. 90–95 (1998)
16. Sathiyamurthy, K., Geetha, T.V.: Association of domain concepts with educational objectives for e-learning. In: Proceedings of Compute 2010, pp. 330–333 (2010)
17. Bloom, B. (ed.): Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain. Longmans, Green (1956)
18. Yilmazel, O., Balasubramanian, N., Harwell, S.C., Bailey, J., Diekema, A.R., Liddy, E.D.: Text categorization for aligning educational standards. In: Proceedings of the 40th Hawaii International Conference on System Sciences, pp. 73–80 (2007)

Learning by Problem-Posing for Reverse-Thinking Problems

Tsukasa Hirashima and Megumi Kurayama

Learning Engineering Group, Information Engineering, Graduate School of Engineering,
Hiroshima University,
1-4-1, Kagamiyama, Higashi-Hiroshima, Hiroshima, Japan
tsukasa@isl.hiroshima-u.ac.jp

Abstract. Learning by problem-posing is a promising way to learn arithmetic or mathematics. We have already developed several interactive learning environments for learning by problem-posing. In this research, we have paid a special attention to "reverse-thinking problems" in arithmetic word problems that can be solved either by addition or subtraction. In the reverse-thinking problems, since "story operation structure" and "calculation operation structure" are different, they require learners to comprehend the relations between problems and solutions more than "forward thinking problems" where "story operation structure" and "calculation operation structure" are the same ones. Based on a learning environment for posing the forward thinking problems developed previously, we have expanded it for reverse thinking problems. This learning environment has been used in a class of fourth grade at an elementary school for eight lesson times. We have also reported the results of this practical use.

Keywords: Problem-posing, Reverse thinking problem, Story operation structure, Calculation operation structure.

1 Introduction

Learning by problem-posing is an alternative and promising way to promote learners to master the use of solution methods [1,2,3]. In the problem-posing, however, since learners are usually allowed to pose several kinds of problems, it is difficult for teachers to complete assessment and feedback for the posed problems in classroom practically. Therefore, it is not popular as a teaching method even though its effectiveness is well-known. Based on these considerations, in order to make this teaching method practical one in classroom, we have been investigating computer-based learning environments that can assess and give feedback to each posed problem [4,5].

As a realization of the goal, we have already designed and developed an interactive environment for learning by problem-posing, named MONSAKUN (a problem-posing kid in Japanese), for arithmetic word problems that are solved by one operation of either

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 123–130, 2011.

addition or subtraction [6]. In MONSAKUN, "solution-based problem-posing" [7] as "sentence integration" has been realized. In the problem-posing in MONSAKUN, a learner is provided with a set of sentence cards and a calculation expression. For examples, "5+3" or "7-4", and then, the learner is required to pose a problem that can be solved by the calculation (that is, solution). One sentence represents an object or event, countable attribute and a value of the attribute. A learner is required to pose a problem by selecting and ordering the cards. Because machine readable metadata is attached to each card and the domain is tightly restricted, it is possible to adequately diagnose problems posed in this environment. In this problem-posing, then, although learners do not make sentences, they are required to interpret the provided sentences. Moreover, they have to integrate the sentences into one problem in the same way with usual problem-posing. Several investigation of problem-solving or understandings have already indicated that this integration process is an essential activity in the learning. Since the focus of this problem-posing method is on "integration phase" of general model of problem-solving process of arithmetic word problems, we call this problem-posing as "problem-posing as sentence-integration".

In our previous version of MONSAKUN, however, only "forward thinking" problems are dealt with. An arithmetic word problem that can be solved by arithmetic operations includes two kinds of numerical relations; one is story operation structure and the other is calculation operation structure. For example, in the following problem, the story operation structure is "3 + 4 = ?".

{Tom had 3 pencils. Tom bought 4 pencils. Tom has several pencils. How many pencils does Tom have?}

In this story, since Tom obtained more pencils, the number of his pencils increases. Therefore, the story focuses on "increase story". Then, the calculation operation structure becomes as "3 + 4 (=?)". Hence, the two structures are the same one. This kind of problem is usually called "forward thinking problem" because calculation operation structure is able to find by reading and understanding the story from the first sentence in order. This forward thinking problem is usually easy for learners to solve.

As for the following problem where only the unknown value is replaced against the previous problem, story operation structure "3 + ? = 7" and calculation operation structure: "7 – 3(= ?)" are different.

{Tom had 3 pencils. Tom bought several pencils. Tom has 7 pencils in total. How many pencils did Tom buy?}

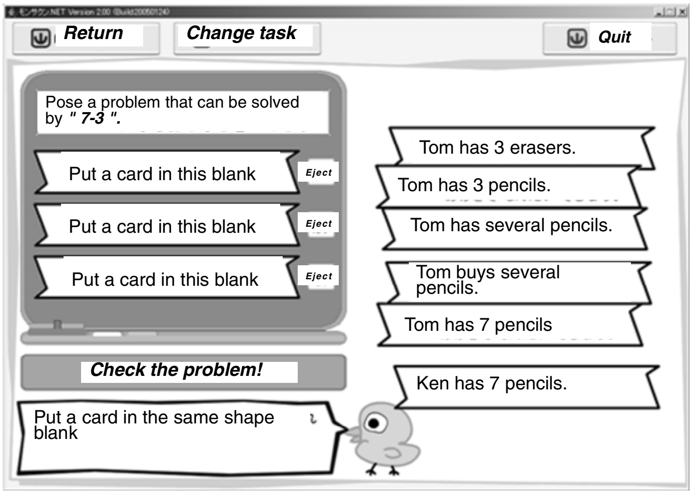
This kind of problem is usually called "reverse thinking problem" because learner is required to think about the calculation method only after understanding the story. To know the difference between thinking about the story and the calculation is one of the most important purposes to learn arithmetic word problems, the reverse thinking problem is indispensable topic the learners have to overcome. Therefore, we focused on implementation of problem-posing of the reverse thinking problems in the current phase of our investigation.

In the following sections, implementation of MONSAKUN for reverse thinking problems and a practical use of it are explained.

2 Implementation of MONSAKUN

2.1 Interface of MONSAKUN

The interface of problem-posing in MONSAKUN is shown in Figure 1. The area on the left side, imaged blackboard, is "problem-composition area". At the top, a calculation expression is given. A learner would pose a problem which will be solved by the calculation expression, that is, either by an addition or subtraction. Several sentence cards are presented at the right side of the interface. To pose a problem, the learner selects several sentence cards and arranges them in a proper order. Although interpretation of each sentence is easy, the learner has to consider the relation among them to pose an adequate problem including the suitable relation for the calculation expression. This process is usually called “sentence integration” where structural understandings of problems and calculations play a crucial role [8,9,10].



Pose a problem that can be solved by *" 7-3 ".*

Tom has 3 erasers.

Tom has 3 pencils.

Tom has several pencils.

Tom buys several pencils.

Tom has 7 pencils

Ken has 7 pencils.

Put a card in the same shape blank

Fig. 1. Interface of MONSAKUN

A sentence card is put into a blank in the problem-combination area. There are three blanks in Figure 1, a learner should select three cards from the card set at right side and arrange them in a proper order. A learner can move a card by drag & drop method in the interface. When a learner pushes "diagnosis button" under the problem-composition area, the system diagnoses the combination of sentences. The results of the diagnosis and message to help the learner's problem-posing is presented by another window.

2.2 Task Model

In order to deal with the reverse thinking problems, we have proposed a task model of problem-posing as sentence-integration shown in Figure 2. The task model of problem-posing consists of following four tasks, (1) selecting calculation operation structure, (2) selecting story operation structure, (3) selecting story structure, and (4) selecting problem sentences. A learner should complete these tasks to pose a problem correctly though the execution procedure of the tasks is not decided in the model.

Here, it is assumed that the task is executed from (1) to (4) in order to explain the tasks. In the first step of MONSAKUN, subtraction or addition is selected as a calculation operation. In the second step, a story operation structure is decided. For example, for subtraction, four story operation structures can be selected. Only one story operation structure "x-y=?" is same with the calculation operation structure and others are different, that is, "?+y=x", "y+?=x", "x-?=y". Because these variations require learners to carry out abstract structure transformation it is often very difficult for learners to pose and solve them.

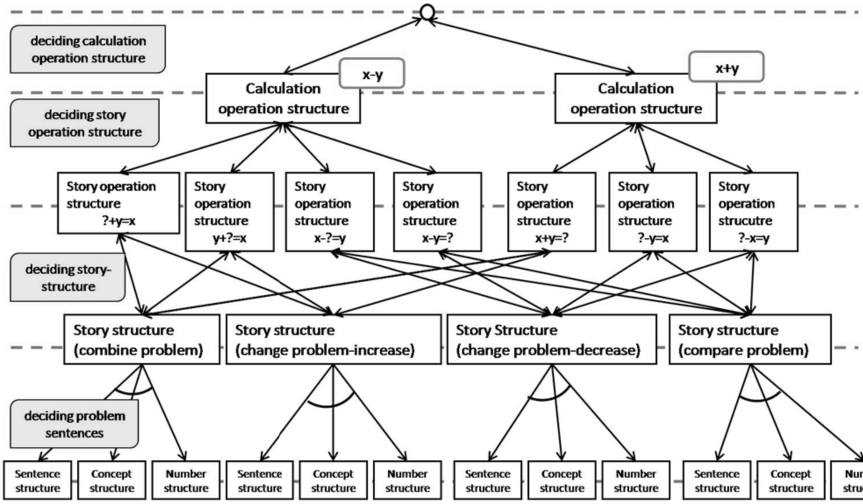

deciding calculation operation structure
x-y
x+y
Calculation operation structure
Calculation operation structure
deciding story operation structure
Story operation structure ?+y=x
Story operation structure y+?=x
Story operation structure x-?=y
Story operation structure x-y=?
Story operation structure x+y=?
Story operation structure ?-y=x
Story operation strucutre ?-x=y
deciding story-structure
Story structure (combine problem)
Story structure (change problem-increase)
Story Structure (change problem-decrease)
Story structure (compare problem)
deciding problem sentences
Sentence structure
Concept structure
Number structure

Fig. 2. Task Model of Problem-Posing

Arithmetic word problems solved by one addition or subtraction are usually categorized into four types: 1) increase-change, 2) decrease-change, 3) combine, and 4) compare [11]. Each type of problem has its own structures. For example, decrease-change problem is composed of "existence sentence (there were seven apples)", "decrease sentence (several apples were eaten)" and "existence sentence (there are four apples now)". In the phase of selecting story structure, a learner should select one of them.

In selecting problem sentences, sentences are put into the story structure following the story operation structure. This task is divided into three more tasks: selecting sentence structure, selecting concept structure and selecting number structure. The selecting sentence structure means that to select and order sentences following the story structure. For example, if the story structure is the decrease-change, make a sentence structure composed of the existence sentence, the decrease sentence, and the existence sentence in turn. In the decision of concept structure, concepts dealt with the problem are decided. For example, if the problem is requested to answer about the total number of apples and oranges, then the sentences should be dealt with the apples and oranges. In the decision of number structure, the numbers dealt with the problem is decided. In arithmetic word problems, a negative number should not be used.

In the previous version of MONSAKUN, since learners are given a story operation structure (only corresponding to forward thinking problem) and a story structure and required to pose a problem, their task was only selecting the problem sentences.

2.3 Setting of Problem-Posing Exercises

In MONSAKUN, problem-posing exercises are categorized into 8 levels based on the tasks included in the problem-posing. In the first level, learners' task is only selecting the problem sentences. In the second level, the learners are required to decide both story structure and problem sentences. In this level, although the reverse thinking problems are also posed, because story operation structure is given to the learners directly, they don't have to think about the difference between the story operation structure and the calculation operation structure. In the third level, learners are required to carry out decision task of story operation structure based on calculation operation structure. Here, the learners should be aware of the difference between the story and the calculation. In the 4th level, learners are required two different problems for a specific addition calculation and in the 5th level, for subtraction. In the 6th level, to pose all kind of problems for a specific addition calculation is the problem-posing assignment, and then in the 7th level for a specific subtraction calculation. In the 8th level, learners are required to pose all kind of problem that can be solved by an addition or subtraction by using a set of sentence cards.

2.4 Diagnosis and Feedback

MONSAKUN has an ability to solve the posed problems. It can also pose adequate problems for each problem-posing assignment based on the task model. In MONSAKUN, a posed problem can be diagnosed by solving and comparing it with the adequate problems based on the task model. When the posed problem cannot be solved, the system would point out to the learner that the posed problem is an unsolvable one. The unsolvable reasons are categorized into the following three: (1) there is no unknown value, (2) the calculation result becomes a negative value, and (3) it is impossible to calculate. In the cases of (1) and (2), the reasons are pointed out directly. In the case of (3), the problem structure includes some defects in sentence structure, concept structure and/or number structure. If it is possible to correct the posed problem to adequate one by replacing one sentence card, the system indicates the sentence card that the student should replace it. If it is not, the system indicates the error types, that is, an error in sentence structure, concept structure and number structure. In the current system doesn't teach a correct one directly.

If the posed problem is solvable but is not adequate for the problem-posing assignment, the system explains the posed problem and indicates difference from the assignment. For example, although the assignment requires a student to pose a problem solved by “5+3”, the student might pose a problem that can be solved by “5-3”. In this case, the system indicates the student that the posed problem is solved by 5-3 but the request is “5+3”.

3 Experimental Use of MONSAKUN

In this section, we have reported the results of practical use of MONSAKUN at an elementary school. In this experimental use, 39 students of fourth grade used MONSAKUN in arithmetic classroom (two students were absent from the pre-test and one student was absent from the post-test and questionnaire). They used eight lesson times (45 minutes per lesson) in 13 weeks. The students took a pretest before the use and a post-test and questionnaires after the use. In the pre- and post-tests, a learner is required to pose four problems by composing several sentence cards provided beforehand.

The average number of the problems that a student posed in this experiment was 269, and the average number of the correct problems was 192. There were 1.27 problems posed to complete a forward thinking problem and 2.90 problems to complete a reverse thinking problem. In this, 6 learners reached Level-6, 9 learners to Level-5 and 20 learners to Level-4. Only 4 learners had not completed Level-4 that is the first step of posing reverse-thinking problems.

3.1 Learning Effects

Effects of the learning with MONSAKUN were examined by comparing the results of pre-test and post-test where learners posed four problems without MONSAKUN. We checked them whether they were correct or not, and categorize correct ones into forward thinking problems and reverse thinking problems. We also categorized learners into high score group and low score group based on the pre-test score. The results shown in Table 2 and Figure 3 were analyzed with a three-way 2 (high score group or low score group) x 2 (pre-test score or post-test score) x 4 (posed problems, correct problems, forward thinking problems or reverse thinking problems) mixed ANOVA, multiple comparison was made using Ryan's method. As the results, problem-posing performance of the low group was improved in post-test dramatically (ex. correct problems: p=0.000, η^2=0.410). As for the high group, the number of forward thinking problems decreased (p=0.0052, η^2=0.060) but the number of reverse thinking problems increased (n.s., p=0.16, η^2=0.014). The number of reverse thinking problems made more than forward thinking problems only in the post-test of the high group (p=0.000 ; r=0.25).

Based on these results, we have confirmed that MONSAKUN is useful for the low score group, at least. As for high score group, these results suggest that the learners in the group intentionally tried to pose the reverse thinking problems and avoided to pose forward thinking problems although they sometimes made mistakes. Although clear learning effect could not be confirmed, it is suggested that the learners in the high score group were promoted to be aware of the difference between reverse thinking problems and forward thinking problems more clearly.

Table 1. Results of Pre-test and Post-test

	Posed Problems(PP)		Correct Problems(CP)		Forward Thinking Problems (FTP)		Reverse Thinking Problems(RTP)	
	Pre	Post	Pre	Post	Pre	Post	Pre	Post
High Score Group(n=21)	3.90 (SD=0.29)	4 (0.00)	3.48 (0.50)	3.10 (0.92)	1.76 (0.87)	1.00 (0.76)	1.71 (0.83)	2.10 (1.02)
Low Score Group(n=15)	2.47 (1.02)	4 (0.00)	1.33 (0.79)	3.33 (0.70)	0.73 (0.85)	1.53 (0.88)	0.60 (0.71)	1.80 (1.05)

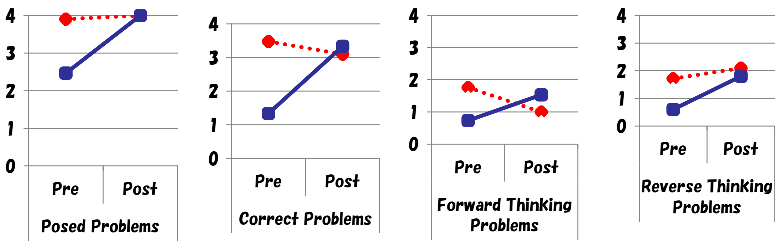


Fig. 3. Graphs of the results of Table 1. (Solid line: High group, Dashed line: Low group)

3.2 Questionnaires

The results of questionnaires are shown in Table 2. More than 70% students answered that they enjoyed posing problems with MONSAKUN though only 20% students considered it as easy. Almost 80% students considered that their ability to pose problems was improved and the activity was important in arithmetic learning. Even after the long term use, more than 70% students are expected to use it more. These results suggested that MONSAKUN was accepted by the students as a useful tool in learning. The teacher who was in-charge of the classes also agreed to these answers and their considerations.

Since MONSAKUN was used in eight formal lesson times in three months, these results suggest that MONSAKUN is a promising application that realizes learning by problem-posing at school. However, this experiment was small in size with the number of subjects and also we did not have a control group. Moreover, we evaluated the effect by the scores of problem-posing. Based on these considerations, we have been planning a larger size experiment in future to examine the effect of this learning.

Table 2. Results of Questionnaires

Answer Question	Yes	No	No idea
Did you enjoy posing problems with MONSAKUN?	29	9	0
Is it easy for you to pose problems?	8	21	9
Do you think you could make problems easier than before?	29	9	0
Do you think MONSAKUN is useful for arithmetic learning?	30	7	1

4 Concluding Remarks

We have proposed a task model of problem-posing that dealt with not only the forward thinking problem but also the reverse thinking problem. Based on the model, we have expanded MONSAKUN which is the interactive learning environment of problem-posing as sentence-integration. We then conducted practical use of MONSAKUN at formal arithmetic lesson at an elementary school. These results suggest that MONSAKUN is a useful tool to improve student's ability of problem posing and it is accepted by learners and teachers as a useful learning tool. We believe that these results are sufficient to confirm the possibility of computer-based learning by problem-posing.

From the viewpoint of evaluation, however, this experiment was smaller in size with the number of subjects and we did not have a control group. Moreover, the learning effect was evaluated by the scores of problem-posing not by the ability of arithmetic. Based on these considerations, we have been planning a larger sized experiment to examine the effect of this learning in future. Expansion of applicable domain of learning by problem-posing with agent-assessment is imperative for future work.

References

1. Polya, G.: How to Solve It. Princeton University Press, Princeton (1945)
2. Ellerton, N.F.: Children's Made Up Mathematics Problems: A New Perspective on Talented Mathematicians. Educational Studies in Mathematics 17, 261–271 (1986)
3. Yu, F., Liu, Y.H., Chan, T.W.: A Networked Question-Posing and Peer Assessment Learning System: a Cognitive Enhancing Tool. Journal of Educational Technology Systems 32(2), 211–226 (2003)
4. Nakano, A., Hirashima, T., Takeuchi, A.: Problem-Making Practice to Master Solution-Methods in Intelligent Learning Environment. In: Proc. of ICCE 1999, pp. 891–898 (1999)
5. Hirashima, T., Nakano, A., Takeuchi, A.: A Diagnosis Function of Arithmetical Word Problems for Learning by Problem Posing. In: Proc. of PRICAI 2000, pp. 745–755 (2000)
6. Hirashima, T., Yokoyama, T., Okamoto, M., Takeuchi, A.: Learning by Problem-Posing as Sentence-Integration and Experimental Use. In: Proc. of AIED 2007, pp. 254–261 (2007)
7. Silver, E.A., CAI, J.: An Analysis of Arithmetic Problem Posing by Middle School Students. Journal for Research in Mathematics Education 27(5), 521–539 (1996)
8. Kintsch, W., Greeno, J.G.: Understanding and Solving Word Arithmetic Problem. Psychological Review 92-1, 109–129 (1985)
9. Mayer, R.E.: Frequency norms and structural analysis of algebra story problems into families, categories, and templates. Instructional Science 10, 135–175 (1981)
10. Mayer, R.E.: Memory for algebra story problem. Journal of Educational Psychology 74, 199–216 (1982)
11. Riley, M.S., Greeno, J.G., Heller, J.I.: Development of Children's Problem-Solving Ability in Arithmetic. In: Ginsburg, H. (ed.) The Development of Mathematical Thinking, pp. 153–196. Academic Press, London (1983)

Affect Detection from Multichannel Physiology during Learning Sessions with AutoTutor

M.S. Hussain[1,2], Omar AlZoubi[2], Rafael A. Calvo[2], and Sidney K. D'Mello[3]

[1] National ICT Australia (NICTA), Australian Technology Park, Eveleigh 1430, Australia
[2] School of Electrical and Information Engineering, University of Sydney, Australia
[3] Institute for Intelligent Systems, University of Memphis, Memphis, USA
Sazzad.Hussain@nicta.com.au,
{omar.alzoubi,Rafael.Calvo}@sydney.edu.au,
sdmello@memphis.edu

Abstract. It is widely acknowledged that learners experience a variety of emotions while interacting with Intelligent Tutoring Systems (ITS), hence, detecting and responding to emotions might improve learning outcomes. This study uses machine learning techniques to detect learners' affective states from multichannel physiological signals (heart activity, respiration, facial muscle activity, and skin conductivity) during tutorial interactions with AutoTutor, an ITS with conversational dialogues. Learners were asked to self-report (both discrete emotions and degrees of valence/arousal) the affective states they experienced during their sessions with AutoTutor via a retrospective judgment protocol immediately after the tutorial sessions. In addition to mapping the discrete learning-centered emotions (e.g., confusion, frustration, etc) on a dimensional valence/arousal space, we developed and validated an automatic affect classifier using physiological signals. Results indicate that the classifier was moderately successful at detecting naturally occurring emotions during the AutoTutor sessions.

Keywords: Affective computing, emotion, AutoTutor, multichannel physiology, learning interaction, self reports.

1 Introduction

It has been widely acknowledged that cognition, motivation, and emotion are the key components of learning. During tutorial sessions with Intelligent Tutoring Systems (ITS) or human tutors, learners experience a host of learning-centered emotions such as confusion, boredom, engagement/flow, curiosity, interest, surprise, delight, anxiety, and frustration. These affective states are highly relevant and influential to both the processes and products of learning [1]. Therefore, researchers in the interdisciplinary arena encompassing psychology, education, neuroscience, and computer science have recently been focused on understanding the relationship between affect and learning [1-4].

Affect-sensitive ITSs aspire to detect and respond to learner emotions in order to improve learning gains along with increasing motivation and task interest [3]. These

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 131–138, 2011.

systems aim to reduce the gap between human tutors and computer tutors by endowing ITSs with a degree of emotional intelligence. Whether it is human or computer, a learning environment requires some degree of accuracy in classifying the learner's affective states. Detecting affective states with reasonable accuracy is an essential challenge for achieving functional affect-sensitive ITS [5].

There has been some research on learners' affect recognition from facial expression, speech, posture and dialog [4, 6]. A study by Arroyo et al. [7] explored how students' experience with tutoring systems shape their feelings and proposed a data-driven model for emotion using four sensors (camera, mouse, chair, and wrist). Physiological signal analysis is another possible approach to affect detection, and the focus of this paper. Here, heart rate, respiration, muscle activity, galvanic skin response, skin temperature, blood pressure etc might be suitable channels for recognizing affective states provided appropriate pattern recognition techniques are utilized. There is some evidence that some of these physiological signals correlate with the "basic emotions" such as anger, sadness, and disgust [5]. Unfortunately, these basic emotions are not very prominent in learning situations, at least for the short learning sessions with ITSs [8], where the learning-centered emotions listed above play a more prominent role. Challenges emerge during the process of collecting physiological data in learning interactions. Sensors for measuring physiological signals are often unsuitable for learning environments as they tend to interfere with learning activities. Due to these challenges, affect recognition with physiological signals is quite rare in educational settings (exception includes [9]). It is important to note that recent advances in wearable physiological sensors circumvents some of these practical challenges and create new opportunities to infer learner affect from physiology. In this paper we revisit the physiological-based learning-centered affect detection problem by using machine learning techniques to classify affective states from learners' physiological patterns (heart activity, skin response, respiration, facial muscle activity) during learning sessions with AutoTutor, an ITS with conversational dialogues [10].

It is important to emphasize two points before proceeding with a description of our Methods and Results. First, although several theories of emotion focus on *categorical* models, which consider discrete emotions such as fear, anger, etc, the concept, the value, and even the existence of such 'labeled' states is still a matter of considerable debate. Others have proposed *dimensional* models, where a person's affective states are represented as a point in a multi-dimensional space such as a valence-arousal space (see [11] for a discussion). Russell and Barrett [11] proposed a theory that somewhat unites these two views. According to this theory, physiological features are not necessarily correlated with specific emotional states (discrete or categorical emotions), but instead to the underlying dimensions of these states. For example, there is some evidence that valence correlates positively with heart rate while arousal correlates positively with skin conductance level [12]. Perhaps the most defensible position is to adopt a model that incorporates both perspectives by mapping discrete emotions on a valence/arousal space. However, while such a mapping has been proposed for the basic emotions [11], no such empirically grounded mapping exists for the learning-centered emotions. One model has been proposed by Kort, Reilly, and Picard [13], however, this model has yet to be supported with empirical data. Consequently, one of the aims of this study is to provide an empirically grounded mapping of a set of discrete learning-centered affective states into a valence/arousal space. This was achieved by asking learners to provide self-reports of affect based on both categorical and dimensional (valence/arousal) models.

Second, the present focus is on detecting naturally occurring affective states. This is an important point because many physiological-based affect detection systems have relied on artificially-induced emotions using different affect elicitation methods (e.g. photos, films, music, self imagining) [14, 15]. People express their emotions in variable ways, and the same emotion can be expressed differently in different situations. This raises the question of whether physiological-based affect detection will be equally effective in naturalistic contexts. We addressed this question by providing a comparison of the classification performance of affect detection from physiological data for two scenarios: (a) induced emotions via IAPS (International Affective Picture System) [16] and (b) emotions that naturally arise during interactions with AutoTutor.

2 Method

2.1 Participants, Materials and Procedures

Participants were 20 healthy volunteers from the University of Sydney. Participants' age ranged from 18 to 30 years and there were 8 males and 12 females. Participants were instructed not to take any drugs and to avoid caffeine consumption prior to the experiment. Participants signed an informed consent prior to the experiment. The experiment took approximately two hours and participants were rewarded with $20 book vouchers for their participation.

Participants were equipped with physiological sensors that monitored electrocardiogram (ECG), facial electromyogram (EMG), respiration, and galvanic skin response (GSR). The physiological signals were acquired using a BIOPAC MP150 system with AcqKnowledge software at 1000 samples per second for all channels. ECG was collected with two electrodes placed on the wrists. Two channels of EMG were recorded from the zygomatic and corrugator muscles respectively. A respiration band was strapped around the chest and GSR was recorded from the index and middle finger of the left hand.

The experiment consisted of two parts. The first part involved a 40 min recording of physiological signals while participants viewed emotionally charged photos from the IAPS collection [16]. A total number of 90 images (three blocks of 30 images each) for 10 seconds each were presented, followed by 6 seconds pauses between the images. The images were selected so that the IAPS valence and arousal scores for the stimuli spanned a 3×3 valence/arousal space (IAPS normed ratings). Participants also self-reported their emotions by clicking radio buttons on the appropriate location of 3×3 valence/arousal grid after viewing each image [17].

In the second part of the experiment, subjects completed a 20-minute tutorial session with AutoTutor on topics in computer literacy. AutoTutor is a dialogue based ITS for Newtonian physics, computer literacy, and critical thinking. AutoTutor's dialogues are organized around difficult questions and problems (called main questions) that require reasoning and explanations in the answers [10]. During this interaction, a video of the participant's face and a video of the computer screen were recorded. Participants made affect judgments (video annotation) immediately after the learning session at 10 seconds fixed intervals over the course of viewing their face and screen videos [6]. They were asked to provide two types of judgments: (a) categorical judgments which included eight learning-centered affective states (frustration, confusion, flow/engagement, delight, surprise, boredom, curiosity, and neutral) [6, 9] and (b) dimensional judgments consisting of valence/arousal (low, medium, high) ratings using the 3×3 grid described earlier.

2.2 Computational Models for Affect Detection

The Augsburg Matlab toolbox [18] for physiological signal processing was used for extracting statistical features. Video annotations were synchronized with the physiological signals and features were extracted using a 10 seconds window. The feature vectors were also labeled with the corresponding video annotations (1-3 degrees of valence/arousal). A total of 214 features were extracted from the five physiological signals and were merged to achieve feature-level fusion. Some features were common for all signals (e.g. mean, median, and standard deviation, range, ratio, minimum, and maximum) and others were related to their characteristics (e.g. heart rate variability, respiration pulse, frequency). The detailed description of the features can be found in [18]. To reduce the dimensionality of the large number of features, chi-square (X^2) feature selection was used for ranking the ten best features. The X^2 feature selection technique evaluates features by computing the value of the chi-squared statistic with respect to the class, in this case affective states.

The Waikato Environment for Knowledge Analysis (Weka), a data mining package [19], was used for classification. We selected three machine learning algorithms; k-nearest neighbor (KNN), linear support vector machine (SVM), and decision trees for classification Finally, a Vote classifier for combining classifiers was applied with the *average probability* rule [20]. The training and testing for both IAPS dataset and AutoTutor dataset was performed separately with a 10-fold cross validation. The kappa statistic was used as the overall classification performance metric and the F-measure (from precision and recall) was calculated as an indication of how well each affective state was classified. For the classification scores of precision (P) and recall (R), the F-measure (F1) is calculated by; $F1=2((P*R)/(P+R))$.

3 Results and Discussion

3.1 Discrete Emotions Mapping onto the Dimensional Valence/Arousal Plane

The key self-reported states were *neutral* (20%), *boredom* (21%), *confusion* (15%), *flow/engagement* (14%), *curiosity* (10%), and *frustration* (14%), whereas *surprise* (2%), *delight* (4%) were comparatively rare. Mapping of the discrete affective states onto the dimensional (valence/arousal) plane was performed by computing the mean valence and arousal (across 20 participants) associated with each emotion and projecting these on the valence/arousal space. The mapping is presented in Figure 1. It should be noted that a small translation procedure was adopted so that *neutral* was mapped onto the origin.

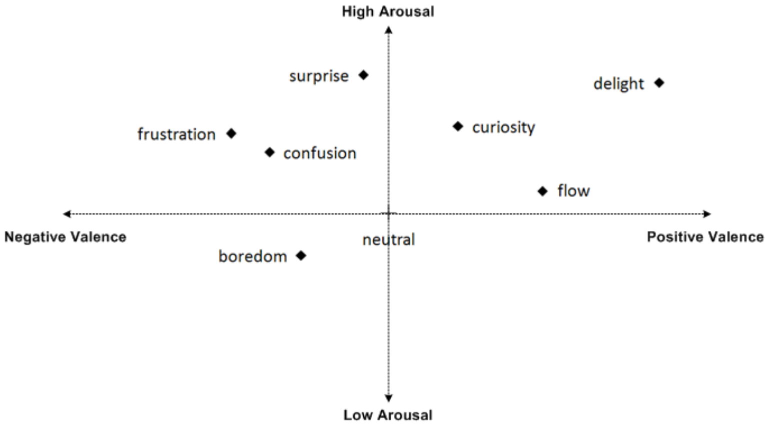


Fig. 1. Mapping of the discrete emotion labels on the valence/arousal plane (horizontal & vertical axes representing dimensions for valence and arousal respectively)

As Figure 1 indicates, *surprise* has no notable valence but has the highest arousal. In contrast, *flow/engagement* has arousal levels similar to neutral but is positively valenced. *Delight* and *curiosity* are characterized by high arousal and valence (especially delight). Both *confusion* and *frustration* have high arousal and negative valence. As could be expected, *boredom* is also negative valence with lower arousal. Most previous studies [e.g. 1, 6, 10] only used discrete affective states to annotate ITS interactions. Our mapping of discrete affective states onto a dimensional model (based on the empirical data) is a novel approach to combining results for the two models.

3.2 Classification Results from Physiological Signals

In this section we present the classification results for detecting 1-3 degrees (low, medium, high) of valence and arousal from physiological features, and leave classification of discrete emotions as part of future work. Self reports normally produce highly skewed class distribution, therefore up sampling and down sampling techniques are commonly used. For the initial analysis presented in this paper, we selected datasets/subjects with approximately balanced distribution of classes without using any up/down sampling techniques. Finally, the classes with extremely low or high number of instances were removed at the subject level. Separate classification analyses were performed for the valence and arousal dimensions. Table 1 presents the mean and standard deviation of kappa scores across learners for detecting 1-3 degrees of valance and arousal from physiological features (for both IAPS and AutoTutor sessions).

We note that the overall performance (kappa scores) of affect detection using IAPS is higher than performance during the AutoTutor interaction. This is expected because the IAPS is designed to elicit basic emotions of higher intensity than the learning emotions obtained over the course of the AutoTutor sessions. Despite the lower

Table 1. Mean (M) and standard deviation (SD) of kappa scores for detecting 1-3 degrees of valance and arousal from physioligical signals across learners

Affect	**IAPS**		**AutoTutor**	
	M	*SD*	*M*	*SD*
Valence	0.49	0.27	0.35	0.22
Arousal	0.31	0.16	0.23	0.03

overall performance, however, kappa scores are clearly greater than chance (kappa = 0) for the naturalistic emotions. In a previous study by D'Mello & Graesser [6] kappa score of 0.29 was achieved from face, dialog and posture using a similar AutoTutor setup. This indicates that learners' valence and arousal can be detected from physiological signals and the performance is quite satisfactory even when compared to controlled emotion elicitation.

While the kappa score provides a measurement for the overall performance, the F-measure indicates how well the individual affective categories were classified. Figure 2 presents the mean and standard deviation of the F-measure for detecting 1-3 degrees of valance and arousal from physiological signals across learners for both IAPS and AutoTutor sessions.

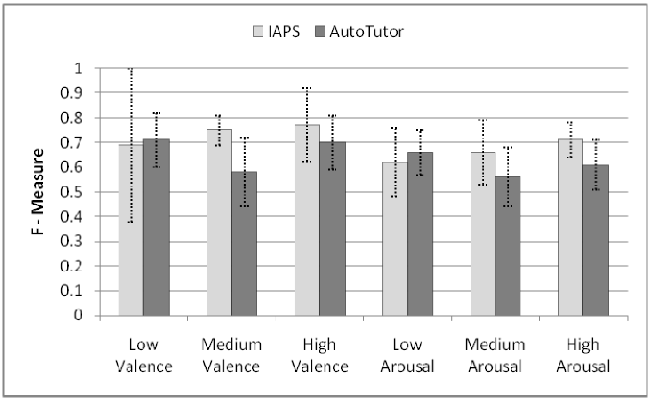


Fig. 2. Mean and standard deviation of the F-measure for detecting 1-3 degrees of valance and arousal from physiological signals across learners for both IAPS and AutoTutor sessions

Observing the results from Figure 2 separately for IAPS and AutoTutor; the performance (F-measure) of detecting the degrees of valance and arousal for IAPS increases from low to high for both valance and arousal. On the contrary, during AutoTutor sessions, a curvilinear relationship was observed. Highest performances occur for low valence and low arousal and also for high valence and high arousal. Performance for medium valence and medium arousal is in between these two extremes. While comparing results for IAPS and AutoTutor, we note that the performance of detecting low valence and low arousal from physiology during naturalistic interactions is comparable to controlled emotion elicitation. The performance of detecting medium and high valence/arousal is also quite satisfactory. A paired t-test for comparing the F-measure means for the six categories of IAPS (*M*= .70) and AutoTutor (*M= .64*) revealed no significant difference (*p >0.05*), which indicates that the accuracy of detecting affective states were not very different for the two models. As part of future work, this could be very suitable for creating a model where the classifier can be trained using the IAPS dataset and tested for the AutoTutor interactions.

4 Conclusion

The implementation of an adaptive, multimodal, robust affective sensitive ITS with sufficient reliability is still far from reality. Despite the challenges of affect recognition from physiological signals, this research presents an automatic affect classifier to detect learners' affective states from multichannel physiological signals with the support of a systematic experimental setup, feature selection techniques, and machine learning approaches. Results show that for the AutoTutor interaction, valence and arousal can be classified with moderate accuracy from multichannel physiology. Other modalities such as facial expressions, dialog and posture features [6] can be included along with physiological channels which may improve the performance of affect detection during ITS interactions. Classification of descrete affective states and finding their relationships with the dimensional model using multichannel physiology will be explored in the future.

Acknowledgments. M. S. Hussain was supported by Endeavour Awards and National ICT Australia (NICTA). Sidney D'Mello was supported by the National Science Foundation (ITR 0325428, HCC 0834847). Any opinions, findings and conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of NSF.

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. Learning, Media and Technology 29, 241–250 (2004)
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. Calvo, R.A., D'Mello, S.: New perspectives on affect and learning technologies. Springer, New York (in preparation)
4. Kapoor, A., Picard, R.W.: Multimodal affect recognition in learning environments. In: Proceedings of the 13th Annual ACM International Conference on Multimedia, Hilton, Singapore, pp. 677–682 (2005)
5. Calvo, R.A., D'Mello, S.: Affect Detection: An Interdisciplinary Review of Models, Methods, and their Applications. IEEE Transactions on Affective Computing 1, 18–37 (2010)

6. D'Mello, S., Graesser, A.: Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. User Modeling and User-Adapted Interaction 20, 147–187 (2010)
7. Arroyo, I., Cooper, D., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion Sensors Go To School. In: Proceeding of the 2009 Conference on Artificial Intelligence in Education, Amsterdam, vol. 200, pp. 17–24 (2009)
8. Lehman, B., Matthews, M., D'Mello, S.K., 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)
9. Aghaei Pour, P., Hussain, M., AlZoubi, O., D'Mello, S., Calvo, R.: The Impact of System Feedback on Learners' Affective and Physiological States. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 264–273. Springer, Heidelberg (2010)
10. Graesser, A.C., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. IEEE Transactions on Education 48, 612–618 (2005)
11. Russell, J.A., Barrett, L.F.: Core affect, prototypical emotional episodes, and other things called emotion: Dissecting the elephant. Journal of Personality and Social Psychology 76, 805–819 (1999)
12. Lichtenstein, A., Oehme, A., Kupschick, S., Jürgensohn, T.: Comparing Two Emotion Models for Deriving Affective States from Physiological Data. In: Peter, C., Beale, R. (eds.) Affect and Emotion in Human-Computer Interaction. LNCS, vol. 4868, pp. 35–50. Springer, Heidelberg (2008)
13. Kort, B., Reilly, R., Picard, R.W.: An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion. In: IEEE International Conference on Advanced Learning Technologies, Madison, Wisconsin, pp. 43–46 (2001)
14. Picard, R.W., Vyzas, E., Healey, J.: Toward machine emotional intelligence: analysis of affective physiological state. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 1175–1191 (2001)
15. Wagner, J., Kim, J., Andre, E.: From physiological signals to emotions: Implementing and comparing selected methods for feature extraction and classification. In: IEEE International Conference on Multimedia and Expo., ICME 2005, Amsterdam, The Netherlands, pp. 940–943 (2005)
16. Lang, P.J., Bradley, M.M., Cuthbert, B.N.: International affective picture system (IAPS): Technical manual and affective ratings. The Center for Research in Psychophysiology, University of Florida, Gainesville, FL (1995)
17. Russell, J.A.: A circumplex model of affect. Journal of Personality and Social Psychology 39, 1161–1178 (1980)
18. Wagner, J., Kim, J., Andre, E.: From physiological signals to emotions: Implementing and comparing selected methods for feature extraction and classification. In: IEEE International Conference on Multimedia and Expo. 2005, Amsterdam, The Netherlands, pp. 940–943 (2005)
19. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, San Francisco (2005)
20. Kuncheva, L.I.: Combining pattern classifiers: methods and algorithms. Wiley-Interscience, Hoboken (2004)

Short and Long Term Benefits of Enjoyment and Learning within a Serious Game

G. Tanner Jackson, Kyle B. Dempsey, and Danielle S. McNamara

Psychology Department, University of Memphis,
Memphis, TN 38152 USA
{gtjacksn,kdempsey,dsmcnamr}@memphis.edu

Abstract. Intelligent Tutoring Systems (ITSs) have been used for decades to teach students domain content or strategies. ITSs often struggle to maintain students' interest and sustain a productive practice environment over time. ITS designers have begun integrating game components as an attempt to engage learners and maintain motivation during prolonged interactions. Two studies were conducted to investigate enjoyment and performance at short-term (90 minutes) and long-term (3 weeks) timescales. The short-term study (n=34) found that students in a non-game practice condition performed significantly better and wrote more than the game-based practice. However, the long-term study (n=9) found that when students were in the game-based environment they produced longer contributions than when in the non-game version. Both studies revealed trends that the game-based system was slightly more enjoyable, though the differences were not significant. The different trends across studies indicate that games may contribute to an initial decrease in performance, but that students are able to close this gap over time.

Keywords: Serious Games, Intelligent Tutoring Systems, game-based learning.

1 Intelligent Tutoring and Games

Intelligent Tutoring Systems (ITSs) have been producing significant learning gains for decades; however, one common problem with these systems is maintaining student engagement throughout extended interactions. This problem is especially pertinent for skill-based tutors. Acquiring a new skill usually requires a significant commitment to continued practice and application. Skills are often developed and improved with practice over an extended period of time [1]. A few ITSs that focus on skill acquisition require interactions that last ~100 hours or integration within school curricula [2] Due to the long-term nature of these interactions, students often become disengaged and uninterested in using the systems [3]. To combat this problem, researchers have begun to incorporate game features within tutoring environments [4].

Well-designed games are appealing because they address affective states, motivation, and expectations of the player [5]. Progressing and succeeding within a serious game requires that the learner be involved, and to some extent, engaged in the game. When disengaged from the game, the learner runs the risk of losing the game. However, engagement is not guaranteed simply because game features are present.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 139–146, 2011.

For a serious game to be effective, the learner must want to continue using the system over time. It is also possible for entertaining game elements to hinder learning by distracting the learner from the intended learning task. Thus, one concern when integrating ITS and game components is how to effectively engage the learner in both the game and the learning elements, without distracting from the learning task.

2 Development of iSTART-ME

The Interactive Strategy Training for Active Reading and Thinking (iSTART) tutor is a web-based reading strategy trainer that provides young adolescent to college-aged students with reading strategy training to better understand challenging science texts [6]. iSTART is comprised of three modules: Introduction, Demonstration, and Practice. In the Introduction module, three animated agents engage in a vicarious dialogue to introduce the learner to the concept of self-explanation and each of the reading strategies. The Demonstration module includes two animated agents who generate and discuss the quality of example self-explanations and prompt the learner to identify which strategies may have been used within each example. The Practice module requires learners to generate their own self-explanations and an animated agent (Merlin) provides qualitative feedback on how to improve the self-explanation quality. An Extended Practice environment continues this generative practice over a longer time period and allows teachers to assign specific texts.

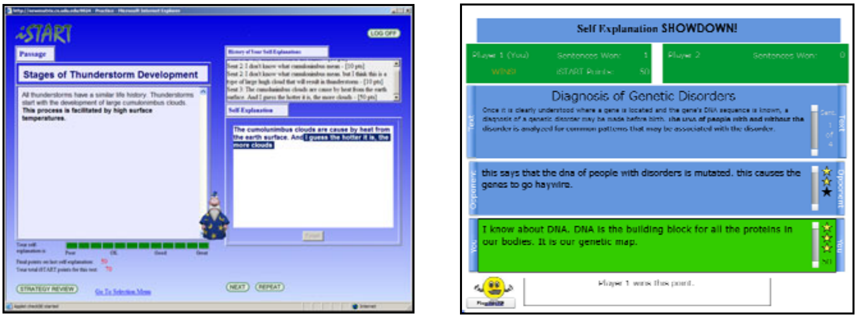
Students using iSTART have demonstrated significant improvement in reading comprehension [7]. While learners consistently make significant improvements by interacting with iSTART, skill mastery requires long-term interaction with repeated practice [8]. One unfortunate side effect of this long-term interaction is that students often become disengaged and uninterested in using the system [3]. Thus, iSTART-ME (Motivationally Enhanced) has been developed on top of the existing ITS and incorporates serious games and other game-based elements [9].

The iSTART-ME game-based environment builds upon the existing iSTART system. The main goal of the iSTART-ME project is to implement several game-based principles and features that are expected to support effective learning, increase motivation, and sustain engagement throughout a long-term interaction with an established ITS. The previous version of iSTART extended practice progressed students from one text to another with no intervening actions. The new version of iSTART-ME is controlled through a selection menu interface. Researchers claim that motivation and learning can be increased through multiple elements of a task including feedback, fantasy, personalization, choice, and curiosity [10, 11]. Therefore, these features have been incorporated into the design of the iSTART-ME selection menu. This selection menu interface provides students with opportunities to interact with new texts, earn points, advance through levels, purchase rewards, personalize a character, and play educational mini-games (designed to use the iSTART strategies).

2.1 Coached Practice and Showdown

Some of the iSTART-ME mini-games require students to practice generating their own self-explanations. All generative games present users with a bolded target sentence, the prior text, and an area to type in their own self-explanation. These generation games utilize the same natural language assessment algorithm [12] that has produced results comparable to humans [13]. The feedback from the algorithm is interpreted computationally and presented in various forms within the different environments (i.e., length and color of bar or number of stars).

Coached Practice. Coached Practice is a revised version of the original practice module within iSTART (see Figure 1 for screenshot). Learners are asked to generate their own self-explanation when presented with a text and specified target sentence. Students are guided through practice by Merlin, a wizard who provides qualitative feedback for user-generated self-explanations. Merlin reads sentences aloud to the participant, stops after reading a target sentence, and asks the participant to self-explain the bolded sentence. After the participant completes each self-explanation, Merlin provides feedback on the quality of the self-explanation using the automatic assessment algorithm. If the current contribution quality is low, students can try again and use Merlin's feedback to improve their current self-explanation. The only game-like elements within Coached Practice are a colored qualitative feedback bar (visually indicating: poor, fair, good, great) and points associated with each self-explanation.



Stages of Thunderstorm Development

All thunderstorms have a similar life history. Thunderstorms start with the development of large cumulonimbus clouds. **This process is facilitated by high surface temperatures.**

Sent 2: I don't know what cumulonimbus mean - [10 pts]
Sent 2: I don't know what cumulonimbus mean, but I think this is a type of large high cloud that will result in thunderstorms - [10 pts]
Sent 3: The cumulonimbus clouds are cause by heat from the earth surface. And I guess the hotter it is, the more clouds - [50 pts]

The cumolunimbus clouds are cause by heat from the earth surface. And I guess the hotter it is, the more clouds

Final points on last self explanation: 50
Your total iSTART points for this text: 70

Coached Practice

Player 1 (You) Sentences Won: 1
WINS! iSTART Points: 50
Player 2 Sentences Won: 0

Diagnosis of Genetic Disorders

Once it is clearly understood where a gene is located and the gene's DNA sequence is known, a diagnosis of a genetic disorder may be made before birth. **the DNA of people with and without the disorder is analyzed for common patterns that may be associated with the disorder.**

this says that the dna of people with disorders is mutated. this causes the genes to go haywire.

I know about DNA. DNA is the building block for all the proteins in our bodies. It is our genetic map.

50

Player 1 wins this point.

Showdown

Fig. 1. Screenshots of Coached Practice and Showdown

Showdown. Showdown is a game-based method of practice that requires students to generate their own self-explanation for a specified target sentence (see Figure 1 for screenshot). Participants compete against a computer player to win rounds by writing better self-explanations. Participants are guided through the game by text-based instructions. After the participant completes each self-explanation, the computer scores the self-explanation on a scale of 0–3 and displays the score as stars (using same algorithm as Coached Practice). The opponent's self-explanation is also presented and scored. The self-explanations for the virtual player are randomly drawn from a database of existing, pre-evaluated self-explanations. The self-explanation scores are compared and the player with the highest score wins the round. In case of a tie score, the player is given another target sentence worth two points instead of one. The player with the most points at the end of a text is declared the winner.

3 Current Studies

Two studies were conducted that included comparisons between Coached Practice and Showdown. The first was a short-term, 90 minute, between-subjects study that was designed to investigate the immediate effects of the two training environments on enjoyment and performance. The second was a small longer term (3 week) within-subjects study that allowed students to use both training environments multiple times. Analyses of performance and enjoyment were conducted for both studies. Performance measures included the number of words used by students within their self-explanations as well as their average self-explanation quality score as computed by the natural language assessment algorithm. Enjoyment measures included participants' responses from posttest survey questions.

3.1 Study 1: Short-Term Assessment

The main goal of the first study is to examine whether the inclusion of game elements have an immediate effect on students' enjoyment and performance. All students (n=34) completed a short demographics survey and were transitioned into an abbreviated version of iSTART training consisting of only the introduction modules. After the introductory lessons, students were randomly assigned to interact with either Coached Practice (n=18) or Showdown (n=16) for two texts. Text order was counterbalanced, and the same texts were used in both practice environments. At the end of the study, participants completed the Jennett et al., enjoyment and engagement questionnaire (likert scale 1-6, higher numbers indicating stronger agreement) [14].

Results. Several analyses were conducted to investigate performance and enjoyment differences between training conditions. An ANOVA including the between-subjects factor of condition revealed that students in Coached Practice (M=31.34, SD = 10.81) included significantly more words within their self-explanations than the students in Showdown (M=17.24, SD=6.53), F(1,32)=20.52, p<.001. An ANOVA also found that students within Coached Practice (M=2.50, SD=.34) wrote significantly higher quality self-explanations than the students in Showdown (M=1.91, SD=.46), F(1,32)=18.21, p<.001 (see Figure 2). These analyses indicated that students who interacted with Coached Practice wrote significantly longer and higher quality self-explanations (SEs) than those students who played Showdown.

Analyses were also conducted to investigate differences on two of the posttest enjoyment questions. The trends indicate more enjoyment and an increased likelihood of return uses for Showdown, however, these differences were not significant, F(1,32)=.362, p=.552 ,and F(1,32)=1.14, p=.294, respectively. It appears that students within Coached Practice performed better during training than the students within Showdown. However, Coached Practice includes more fine-grained formative feedback and it allows for students to try again for low quality self-explanations.

In addition, a linear regression was conducted to predict overall self-explanation scores. The predictors for this analyses included experimental condition and ratings from enjoyment and reuse. The average number of words was omitted from the regression because it is one of the measures included in the self-explanation scoring

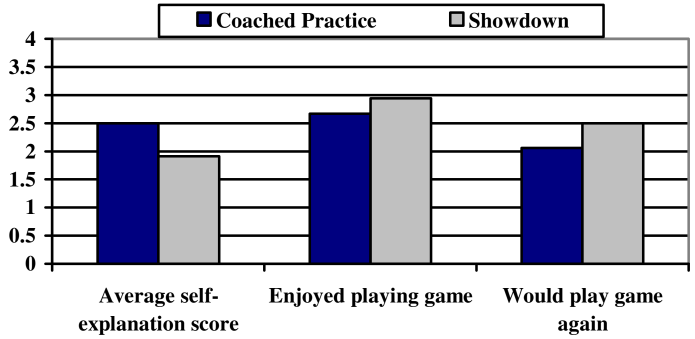


Fig. 2. Study 1 means for performance and enjoyment

algorithm. The regression produced a significant model, F(3,30)=7.21, R^2=.419 *p*<.001, with experimental condition accounting for 30.8 percent of the variance from the overall model. Unfortunately the contributions of the enjoyment and reuse ratings were not significant. This result indicates that the training environment significantly contributes to the self-explanation quality over and above system enjoyment.

3.2 Study 2: Long-Term Assessment

Previous research with iSTART-ME has focused on short-term studies that investigated individual elements within the system. The current study deviates from this precedent and includes a smaller number of participants that interacted with the full iSTART-ME system across multiple sessions spanning several weeks. All participants (n=9) completed the full iSTART-ME training, including Introduction, Demonstration, Practice, and the Selection Menu. After completing the initial training and Practice module, students spent the remainder of the sessions freely using all features within the Selection Menu (all practice methods and mini-games). After interacting with iSTART-ME for 7 sessions, participants completed a posttest survey, which included questions about attitudes, enjoyment, and motivation.

Results. The current analyses focus on the student interactions with Coached Practice and Showdown during these sessions, as well as the corresponding enjoyment questions from the posttest. In contrast to the results from Study 1, a within-subjects ANOVA comparing self-explanation length as a function of training environments indicated that students in this study wrote significantly longer self-explanations within Showdown (M=27.44, SD=9.65) than during Coached Practice (M=14.46, SD=4.19), F(1,8)=27.09, *p*<.001. Also, the differences in self-explanation quality found in the short-term study were not present during a long-term interaction and another within-subjects ANOVA that compared self-explanation quality found no significant differences in self-explanation quality between Coached Practice (M=2.55, SD=.56) and Showdown (M=2.39, SD=.70), F(1,7)=.627, *p*=454 (see Figure 3). Similar to the results from Study 1, the trends suggest that students prefer Showdown, but no significant differences were found between the ratings for system enjoyment and return use, F(1,8)=.813, *p*=.393, and , F(1,8)=.667, *p*=.438, respectively.

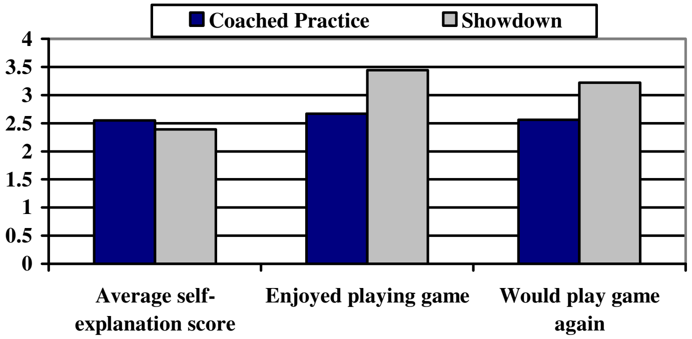


Fig. 3. Study 2 means for performance and enjoyment

4 Conclusions

Considering the results from both these two studies we can see different outcomes that depend on the duration of training and interactions. The first study serves as a strictly controlled short-term comparison between the two training environments (Coached Practice and Showdown) in which the students interacted with only two texts during a 90-minute session. By contrast, the second study provides a more ecologically valid long-term investigation of how students may benefit from these environments over time. The results from Study 1 suggest that the game environment may initially detract from the interaction and inhibit potential learning. These lower scores could be due to differences in pedagogy (feedback vs. modeling), as well as specific game-based factors, including extra cognitive effort spent to understand the rules and methods to win the game, added pressure to perform in a competitive environment, and attempts to game the system rather than complete the intended task. The equivalent performance outcomes in Study 2 tentatively suggest that this deficit may lessen over time as the students become more familiar with the target skill and the various aspects of the system. Indeed, the game-based aspects of iSTART-ME were intended for just this purpose, when the students were interacting with the system over long periods of time.

One of the most obvious and biggest limitations of the current work is the small sample size for Study 2. Previous work with iSTART-ME has taken the traditional empirical approach and used larger samples that focused on the immediate effects of game-based learning. However, long-term studies offer significantly more interactions per participant, and provide insight into effects that may develop differently over time (as is the case here). Additionally it is often impractical, and expensive, to conduct long-term laboratory studies that span multiple weeks with large samples. Therefore, the second study was conducted on a smaller sample and served as an ecological investigation to explore how students would use and benefit from the newly developed game-based system. The lower number of participants in Study 2 precluded conducting a linear regression, as we did for Study 1. However, the other analyses suggest that a long-term assessment is essential to building a more complete picture of how enjoyment and learning are affected differently across time.

The results of these two studies suggest interesting trends supported by previous research. In both studies, the game-based method of practice tended to receive higher enjoyment ratings than the non-game environment (supported by [10,11]). Previous research has proposed that games are more engaging and potentially 'could' lead to better or more sustained learning [15,16]. Additionally, previous work has shown that there is little research comparing the effectiveness of gaming environments to more traditional ITS environments [5]. Therefore, despite its limitations, the current work contributes to the growing body of research with serious games.

The results from Study 2 offer an encouraging counterpart to those from Study 1. The latter results support the assumption that game-based elements (including choice and control) could sustain learners' attention and keep them interested in the system long enough to help them improve skills over time. The results from Study 2 are also encouraging because they support the current design of iSTART-ME and indicate that the duration of assessment can have a large impact on the overall outcomes.

This combination of results that investigate changes across longer time spans has not been adequately explored within previous research. The current work offers a unique contribution to the field of serious games, and should hopefully encourage other researchers to investigate various timelines within their own learning technologies. This further suggests that previous research using short-term game-based interventions may provide an incomplete picture of the implications for system design and implementation. The studies presented here offer a starting point for future work that more fully investigates the timelines of effects for various game elements and how they impact learning, motivation, and engagement.

Acknowledgments. This research was supported in part by the National Science Foundation (IIS-0735682). 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.

References

1. Newell, A., Rosenbloom, P.: Mechanisms of skill acquisition and the law of practice. In: Anderson, J.R. (ed.) Cognitive Skills and their Acqusition, pp. 1–55. Hillsdale, NJ (1981)
2. 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, pp. 61–78. Cambridge University Press, Cambridge (2006)
3. Bell, C., McNamara, D.S.: Integrating iSTART into a high school curriculum. In: Proceedings of the 29th Annual Meeting of the Cognitive Science Society, Cognitive Science Society, Austin (2007)
4. McNamara, D.S., Jackson, G.T., Graesser, A.C.: Intelligent tutoring and games (ITaG). In: Baek, Y.K. (ed.) Gaming for Classroom-Based Learning: Digital Role-Playing as a Motivator of Study. IGI Global (2010)
5. O'Neil, H.F., Wainess, R., Baker, E.L.: Classification of learning outcomes: Evidence from the computer games literature. Curriculum Journal 16, 455–474 (2005)
6. McNamara, D.S., Levinstein, I.B., Boonthum, C.: iSTART: Interactive strategy trainer for active reading and thinking. Behavioral Research Methods, Instruments, & Computers 36, 222–233 (2004)

7. Magliano, J.P., Todaro, S., Millis, 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 (2005)
8. Jackson, G.T., Boonthum, C., McNamara, D.S.: The efficacy of iSTART extended practice: Low ability students catch up. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 349–351. Springer, Heidelberg (2010)
9. Jackson, G.T., Dempsey, K.B., McNamara, D.S.: The evolution of an automated reading strategy tutor: From classroom to a game-enhanced automated system. In: Khine, M.S., Saleh, I.M. (eds.) New Science of Learning: Cognition, Computers and Collaboration in Education, pp. 283–306. Springer, New York (2010)
10. Cordova, D.I., Lepper, M.R.: Intrinsic motivation and the process of learning: Beneficial effects of contextualization, personalization, and choice. J. Ed. Psyc. 88, 715–730 (1996)
11. Papastergiou, M.: Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. Comput. Educ. 52, 1–12 (2009)
12. 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)
13. Jackson, G.T., Guess, R.H., McNamara, D.S.: Assessing cognitively complex strategy use in an untrained domain. Topics in Cognitive Science 2, 127–137 (2010)
14. Jennett, C., 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, 641–661 (2008)
15. Garris, R., Ahlers, R., Driskell, J.E.: Games, motivation, and learning: A research and practice model. Simulation & Gaming 33, 441–467 (2002)
16. Gee, J.P.: What video games have to teach us about learning and literacy. Palgrave MacMillian, New York (2003)

Error-Flagging Support and Higher Test Scores

Amruth N. Kumar

Ramapo College of New Jersey,
Mahwah, NJ 07430, USA
amruth@ramapo.edu

Abstract. Previously, providing error-flagging support during tests was reported to lead to higher scores. A follow-up controlled study was conducted to examine why, using partial crossover design. Two adaptive tutors were used in fall 2009 and spring 2010, and the data collected during their pre-test stage was analyzed. The findings are: (1) When a student solves a problem correctly on the first attempt, error-flagging support helps the student move on to the next problem more quickly without pausing to reconsider the answer. But, it may also encourage students to use error-flagging as an expedient substitute for their own judgment; (2) Given error-flagging support, many more students will arrive at the correct answer by revising their answer, which explains why students score higher with error-flagging; (3) Students will use error-flagging to reach the correct answer through trial and error even though the problems are not of multiple-choice nature. However, at least some students may engage in informed (as opposed to brute-force) trial and error. (4) Error-flagging support provided during tests could cost students time. (5) Given how often students move on after solving a problem incorrectly, without ever reconsidering their answer, providing error-flagging support during testing is still desirable.

Keywords: Error-flagging, Testing, Adaptation, Evaluation.

1 Introduction and Experiment

Studies on the effect of providing error-flagging feedback during testing have yielded mixed results. Multiple studies of paper-and-pencil testing have reported lower performance due to increased anxiety (e.g., [3, 5]) or no difference (e.g., [9]) when feedback about the correctness of answers was provided. Studies with early Computer Assisted Instruction/Testing showed better performance with such feedback during testing than without (e.g., [2, 10]). Later studies with computer-based multiple-choice testing showed no relative advantage or performance gain from providing such feedback [8, 9]. In a recent study, researchers found that there was little difference among the types of feedback provided during testing with the ACT Programming Tutor [4]. In a more recent study of online tests that do not involve multiple-choice questions [6], we found that students scored better on tests with rather than without error-flagging support. We conducted a follow-up study to find out why they scored better – a question of interest since we use online pre-tests to prime the student model used by our adaptive tutors [7].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 147–154, 2011.

In fall 2009 and spring 2010, we used two problem-solving software tutors for the study. The tutors were on functions, an advanced programming concept. One tutor dealt with debugging, and the other, with predicting the behavior of programs with functions. Debugging tutor targeted 9 concepts; Behavior tutor targeted 10 concepts. The tutors presented problems on these concepts, each problem containing a program which had to be debugged or whose output had to be determined by the student. Each software tutor went through pre-test-practice-post-test protocol as follows:

- It first administered a pre-test to evaluate the prior knowledge of students and build the student model. The pre-test consisted of one problem per concept – 9 problems in debugging tutor and 10 problems in behavior tutor.
- Subsequently, it provided practice problems on only those concepts on which students had solved problems incorrectly during pre-test [7];
- Finally, it administered post-test problems on only those concepts on which students had solved sufficient number of problems during practice as indicated by the student model.

The three stages were administered online, back-to-back without any break in between. The software tutors allowed 30 minutes for the three stages combined. *Since we wanted to study the effect of error-flagging on tests, data from only the pre-test portion of the tutor was considered for analysis.*

The evaluations were *in-vivo*. The tutors were used in introductory programming courses at 12 institutions which were randomly assigned to one of two groups: A or B. Subjects, i.e., students accessed the tutors over the web, typically, after class. The tutors remotely collected the data for analysis.

A partial cross-over design was used: students in group A served as control subjects on debugging tutor and test subjects on behavior tutor, while students in group B served as test subjects on debugging tutor and control subjects on behavior tutor. All else being equal, error-flagging feedback was provided during pre-test to students in the test group, but not the control group. Error-flagging, i.e., error-detection, but not error-correction support was provided before the student submitted the answer.

Debugging tutor: In order to identify a bug, the student had to select the line of code which had the bug, the programming object on that line to which the bug applied, and finally, the specific bug that applied to the programming object on the line. For example, the student would select line 8, the variable `count` on line 8, and the bug that `count` was being referenced before it was assigned a value. After the student identified all three, the summary of the bug would appear in the panel that displayed the student's answer. After the summary was displayed, students had the option of deleting the entire bug and starting over, whether or not error-flagging support was provided. In addition, students had the option to click a button that said that the code had no bugs. This button was presented only when the student had selected no bugs.

Behavior tutor: Students identified the output of the program, one step at a time, e.g., if the program printed 5 on line 9, followed by 9 on line 13, students had to enter this answer in two steps. In each step, they entered the output free-hand, and selected the line of code from a drop-down menu. For each step, a button was provided for students to delete it if they so wished. They also had the option to change the output or the line number *in-situ,* without deleting the entire step. In addition, students had the option to click a button that said that the code had no output. This button was presented only when the student had not yet identified any output for the program.

When error-flagging feedback was provided, if an answer was incorrect, it was displayed on red background if incorrect, and green background if correct. When error-flagging support was not provided, the step was always displayed on white background. When error-flagging support was provided, no facility was provided for the student to find out why it (bug or output step) was incorrect, or how it could be corrected. The online instructions presented to the students before using each tutor explained the significance of the background colors. Whether or not the tutor provided error-flagging feedback, students had the option to revise their answer as often as necessary before submitting it. Once again, the instructions presented to the students before using each tutor explained the user interface facilities provided for revising an answer.

On a multiple-choice test question, with error-flagging support, a student could repeatedly guess until arriving at the correct answer. Given n choices in the question, the student would need no more than n guesses. In debugging tutor, though, the number of choices was more than 20 on each problem, and the choices were not arranged as a flat list, but as a hierarchy of selections: line, object and type of bug being the three levels of hierarchy. In behavior tutor, although there were limited choices for the line number, the output itself was entered free-hand, making the number of choices infinite. So, neither debugging tutor nor behavior tutor presented problems that could be considered multiple-choice, and therefore, susceptible to gaming when error-flagging feedback was provided.

2 Results

For analysis, only those students were considered who had used both debugging tutor and behavior tutor. Only those students were considered who attempted most of the pre-test problems: at least 6 of the 9 problems on debugging tutor and 8 of the 10 problems on behavior tutor. Students who scored 0 or 100% on either pre-test were excluded. This left 40 students in Group A and 59 students in Group B. In order to factor out the effect of the difference in the number of problems solved by students, the average score per pre-test problem was considered for analysis, which can range from 0 through 1, rather than total score.

Score Per Problem: A 2 X 2 mixed-factor ANOVA analysis of the score per pre-test problem was conducted with the topic (debugging versus behavior) as the repeated measure and the group (group A with error-flagging on behavior versus group B with error-flagging on debugging pre-test) as the between subjects factor.

A significant main effect was found for error-flagging [F(1,97) = 44.107, $p < 0.001$]: students scored 0.519 ± 0.048 without error-flagging and 0.689 ± 0.035 with error-flagging (at 95% confidence level). The difference was statistically significant [t(98) = -3.069, $p = 0.003$]. The effect size (Cohen's d) is 0.46, indicating medium effect. *Students scored more with error-flagging support during the test than without.*

The between-subjects effect for group (A versus B) was not significant [F(1,97) = 2.340, p = 0.129], indicating that the two groups were comparable, whether they got error-flagging support on debugging pre-test or on behavior pre-test.

A large significant interaction was found between treatment and group [F(1,97) = 123.022, p < 0.001]. As shown in Table 1, the group with error-flagging scored statistically significantly more than the group without error-flagging on both debugging pre-test [t(97) = -2.638, p = 0.01] and behavior pre-test [t(97) = 5.604, p < 0.001]. It turned out that students found debugging pre-test to be harder than behavior pre-test, scoring significantly less on it (average 0.4736) than on behavior pre-test (average 0.7241) [t(98) = -8.336, p < 0.001]. This explains why group B scored less with error-flagging on debugging pre-test (0.521) than without error-flagging on behavior pre-test (0.635).

Table 1. Average Pre-test Score with and Without Error-Flagging

	Debugging pre-test	Behavior pre-test
Without Error-Flagging	0.403 ± 0.074	0.635 ± 0.061
With Error-Flagging	0.521 ± 0.045	0.856 ± 0.054

In order to answer why error-flagging support led to better scores, we considered **four cases**:

1. Students solved a problem correctly without any revisions – did students with error-flagging support solve them faster, because they were not tempted to reconsider their answer?
2. Students solved a problem incorrectly without any revisions – since students in the experimental group did not take advantage of error-flagging feedback, the two groups should be comparable in how quickly they solved problems.
3. Students solved a problem correctly with revisions – did students with error-flagging support take longer to solve the problem? Did they revise more often?
4. Students solved a problem incorrectly with revisions – if this category applied to students with error-flagging support, it would suggest that error-flagging support is not a substitute for knowing the correct answer.

On Debugging tutor, students had only one mechanism to revise their answer once they got error-flagging feedback: delete the entire bug. On Behavior tutor, students had two mechanisms to revise their answer: either delete the entire step, or edit the step *in-situ*. Since *in-situ* editing events were only collected in spring 2010 and not in fall 2009, revision data for Behavior tutor was incomplete and was dropped from further analysis.

Case 1: When solving problems correctly without revision, students with error-flagging support solved them faster (53.86 seconds) than those without (70.61 seconds), and this difference was significant [t(231) = 2.057, p = 0.041]. This supports the hypothesis of our first case, that *given the positive reinforcement that an answer is correct, students with error-flagging feedback move on to the next problem more quickly without pausing to reconsider their answer.* Table 2, where data of control group is shown as "No" and experimental group with error-flagging as "EF", shows this to be the case for all but two problems (4 and 7). The difference on problems 5 and 6 were marginally significant (p = 0.09), whereas the rest of the differences were not statistically significant.

Table 2. Time spent per problem when solving the problem correctly without revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	81.23	74.73	75.31	60.0	94.69	128.0	40.80	55.40	53.94
EF (N=59)	64.33	63.69	46.29	70.0	52.19	63.40	53.35	42.54	45.20

However, the news is not all positive. As shown in Table 3, on every problem, the percentage of students who solved problems correctly without revision was smaller with error-flagging than without. Given that the two groups were comparable, this could suggest that *when error-flagging feedback is provided, students may be using it as a crutch, as an expedient replacement for their own judgment.* In other words, in at least some cases, they arrived at the correct answer through revisions even though they could have do so with additional deliberation instead of revisions - they resorted to revising their answer even when they did not need to, just because they could.

Table 3. Percentage of students who correctly solved the problem without revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	32.50	37.50	40.0	30.0	32.50	25.0	37.50	50.0	42.50
EF (N=59)	20.34	22.03	11.86	6.78	27.12	8.47	28.81	22.03	25.42

Case 2: In contrast, when solving problems incorrectly without revision, there was no significant difference in the time taken by students with (80.5 seconds) or without (75.37 seconds) error-flagging support [t(265) = -0.636, p = 0.525]. This supports our second case – since students in the experimental group did not take advantage of error-flagging feedback even when their answer was incorrect, the two groups should be comparable in how quickly they solved problems. Table 4 shows the average time taken by the two groups to solve each problem when they solved it incorrectly without revision. The difference between the two groups was statistically significant only on problem 9.

Table 4. Time spent per problem when solving the problem incorrectly without revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	92.52	73.0	77.56	78.59	64.08	93.17	74.12	50.65	67.06
EF (N=59)	86.63	98.85	59.57	60.33	82.30	98.0	95.60	59.50	46.0

Table 5 shows that when error-flagging support is provided, far smaller percentage of students solves a problem incorrectly without revising it. In other words, students take advantage of error-flagging to fix an incorrect answer. The percentage of students without error-flagging support who moved on after solving a problem incorrectly, but without revising their answer even once is rather large (40% - 60%).

Prompting such a large percentage of students to reconsider their answer is the goal of providing error-flagging feedback during tests. If a student knows the material, and solves the problem correctly, but enters the answer incorrectly, this would help the student uncover incidental or accidental mistakes. If a student knows the material, but did not solve the problem correctly, this would prompt the student to go over the steps of solving the problem again. If a student does not know the material, this would make the student aware of his/her lack of knowledge, which is also desirable.

Table 5. Percentage of students who incorrectly solved the problem without revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	52.50	50.0	40.0	55.0	60.0	60.0	42.50	42.50	40.0
EF (N=59)	13.56	33.90	23.73	10.17	16.95	23.73	8.47	13.56	8.47

Case 3: Table 6 lists the percentage of students who solved problems correctly after revising their answer at least once. As could be expected, the percentage of students who solved problems correctly by revising their answers was much greater with error-flagging than without. *This explains the significantly better score of students with error-flagging than without.*

Table 6. Percentage of students who correctly solved the problem with revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	5.0	2.50	5.0	2.50	0	0	2.50	2.50	0
EF (N=59)	25.42	22.03	16.95	40.68	45.76	16.95	18.64	30.51	38.98

Since so few students without error-flagging support actually revised their answers (at most 2), in the next analysis, we considered the time spent per problem and the number of revisions per problem of only those who got error-flagging feedback. Table 7 lists these as "Rev Time" and "Revisions" respectively. For comparison purposes, it also lists the average time spent per problem by the students who solved the problem correctly without any revision (with or without error-flagging) as "NoRevTime". Note that in order to revise and answer correctly, students with error-flagging support spent more time, often, twice as much time, than those who did not revise their answer. This difference was statistically significant: 104.6 seconds with revisions versus 63.3 seconds without revisions [t(390) = -6.11, p < 0.001].

Table 7. Students with error-flagging support who correctly solved a problem with revisions

Problem	1	2	3	4	5	6	7	8	9
N	15	13	10	24	27	10	11	18	23
NoRevTime	73.1	69.6	66.5	62.5	71.2	106.5	47.5	50.3	49.8
Rev Time	135.6	108.5	132.6	125.8	81.7	104.9	105.5	93.2	80.5
Revisions	4.27	5.62	7.40	8.63	4.22	13.0	6.0	4.17	4.04
Errors	28	41	50	55	23	45	55	41	31

As shown on "Revisions" row in Table 7, students on average revised their answer at least 4 times per problem. Each problem had only one correct answer (although this was not communicated to the students). So, an average of 4 revisions per problem indicates that *students used error-flagging support to reach the correct answer through trial and error*, which is clearly undesirable. The last row titled "Errors" lists the possible number of error options for each problem. Each problem contained 13-18 lines of code over which these error options were spread. So, while an average of 4 attempts to identify one bug is excessive, it represents less than 18% of the total number of possible error options for each problem, suggesting that *at least some students may have engaged in informed (as opposed to brute-force) trial and error.*

Case 4: Table 8 lists the percentage of students who solved problems incorrectly even after revising their answer at least once. Curiously, this percentage is much larger with error-flagging than without. This suggests that error-flagging support did not always help students arrive at the correct answer, and is not a substitute for knowing the answer at the outset.

Table 8. Percentage of students who incorrectly solved problems with revision

Problem	1	2	3	4	5	6	7	8	9
No (N=40)	5.0	7.50	5.0	7.50	7.50	10.0	12.50	2.50	0
EF (N=59)	23.73	15.25	33.90	40.68	10.17	47.46	37.29	22.03	8.47

For the follow-up analysis, once again, we excluded data of students without error-flagging since too few of them revised their answers (usually 2 or 3). As shown in Table 9, even after 3 or more revisions facilitated by error-flagging support, a large number of students solved problems incorrectly any way. The table lists the time spent per problem as "RevTime" and average number of revisions as "Revisions". For comparison purposes, we listed the time spent per problem by students who solved each problem incorrectly, but without any revisions, as "NoRevTime". We compared against this group because, if our experimental group was going to solve a problem incorrectly any way, we wanted to find out the time penalty, if any, of all the revisions prompted by error-flagging support. When solving problems incorrectly, students who revised took significantly more time per problem (127.72 seconds) than those who did not revise their answer (77.10 seconds) [t(429) = -7.414, $p < 0.001$]. As the table shows, even when the answer eventually turned out to be incorrect, students spent up to twice as long as the group that did not. Since all the additional time spent on these problems did not increase the students' score on the test, *error-flagging support provided during tests could cost students time* by encouraging fruitless speculation.

Once again, note that students revised their answer at least 3 times per problem. While it is common knowledge that students solve problems through trial and error if error-flagging support is provided on multiple-choice questions, the finding of this study is that they resort to trial and error even when the problem is not of multiple-choice nature. One mechanism to discourage or minimize excessive revisions might be to limit the number of revisions allowed per problem. This would prevent students from arriving at the correct answer through repeated trials, as well as reduce the time they spend speculating on problems for which they do not eventually find the correct answer.

Students spent less time with than without error-flagging in case 1, and more time in cases 3 and 4. Since there was no significant difference with versus without

Table 9. Students with error-flagging support who incorrectly solved a problem with revisions

Problem	1	2	3	4	5	6	7	8	9
N	14	9	20	24	6	28	22	13	5
NoRevTime	90.9	85.9	69.2	74.7	69.4	95.0	79.0	53.5	62.0
RevTime	109.6	123.4	137.7	109.1	116.8	158.3	137.9	103.2	61.6
Revisions	4.93	4.44	8.79	4.58	6.0	9.21	13.1	5.62	3.0

error-flagging in the overall time taken on either debug or behavior tutor pre-test, *error-flagging support led students to save time on the problems they knew how to solve and spend it attempting problems for which they did not readily know the solution.* This re-allocation of time is desirable in tests and tutors, which makes the case for providing error-flagging support. *But,* with error-flagging support, students often used trial and error to arrive at the correct solution, and spent significantly more time futilely revising their answers, neither of which is desirable. In order to address these concerns and further elucidate how error-flagging support can be beneficially used in tutors and tests, we plan to conduct a follow-up study after imposing a limit on the number of revisions allowed per problem.

Acknowledgments. Partial support for this work was provided by the National Science Foundation under grant DUE-0817187.

References

1. Aïmeur, E., Brassard, G., Dufort, H., Gambs, S.: CLARISSE: A Machine Learning Tool to Initialize Student Models. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 718–728. Springer, Heidelberg (2002)
2. Anderson, R.C., Kulhavy, R.W., Andre, T.: Feedback procedures in programmed instruction. J. Educational Psychology 62, 148–156 (1971)
3. Bierbaum, W.B.: Immediate knowledge of performance on multiple-choice tests. J. Programmed Instruction 3, 19–23 (1965)
4. Corbett, A.T., Anderson, J.R.: Locus of feedback control in computer-based tutoring: impact on learning rate, achievement and attitudes. In: Proc. SIGCHI, pp. 245–252 (2001)
5. Gilmer, J.S.: The Effects of Immediate Feedback Versus Traditional No-Feedback in a Testing Situation. In: Proc. Annual Meeting of the American Educational Research Association, pp. 8–12 (April 1979)
6. Kumar, A.N.: Error-Flagging Support for Testing and Its Effect on Adaptation. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 359–368. Springer, Heidelberg (2010)
7. Kumar, A.N.: A Scalable Solution for Adaptive Problem Sequencing and Its Evaluation. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) AH 2006. LNCS, vol. 4018, pp. 161–171. Springer, Heidelberg (2006)
8. Plake, B.S.: Effects of Informed Item Selection on Test Performance and Anxiety for Examinees Administered a Self-Adapted Test. Educational and Psychological Measurement 55(5), 736–742 (1995)
9. Shermis, M.D., Mzumara, H.R., Bublitz, S.T.: On Test and Computer Anxiety: Test Performance Under CAT and SAT Conditions. J. Education Computing Research 24(10), 57–75 (2001)
10. Tait, K., Hartley, J.R., Anderson, R.C.: Feedback procedures in computer-assisted arithmetic instruction. British Journal of Educational Psychology 43, 161–171 (1973)

Intelligent Tutoring Goes to the Museum in the Big City: A Pedagogical Agent for Informal Science Education

H. Chad Lane[1], Dan Noren[2], Daniel Auerbach[1]
Mike Birch[1], and William Swartout[1]

[1] Institute for Creative Technologies
University of Southern California
Playa Vista, CA USA
{lane,auerbach,mbirch,swartout}@ict.usc.edu
[2] Boston Museum of Science
One Science Park
Boston, MA USA
dnoren@mos.org

Abstract. In this paper, we describe *Coach Mike,* a virtual staff member at the Boston Museum of Science that seeks to help visitors at Robot Park, an interactive exhibit for computer programming. By tracking visitor interactions and through the use of animation, gestures, and synthesized speech, Coach Mike provides several forms of support that seek to improve the experiences of museum visitors. These include orientation tactics, exploration support, and problem solving guidance. Additional tactics use encouragement and humor to entice visitors to stay more deeply engaged. Preliminary analysis of interaction logs suggest that visitors can follow Coach Mike's guidance and may be less prone to immediate disengagement, but further study is needed.

Keywords: pedagogical agents, intelligent tutoring systems, coaching, informal science education, entertainment, computer science education.

1 Introduction

Since their inception in early 1960's, the list of intelligent tutoring system (ITS) success stories continues to grow [1, 2]. Most of these systems have been developed for use in formal learning environments and have the singular aim of producing cognitive gains in learners. Although the number of ITSs that consider non-cognitive issues, such as affect and metacognition, has grown rapidly in recent years [2], the most commonly sought outcomes of ITS research continues to be cognitive gains and deep understanding. While this focus is certainly justified, it is also worthwhile to take a broader perspective and investigate technologies that seek to inspire learners and promote the intrinsic value of learning. For the last half-century, this has been the goal of research on learning in informal settings, such as museums and science centers, where free choice and self-direction play prominent roles [3, 4]. Visitors decide *where* to go, *what* to do, and *how long* to do it. This elevates the prominence of motivation and affect given its role in these decisions. Any advanced learning technologies used in informal contexts should address these important non-cognitive

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 155–162, 2011.

factors. In this paper, we investigate the question of how ITS techniques can be applied in an informal setting where visitors are free to disengage at any moment.

1.1 Robot Park

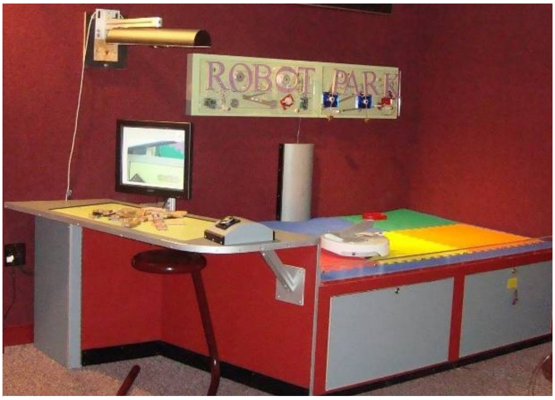


Fig. 1. The original Robot Park Exhibit at MoS

Located in Cahner's Computer Place at the Museum of Science (MoS), Boston. *Robot Park* is an interactive exhibit where visitors can control an iRobot Create^{TM} robot by assembling jigsaw-like blocks into chains of robot commands. It opened in October of 2007, was used by approximately 20,000 people in its first year [5], and continues as a permanent exhibit in the museum (see Figure 1). The primary purpose is to give visitors an opportunity to learn programming basics in a fun and engaging context. Each physical block corresponds to a robot action. Programs are compiled and executed by pressing a "run" button, which triggers a camera to take a snapshot of the programming area. Further, individual blocks can be placed on a tester so the visitor can see their effect. Commands are recognized by fiducial markers on top of the blocks, then transmitted to the robot. The programming language, *Tern*, includes basic movement actions, such as LEFT and FORWARD, others for sound and play, like GROWL and SHAKE, and some basic control structures. Studies have focused on Robot Park's tangible interface versus a graphical one, showing its ability to produce longer stay times, more sophisticated programs, and deeper conversations between visitors [5].

1.2 Pedagogical Agents and Informal Science Education

An established approach for reducing early disengagement from an interactive exhibit is to design for *immediate apprehendability*. This principle states that exhibits should use simple interfaces, leverage familiar ideas and controls, and give immediate feedback that allows visitors to self-monitor and observe changes [6]. The presence of museum staff has also been linked to a variety of positive outcomes, such as longer stay times [7] and greater proficiency with exhibits [8]. Given this result, it begs the question: would a virtual staff member achieve similar results?

In general, pedagogical agents can profoundly influence users' virtual experiences [9, 10]. Although the evidence is fragmented regarding their impact on learning [11], substantial evidence exists tying pedagogical agents' *external properties* (e.g., appearance) to non-cognitive outcomes, such as *satisfaction*, *interest*, and *sense of presence* [12]. Given the particularly important roles of these factors in informal settings, pedagogical agents seem like a natural fit. Indeed, a number of interactive virtual characters and robots have been developed for museums and other informal settings. These include our prior work with MoS, the virtual human twins Ada and Grace [13], the virtual robot Tinker [14], and the museum tour guide robot [15].

2 Coach Mike: An Informal Intelligent Tutor

We have rebuilt Robot Park with a 42” LCD screen to hold an embodied pedagogical agent named *Coach Mike* along with the original display to show programs. He seeks to help visitors understand and interact with the exhibit. In this section, we describe the conceptualization, design, and implementation of the system.

2.1 Interpretation at Robot Park

To design Coach Mike, we first turned to 59 museum staff and volunteers who work or had recently worked in Cahner’s Computer Place. They were asked about their experiences with Robot Park and to report (1) typical questions they are asked about the exhibit, (2) what they say to engage visitors, and (3) observations on how visitors interact with the exhibit and respond to help requests. Although some stylistic differences were evident, several themes did emerge:

- To initiate contact, staff often ask “Would you like to program this robot?” or “If you can give directions, you can program this robot.”
- Visitors tend to ask about the purpose of the exhibit and how to use the blocks.
- Initial explanations often involve exhibit internals (e.g., use of computer vision) and basic instructions on how to move the robot with the tester or run button.
- Specific programming problems are usually suggested for visitors, such as touching the target (which is built in to Robot Park just beneath the sign) or moving the robot in a specific pattern.
- Visitors usually ignore available documentation.

Most generally, these reports suggest that staff tend to encourage visitors to use Robot Park, explain how the software can read and execute programs, and then show them enough of the Tern language to enable visitors to write their own programs.

2.2 Personality, Body, Animations, and Voice

As noted earlier, the appearance of a pedagogical agent can influence affective outcomes. In previous work, we conducted surveys with museum visitors that suggested they preferred a virtual human guide that was approachable, energetic, intelligent, understanding, and patient [13]. We decided to seek these same qualities for Coach Mike. However, with a target audience of ages 7-12 and the general appeal of tangible interfaces to children [5], we chose to use a 3D, cartoon-style body, reminiscent of characters from modern animated films. This also helped distinguish Coach Mike from his fellow virtual staff members, Ada and Grace, who are photoreal and work in the same space, Cahner’s Computer Place. Lastly, to further distinguish Coach Mike, and with the hope that he might act as a role model for younger visitors, we decided to use the creator of Robot Park as inspiration for his appearance.[1]

[1] Dr. Michael Horn, now an Assistant Professor at Northwestern University, created Robot Park in his dissertation research on tangible interfaces at Tufts University.

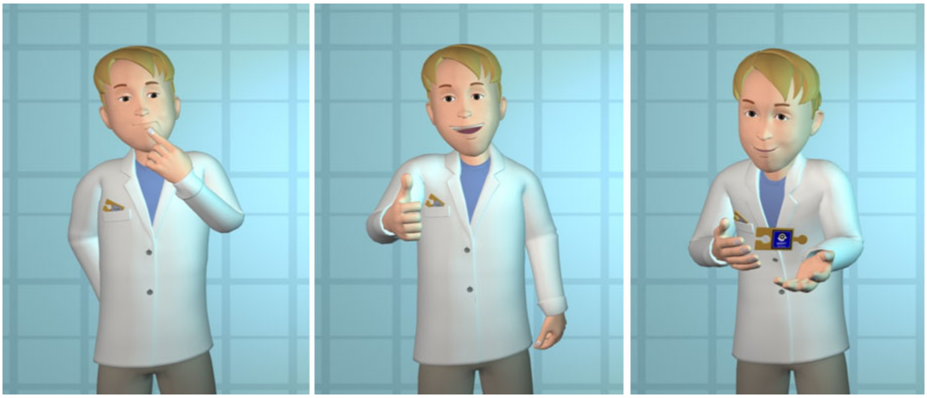


Fig. 2. Mike is a 3D cartoon-style pedagogical agent designed to be approachable, supportive, and understanding (among others). These stills are from animations for thinking, giving positive feedback, and displaying a block (magically).

Coach Mike has a total of 46 animations that range from very subtle to emotionally charged (see figure 2). The set includes basic gestures for breathing, basic idling (e.g., hands forward, hands back), natural communication (e.g., hands out and open, nodding, pointing), reactions to visitor programs (e.g., thinking, thumbs up, clapping), conveying empathy (e.g., head scratching, leaning), and showing blocks. We note that we decided to have blocks magically appear, hover for several seconds, then disappear with Coach Mike behaving as if he were a magician (the right-most image of figure 2 attempts to convey this idea). Other animations include one for flexing his muscles, knocking on the glass, looking all around, and raising his arms to signal a touchdown (as in American football). We have plans to examine the role of these animations in influencing visitor behaviors, attitudes, and interest in Robot Park.

Finally, although recorded speech is generally regarded as superior for clarity and conveying emotion [13], we decided to use synthesized speech for Coach Mike's voice. Given the need to mention a variety of blocks in different contexts, as well as provide support for several specific problems, we decided the flexibility afforded by synthesized speech outweighed the benefits of pre-recording all possible utterances. After considering roughly 20 commercially available speech synthesis systems, we chose a voice from *NeoSpeech* (www.neospeech.com) for its excellent clarity.

2.3 Implementation

Behind the agent is an ITS that shares many similarities with traditional tutoring systems, but also differs in some key ways. For instance, when no one is using Robot Park, Coach Mike waits patiently, occasionally entertaining himself by knocking on the glass (of his monitor), looking around, or using some minor passive gestures. These idle behaviors play a potentially critical role in the decisions of visitors to engage or not. When a visitor is detected, he directs his attention to the work area and greets that person. How the session proceeds from there depends primarily on the subsequent actions (or inaction) of the visitor.

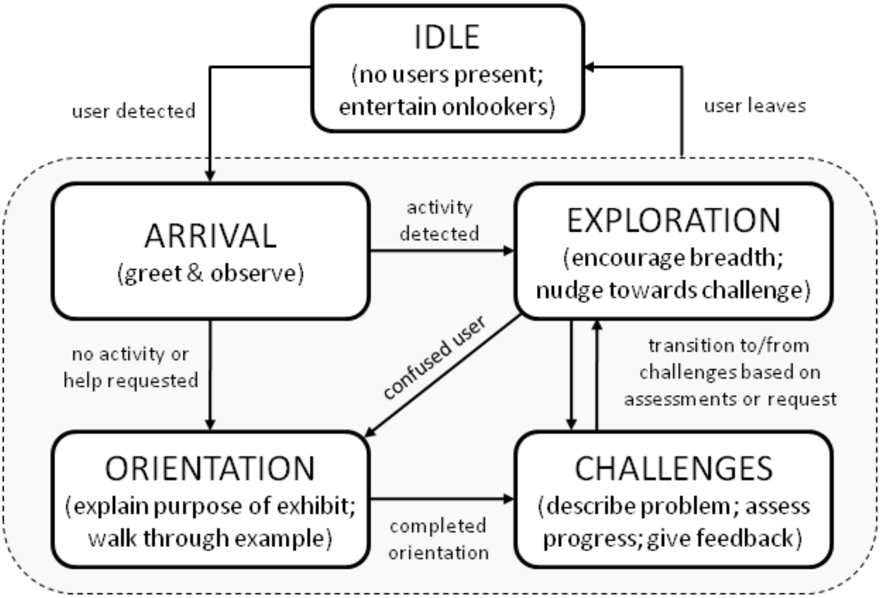

IDLE
(no users present; entertain onlookers)
user detected
user leaves
ARRIVAL
(greet & observe)
activity detected
EXPLORATION
(encourage breadth; nudge towards challenge)
no activity or help requested
confused user
transition to/from challenges based on assessments or request
ORIENTATION
(explain purpose of exhibit; walk through example)
completed orientation
CHALLENGES
(describe problem; assess progress; give feedback)

Fig. 3. State diagram for Coach Mike's interactions with visitors at Robot Park. The goal is to balance support for free exploration with specific problem guidance.

To allow Coach Mike to interact with visitors and monitor interactions with Robot Park, we augmented the existing system [5] with several new software components:

1. *Physical tracking*: weight-sensitive mat, robot camera, help button
2. *Virtual Human system*: animation, speech, lip syncing, art (see [13])
3. *Pedagogical Manager*: session manager, intelligent tutoring system.

The Pedagogical Manager acts as the hub by monitoring physical inputs from the exhibit (including tested blocks and programs), triggering virtual human actions (i.e., speaking and animating), assessing user actions, and providing learning support.

Pedagogical decisions are driven by a rule-based cognitive model of coaching implemented in Jess (www.jessrules.com). We chose Jess because of its ability to model a frequently changing world state and for the flexibility it provides for a modular representation of tutoring tactics. Built to simulate MoS staff's strategies (section 2.1), the model encodes a variety of tutoring and motivation tactics to orient people to the exhibit, encourage them to try new things, suggest specific problems (aka, "Mike's challenges"), and give knowledge-based feedback on their programs. A general aim is to balance the importance of exploration and play with the goal of giving feedback and guidance (as traditional ITSs do) for specific challenges.

Our model of coaching operates in three general modes: *Orientation, Exploration,* and *Challenge* (see figure 3). These capture the styles of interaction we observed with museum staff in our early analysis of interpretation at Robot Park and define the expectations maintained by the system for user behaviors at different times. Of course, informal settings demand robust and flexible policies (to support self-directed learning), and so when divergence from expectations is detected, Coach Mike adjusts accordingly – this is typically a shift to supporting exploration. Below we discuss these modes in more detail as well as the transitions between them (see figure 3).

Orientation. If no activity is detected upon arrival or if the visitor stops exploring fairly quickly, Coach Mike will provide a basic orientation showing how to write a simple program:

> **CM: Can you find the START block and place it on the tester?** [*animation of START block appearing over CM's hands*]
> V: [*holds the START block over the tester*]
> **CM: Great!** [*thumbs up animation*] **Now can you find the FORWARD block and place it on the tester?**
> V: [*holds the FORWARD block over the tester*]
> **CM: Awesome! Now can you attach them on the table and press the RUN button?** [*two blocks come together in CM's hands*]
> V: [*attaches blocks, presses RUN, robot moves forward*]
> **CM:** [*gazes at robot area during execution*] **Nice! When you pressed the RUN button, the camera took a picture of your program and transmitted it to the robot.** [*gesture to robot*]

This continues with Coach Mike asking the visitor to add another block to the program and extolling the value of programming with multi-step programs. If users demonstrate an ability to write a multi-step program on their own, this is not delivered and if difficulties arise, Coach Mike will repeat or provide additional guidance.

Exploration. If the visitor begins interacting with the exhibit upon arrival, or has completed (or abandoned) the challenge problems, Coach Mike supports free exploration. Here, the aim is to simply provide encouragement and promote continued engagement, but gently nudge the visitor towards creating goal-directed, multi-step programs. Tutoring tactics are primarily reactive by responding to variety (i.e., the visitor trying new blocks) and writing non-trivial programs (i.e., multi-step). For feedback, Coach Mike will provide specific explanations of blocks on their first use and see associated animations as part of his reaction. Continued exploration produces more reactions, sometimes including commentary on programs (e.g., "That was a long program. I love it!") or about the robot (e.g., "I think the robot is getting tired. Just kidding!").

Challenges. Coach Mike can also suggest specific problems to the visitor. For example, he might ask for the robot to touch the target or move in a specific pattern, such as a square. We chose a *constraint-based* representation for assessing programs because (1) solutions are checked only when submitted, and (2) constraints flexibly allow for multiple solutions [16]. For example, one constraint for the square problem is that the program should have three turns in the same direction. Another checks for moves between these turns. Further, hints and feedback are attached to constraints permitting messages like "The robot will need at least three turns to move in a square." Support for three problems is available and Coach Mike can provide multiple hints (including displaying pictures on the screen) to help visitors who are particularly frustrated. After a problem is solved, Coach Mike reacts by congratulating the visitor and using a special animation such as clapping, a double-thumbs up, or a fist pump.

Table 1. Robot Park analysis with and without Coach Mike. An *empty* session is defined as one with an abrupt departure (0-1 actions). Standard deviations are shown in parentheses.

session data	**Robot Park (original)**	**Robot Park w/Coach Mike**
duration (minutes)	2.38 (3.13)	3.25 (4.05)
tester uses	4.34 (9.30)	5.25 (8.81)
number of programs	5.33 (6.07)	6.79 (5.71)
average program length	6.46 (4.13)	5.03 (3.69)
blocks used (out of 11)	7.84 (3.38)	7.60 (3.74)
empty sessions /hour	3.01	0.44

3 Preliminary Analysis of Visitor Interactions

Coach Mike is scheduled to officially open at MoS in February 2011. During testing of a pre-release version of the system, however, we collected system logs from visitors' interactions at the exhibit of approximately 9 hours with Coach Mike active and 6 hours without. This version of the system *lacked* several important features, including most animations, support for all of Tern, and use of the help button. Further, staff was present in both sessions to help visitors in both testing sessions.

We first broke the log files up by session, defined as the arrival then departure of one or more visitors. The logs provided information about all uses of the tester, run button, and programs that were submitted. We counted these actions, the lengths of the programs submitted, and the coverage of the Tern program language by during the session (out of the 11 blocks available, how many were used). Table 1 shows the results of the analysis. Although anecdotally, staff reported that the presence of Coach Mike attracted visitors to Robot Park and the differences in these behavioral measures generally favor the presence of Coach Mike, none of the differences in Table 1 were found to be statistically significant. The lower number of quick disengagements from the exhibit may suggest visitors felt more compelled to engage the exhibit. However, we are unable to conclude from this data that the presence of Coach Mike, in this non-animated and limited form, had a substantial impact on the behaviors of visitors.

4 Conclusion and Future Work

In this paper we have described a pedagogical agent for informal science education, *Coach Mike,* that inhabits an exhibit in the Boston Museum of Science. Given that visitors can disengage at any moment in this informal setting, the underlying pedagogical model seeks to simultaneously keep the visitor engaged while promoting their learning of programming. Humor and entertaining animations are used to accomplish the former while specific problems, hints, and feedback are given for the latter. Although the preliminary analysis of interaction data revealed no specific benefits of Coach Mike's presence, this pre-release version of the system lacked important functionality for animation and complete support of problem solving. We are currently conducting formative testing, including in-person observation as well as log files analysis, to determine in more detail how people respond to Coach Mike's guidance, whether humor and entertaining animations induce deeper engagement, and how his presence influences conversations about programming and Robot Park.

Acknowledgments. This material is based upon support by the National Science Foundation under Grant 0813541. We also thank the inspiring staff and volunteers at MoS as well as the extremely creative virtual human and animation teams at ICT.

References

1. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent Tutoring Goes to School in the Big City. IJAIED 8, 30–43 (1997)
2. Woolf, B.P.: Building Intelligent Interactive Tutors: Student-centered Strategies for Revolutionizing E-learning. Morgan Kaufmann, Amsterdam (2009)
3. National Research Council: Learning science in informal environments: People, places, and pursuits. The National Academies Press, Washington, DC (2009)
4. Falk, J.H., Dierking, L.D., Foutz, S. (eds.): In Principle In Practice: Museums as Learning Institutions. AltaMira Press, Lanham (2007)
5. Horn, M.S., Solovey, E.T., Crouser, R.J., Jacob, R.J.K.: Comparing the use of tangible and graphical programming languages for informal science education. In: Proc. 27th Int. Conf. on Human Factors in Computing Systems, pp. 975–984. ACM, Boston (2009)
6. Allen, S.: Exhibit design in science museums: Dealing with a constructivist dilemma. In: Falk, J., Dierking, L., Foutz, S. (eds.) In Principle, In Practice: Museums as Learning Institutions, pp. 43–56. AltaMira Press, Lanham (2007)
7. Bailey, E., Bronnenkant, K., Kelley, J.: Visitor behavior at a constructivist exhibition: Evaluating Investigate!! at Boston's Museum of Science. In: Dufresne-Tasse, C. (ed.) Comm. for Ed. & Cultural Action, pp. 149–168. ICOM/CECA, Montreal, Canada (1988)
8. Randi Korn & Assoc.: Summative Eval: Search for Life. Hall of Science, NY (2006)
9. Dehn, D.M., Mulken, S.v.: The impact of animated interface agents: a review of empirical research. Int. J. Hum.-Comput. Stud. 52, 1–22 (2000)
10. Moreno, R., Mayer, R.E.: Personalized messages that promote science learning in virtual environments. Journal of Educational Psychology 96, 165–173 (2004)
11. Craig, S.D., Gholson, B., Driscoll, D.M.: Animated pedagogical agents in multimedia educational environments: Effects of agent properties, picture features, and redundancy. Journal of Educational Psychology 94, 428–434 (2002)
12. Baylor, A.L.: The impact of pedagogical agent image on affective outcomes. In: Int. Conf. on Intell. User Interfaces, San Diego, CA (2005)
13. Swartout, W., Traum, D., Artstein, R., Noren, D., et al.: Ada and Grace: Toward Realistic and Engaging Virtual Museum Guides. In: 10th Int. Conf. on Intell. Virtual Agents, vol. 6353, pp. 286–300. Springer, Philadelphia (2010)
14. Bickmore, T.W., Pfeifer, L., Schulman, D., Perera, S., Senanayake, C., Nazmi, I.: Public displays of affect: deploying relational agents in public spaces. In: CHI 2008 Extended Abstracts on Human Factors in Computing Systems, pp. 3297–3302. ACM, Florence (2008)
15. Burgard, W., Cremers, A.B., Fox, D., Hähnel, D., et al.: Experiences with an interactive museum tour-guide robot. AI 114, 3–55 (1999)
16. Mitrovic, A., Mayo, M., Suraweera, P., Martin, B.: Constraint-Based Tutors: A Success Story. In: Monostori, L., Váncza, J., Ali, M. (eds.) IEA/AIE 2001. LNCS (LNAI), vol. 2070, pp. 931–940. Springer, Heidelberg (2001)

Modeling Narrative-Centered Tutorial Decision Making in Guided Discovery Learning

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. Interactive narrative-centered learning environments offer significant potential for scaffolding guided discovery learning in rich virtual storyworlds while creating engaging and pedagogically effective experiences. Within these environments students actively participate in problem-solving activities. A significant challenge posed by narrative-centered learning environments is devising accurate models of narrative-centered tutorial decision making to craft customized story-based learning experiences for students. A promising approach is developing empirically driven models of narrative-centered tutorial decision-making. In this work, a dynamic Bayesian network has been designed to make narrative-centered tutorial decisions. The network parameters were learned from a corpus collected in a Wizard-of-Oz study in which narrative and tutorial planning activities were performed by humans. The performance of the resulting model was evaluated with respect to predictive accuracy and yields encouraging results.

Keywords: Narrative-centered learning environments, Game-based learning environments, Guided discovery learning, Dynamic Bayesian Networks.

1 Introduction

Recent years have seen significant growth in research on interactive narrative-centered learning environments for creating story-based learning experiences that are both engaging and pedagogically effective [1,2]. These environments encourage students to learn by actively participating in story-based problem-solving activities. Narrative-centered learning environments can form the basis for discovery learning [3] that supports students' active exploration of a subject matter. Discovery learning encourages students to learn by trial-and-error. Utilizing the scientific method, students pose questions, design and perform experiments, collect data, and evaluate hypotheses [4]. Despite the potential benefit of discovery learning, studies have indicated that it can be ineffective when students receive no guidance in the form of coaching and hints from a teacher or learning environment [5,6]. These studies suggest that discovery learning that is accompanied by guidance can be more effective than pure discovery learning [4,7].

Narrative-centered learning environments actively monitor students interacting with the unfolding storyworld to make decisions regarding the next action to perform

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 163–170, 2011.

in service of guiding students' learning experiences. Through this process, the system attempts to make effective narrative-centered tutorial decisions while managing the story structure and scaffolding student interaction. A key challenge for these environments is devising accurate models of narrative-centered tutorial decision-making, i.e., determining the next narrative-centered tutoring action to perform.

A promising approach to building effective interactive narrative-centered environments is devising empirically informed models of narrative-centered tutorial decision making. By utilizing a corpus of human interactions within a narrative environment, models of tutorial decision-making can be learned from data.

This paper presents a dynamic Bayesian network (DBN) approach to modeling narrative-centered tutorial decision-making. The approach supports learning models from a corpus and integrating different sources of evidence affecting decisions. A corpus collection study was conducted using a Wizard-of-Oz methodology with students interacting with a customized version of the CRYSTAL ISLAND interactive narrative-centered learning environment [2] in which wizards provide the narrative planning, tutorial planning, and natural language dialogue functionalities of the system. Students exhibited positive learning outcomes while interacting with the learning environment. Analyses of the DBN models learned from the corpus reveal that empirically informed dynamic Bayesian networks offer a promising approach for narrative-centered tutorial decision making. To our knowledge, this is the first model of narrative-centered tutorial decision making that has been learned from a corpus of human-human tutorial interactions.

2 Background

Narrative-centered learning environments provide students with the ability to actively participate in problem-solving activities by leveraging narrative to create engaging experiences in rich virtual interactive storyworlds. A broad range of techniques has been proposed to create interactive story-based learning environments that are both engaging and pedagogically effective. TEATRIX is designed to help students in the process of collaborative fairy-tale-based story creation [8]. Carmen's Bright IDEAS implements an agent-based interactive pedagogical drama. It is an interactive health intervention system designed to teach social problem-solving skills to mothers of pediatric cancer patients [9]. FEARNOT! is a storytelling application for social education against bullying [10]. By suggesting coping behaviors for virtual agents involved in bullying incidents, students develop empathetic relationships with the agents. STABILITY and SUPPORT OPERATIONS is a multi-agent system that features socially intelligent virtual humans to assist trainees for developing leadership and negotiation skills [11]. The TACTICAL LANGUAGE AND CULTURE TRAINING SYSTEM is designed to help students learning knowledge of foreign language and culture [1]. Plan-based representations have been explored for driving tailored scaffolding during narrative interaction with students [12]. Although prior work has investigated approaches for narrative and tutorial action selection, little work has explored the creation of empirically informed computational model of narrative-centered tutorial decision-making, which is the focus of the work reported here.

3 Narrative-Centered Tutorial Decision-Making Model

Interactive narrative is a time-based phenomenon. To be able to select the most appropriate tutorial decisions in narrative-centered learning environments, a model of narrative-centered tutorial decision making needs to utilize numerous observations that change over time. Because Dynamic Bayesian networks (DBNs) can explicitly characterize models' belief state over time, DBNs provide a natural representation for describing worlds that change dynamically over time [13], and DBNs have demonstrated significant promise for selecting tutorial actions in ITSs [14].

The high-level structure of the dynamic Bayesian network model created for narrative-centered tutorial decision-making is shown in Figure 1. The figure illustrates three time slices and their corresponding tutorial decisions: *tutorial decision*$_{t-2}$, *tutorial decision*$_{t-1}$, and *tutorial decision*$_{t}$. The three time slices include representations of the narrative observation including information on the physical state of the storyworld and progression of the narrative. Each time slice encodes a probabilistic representation of the belief about the overall state of the narrative.

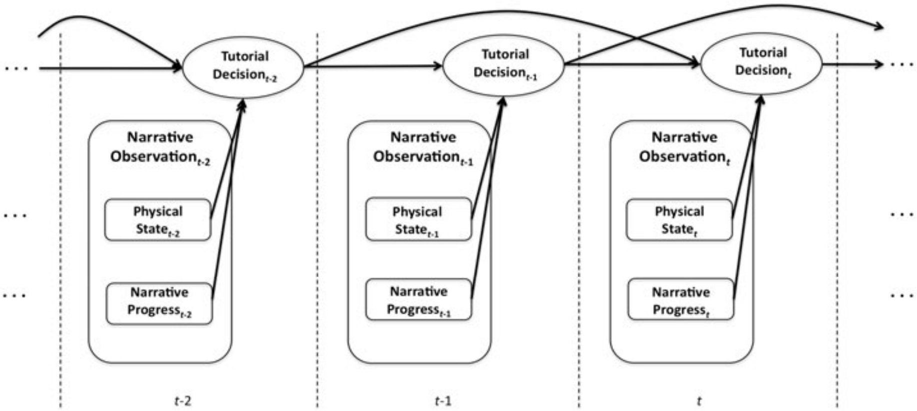

Tutorial Decision$_{t-2}$
Tutorial Decision$_{t-1}$
Tutorial Decision$_{t}$
Narrative Observation$_{t-2}$
Physical State$_{t-2}$
Narrative Progress$_{t-2}$
Narrative Observation$_{t-1}$
Physical State$_{t-1}$
Narrative Progress$_{t-1}$
Narrative Observation$_{t}$
Physical State$_{t}$
Narrative Progress$_{t}$
t-2
t-1
t

Fig. 1. Dynamic Bayesian network model for narrative-centered tutorial decision-making

The *tutorial decision* nodes model the knowledge of prior decisions. The *physical state* nodes model the location of characters in the storyworld (i.e., student, wizard) which are represented as quantized virtual world locations. The *narrative progress* nodes model the storyworld's narrative structure. To characterize the progress of the narrative, we analyzed the story structure utilizing a narrative arc framework. Utilizing the current phase of the narrative arc as an observation provides the model with evidence about the high level structure of the unfolding narrative [15]. The model considers the current beliefs about the physical state and narrative progress represented in *narrative observation*$_{t}$. It also considers prior history of *tutorial decision*$_{t-1}$ and *tutorial decision*$_{t-2}$. Using the links from *tutorial decision*$_{t-1}$, *tutorial decision*$_{t-2}$, *physical state*$_{t}$, and *narrative progress*$_{t}$ the model captures how each of these influences *tutorial decision*$_{t}$.

Given the DBN structure, the values in the conditional probability tables (CPTs) for each observation node in the network can be learned using a corpus. Setting observed evidence on the learned model and updating the network allows the likelihood of decisions to be computed at each time slice.

4 Corpus Collection Environment

A customized WOZ-enabled version of the CRYSTAL ISLAND narrative-centered learning environment with Wizard-of-Oz functionalities was created (Figure 2) to act as a corpus collection tool to investigate narrative-centered tutorial decision-making. CRYSTAL ISLAND is a virtual learning environment designed for the domain of microbiology for eighth grade science education featuring a science mystery situated on a remote tropical island. Within the story, the student plays the role of a science detective attempting to discover the identity of an infectious disease plaguing the island inhabitants. CRYSTAL ISLAND is built with Valve Corporation's Source™ Engine, the game engine utilized for Half Life®2.

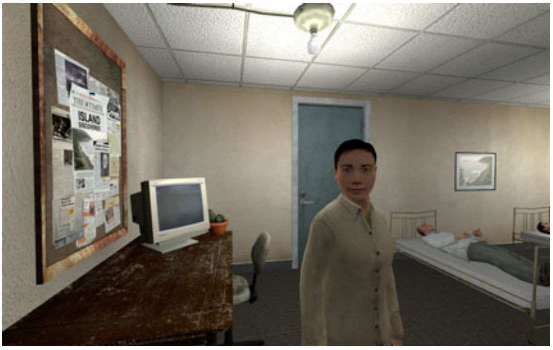


Fig. 2. WOZ-enabled CRYSTAL ISLAND

The WOZ-enabled CRYSTAL ISLAND [15] extends the learning environment, using the networked multiplayer features of the Source™ Engine, to include a character driven by a wizard, who assists the student in solving the mystery. Wizards provide the tutorial planning and narrative planning functionalities as well as spoken dialogue for their character. Playing the role of the camp nurse, the wizard works collaboratively 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 while engaging in problem-solving activities.

In addition to directing the navigation, spoken communication, and manipulation behaviors of the nurse character in the virtual environment, the wizard controls the progression of the story and scaffolds student interactions by utilizing the *narrative dashboard*. The narrative dashboard enables the wizard to initiate key narrative-centered tutorial decisions in the environment (e.g., introducing new patient symptoms) analogous to narrative-centered tutorial planners [16]. Table 1 describes the decisions that can be enacted by the wizard using the narrative dashboard.

Table 1. Narrative-centered tutorial decisions

Decisions	Tutorial Type	Descriptions	Freq
START-SESSION	*Define Problem*	Wizard gives a brief explanation of the student's objectives and goals.	6.2%
INTRODUCE-SCIENTIFIC-METHOD	*Background Information*	Wizard explains to the student and suggests they use the scientific method while diagnosing the mysterious illness.	6.2%
INTRODUCE-WORKSHEET	*Background Information*	Wizard explains usage of the diagnosis worksheet to help the student formulate and refine their hypothesis.	6.2%
EXAMINE-PATIENT-SYMPTOMS	*Hint*	Wizard and student work together to examine symptoms of each of the patients.	8.1%
UPDATE-WORKSHEET	*Confirm Understanding*	Wizard reminds the student to update the diagnosis worksheet with new knowledge and hypothesis.	13.7%
READ-DISEASE-BOOKS	*Hint/Advice*	Wizard guides the student to read relevant disease information in the library, which helps them refine their hypothesis.	13.9%
INTRODUCE-HEADACHE	*Hint*	Wizard triggers an action resulting in a patient moaning and complaining about having a headache.	6.2%
TEST-CAMP-ITEMS	*Advice*	Student and wizard test food items the expedition team took with them from camp.	5.4%
TEST-OUTSIDE-CAMP-ITEMS	*Advice*	Student and wizard test food items the team found during their expedition.	3.4%
TEST-CONTAMINATED-BANANAS	*Advice*	Student and wizard test the bananas, which end up being contaminated.	3.4%
INTRODUCE-DIRTY-WATER	*Hint*	Wizard triggers an event causing a door to open and a water bottle to appear in the infirmary room.	5.2%
INTRODUCE-LEG-CRAMPS	*Hint*	Wizard triggers an event causing one of the patients to complain about leg cramps.	3.6%
COMPLETE-WORKSHEET	*Confirm Understanding*	Wizard asks student to update all remaining information that has not been entered and formulate their final hypothesis.	6.4%
REPORT-RESOLUTION	*Confirm Understanding*	Wizard asks student to explain their final hypothesis and how they arrived at their conclusion using the scientific method.	6.2%
END-SESSION	*Confirm Understanding*	Wizard thanks student and tells her that the patients will be treated based on her finding.	6.2%

There are fifteen narrative-centered tutorial decisions that wizards enact in the environment. Table 1 also summarizes the relative frequency of each decision, i.e., the ratio of the number of occurrences of specific decisions to the total number of decisions in all sessions. The frequencies range from 3.4% to 13.9% (*M* = 6.7%, *SD* = 3.2%). The corpus collection environment records detailed logs of actions performed by the student and wizard within the virtual environment, including decisions made by the wizards using the narrative dashboard. These logs provide a rich source of data to build empirically driven models of narrative-centered tutorial decision making.

5 Corpus Collection Study Method

A corpus collection study was conducted with thirty-three eighth-grade students (15 males and 18 females) from a public school ranging in age from 13 to 15 (*M* = 13.79, *SD* = 0.65). Two wizards participated in the study, one male and one female. Each wizard was trained on the CRYSTAL ISLAND microbiology curriculum and participated in at least three training sessions with college students prior to the study. (Details of the wizard protocol can be found in [15].) Each session in the study involved a single wizard and a single student. The student and wizard were physically located in separate rooms throughout the session. The students' sessions lasted no more than sixty minutes (*M* = 38, *SD* = 5.15). After finishing the session, students completed a post-test which consisted of the same set of questions given as a pre-test approximately one week prior to the study session. The pre-test, trace data logs, and post-test were used to analyze the wizards' narrative-centered tutorial decision-making and measure learning outcomes while interacting with the WOZ-enabled CRYSTAL ISLAND. During model evaluation one of the participants was eliminated as an outlier—the data were more than three standard deviations from the mean—leaving thirty-two usable trace data logs.

6 Results and Discussion

For the DBN model, there are a total of 22 time slices, 88 nodes, and more than 830 conditional probabilities present in the narrative-centered tutorial decision-making network. The model was implemented with the GeNIe/SMILE Bayesian modeling and inference library developed at the University of Pittsburgh's Decision System Laboratory [17]. Given the network structure of the DBN, the probabilities of each node in the network were learned by performing parameter learning for the conditional probability tables (CPTs). The Expectation-Maximization algorithm from the SMILearn library was used to learn the CPT parameters. After CPT parameters were learned, the resulting network was used to make inferences about the narrative-centered tutorial decision nodes in the model.

An analysis was conducted to investigate the use of dynamic Bayesian networks for modeling narrative-centered tutorial decision-making. To compare the effectiveness of the DBN model against a baseline, a bi-gram model was developed in which only the previous tutorial decision was used to predict the next tutorial decision. This network structure was an appropriate baseline for comparing the more complex DBN model against because it presents the most basic form of our dynamic Bayesian network model for tutorial decision-making. The bi-gram model achieved a tutorial decision predictive accuracy of 71%. A leave-one-out cross validation method was employed for both the baseline model and the DBN model. To analyze the effectiveness of the DBN model for narrative-centered tutorial decision-making prediction, an aggregated confusion matrix was built for the model to compute the overall accuracy. In the prediction evaluation, the DBN model achieves tutorial decision prediction accuracy of 93.7%. The DBN model exhibited a 23% accuracy improvement over the bi-gram model. It appears that providing evidence regarding narrative structure, physical locations, and tutorial decision history can significantly improve narrative-centered tutorial decision prediction.

It is important to note that students interacting with the WOZ-enabled version of CRYSTAL ISLAND achieved significant learning outcomes. They 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 shows that the learning gains were significant, $t(19) = 6.24$, $p < 0.0001$. For the learning outcome analysis, thirteen of the participants were excluded due to incomplete data on either the pre-test or post-test.

7 Conclusion

Narrative-centered learning environments offer significant promise for guided discovery learning. Making narrative-centered tutorial decisions is critically important for achieving pedagogically effective story-based learning experiences. In this paper, we have presented an empirically driven narrative-centered tutorial decision-making model for interactive narrative centered learning environments. A corpus collection study was conducted using a Wizard-of-Oz methodology with students interacting with a WOZ-enabled version of the CRYSTAL ISLAND learning environment. Using machine learning, we automatically acquired a narrative decision-making model based on observations of the narrative-centered tutorial decision history, location, and narrative arc. The study reveals that students exhibited significant learning outcomes while interacting with the WOZ-enabled CRYSTAL ISLAND, and using dynamic Bayesian networks for narrative decision-making appears to be a promising approach to devising accurate models.

Two directions for future work are particularly important. First, it will be important to develop models that not only indicate the best narrative-centered tutorial decision to make but also the appropriate time to intervene. A follow-on investigation should be conducted to learn models of the proper timing of narrative-based tutorial decision-making behaviors that contribute to the most effective and engaging learning experiences. Second, during the study, wizards used natural language dialogue to guide students' activities and control the progression of the story, in addition to the utilizing the narrative dashboard. Devising adaptive models of dialogue for narrative-centered learning environments is a promising line of investigation.

Acknowledgments. The authors wish to thank members of the IntelliMedia Group for their assistance and Valve Corporation for access to the Source™ SDK. Special thanks to Joe Grafsgaard and Kate Lester for assisting with the study. This research was supported by the National Science Foundation under Grants REC-0632450 and DRL-0822200, and IIS-0812291. 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. 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)
2. Rowe, J., Shores, L., Mott, B., Lester, J.: Integrating learning and engagement in narrative-centered learning environments. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 166–177. Springer, Heidelberg (2010)

3. Bruner, J.: The Act of Discovery. Harv. Educ. Rev. 31, 21–32 (1961)
4. de Jong, T., Joolingen, W.: Scientific Discovery Learning with Computer Simulations of Conceptual Domains. Rev. Educ. Res. 68(2), 179–201 (1998)
5. Mayer, R.: Should There Be a Three-Strike Rule Against Pure Discovery Learning? American Psychologist 59(1), 4–19 (2004)
6. 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, 75–86 (2006)
7. Shulman, L., Keisler, E.: Learning by Discovery: A Critical Appraisal. Rand McNally, Chicago (1966)
8. Machado, I., Brna, P., Paiva, A.: Learning by Playing: Supporting and Guiding Story-Creation Activities. In: 10th International Conference on Artificial Intelligence in Education, Amsterdam, Netherlands, pp. 334–342 (2001)
9. Marsella, S., Johnson, W.L., LaBore, C.: Interactive Pedagogical Drama for Health Interventions. In: 11th International Conference on Artificial Intelligence in Education, Sydney, Australia (2003)
10. Aylett, R., Louchart, S., Dias, J., Paiva, A., Vala, M.: FearNot! - an experiment in emergent narrative. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 305–316. Springer, Heidelberg (2005)
11. 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)
12. Thomas, J., Young, R.M.: Using Task-Based Modeling to Generate Scaffolding in Narrative-Guided Exploratory Learning Environments. In: 14th International Conference on Artificial Intelligence in Education, Brighton, U.K, pp. 107–114 (2009)
13. Dean, T., Kanazawa, K.: A Model for Reasoning about Persistence and Causation. Computational Intelligence 147(3), 142–150 (1989)
14. Murray, R., VanLehn, K.: DT Tutor: A Decision-Theoretic, Dynamic Approach for Optimal Selection of Tutorial Actions. In: 5th International Conference on Intelligent Tutoring System, Montreal, Canada, pp. 153–162 (2000)
15. Lee, S., Mott, B., Lester, J.: Optimizing story-based learning: An investigation of student narrative profiles. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 155–165. Springer, Heidelberg (2010)
16. Mott, B., Lester, J.: Narrative-centered tutorial planning for inquiry-based learning environments. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 675–684. Springer, Heidelberg (2006)
17. Druzdzel, M.: SMILE: Structural Modeling, Inference, and Learning Engine and Genie: A Development Environment for Graphical Decision-Theoretic Models. In: 16th National Conference on Artificial Intelligence, Orlando, Florida, pp. 342–343 (1999)

Inducing and Tracking Confusion with Contradictions during Critical Thinking and Scientific Reasoning

Blair Lehman[1], Sidney K. D'Mello[1], Amber Chauncey Strain[1], Melissa Gross[1], Allyson Dobbins[1], Patricia Wallace[2], Keith Millis[2], and Arthur C. Graesser[1]

[1] Institute for Intelligent Systems, University of Memphis, Memphis, TN 38152
{balehman,sdmello,dchuncey,magross,
bdbbins1,graesser}@memphis.edu
[2] Department of Psychology, Northern Illinois University, Dekalb, Illinois 60115
{pwallace,kmillis}@niu.edu

Abstract. Cognitive disequilibrium and its affiliated affective state of confusion have been found to be beneficial to learning due to the effortful cognitive activities that accompany their experience. Although confusion naturally occurs during learning, it can be induced and scaffolded to increase learning opportunities. We addressed the possibility of induction in a study where learners engaged in trialogues on critical thinking and scientific reasoning topics with animated tutor and student agents. Confusion was induced by staging disagreements and contradictions between the animated agents, and the (human) learners were invited to provide their opinions. Self-reports of confusion and learner responses to embedded forced-choice questions indicated that the contradictions were successful at inducing confusion in the minds of the learners. The contradictions also resulted in enhanced learning gains under certain conditions.

Keywords: Confusion, cognitive disequilibrium, contradiction, affect, tutoring, intelligent tutoring systems, learning.

1 Introduction

Connections between complex learning and emotions have received increasing attention in the fields of psychology [1-3], education [4-6], neuroscience [7], and computer science [8-11]. An understanding of affect-learning connections is needed to design engaging educational artifacts that range from affect-sensitive intelligent tutoring systems (ITSs) on technical material to entertaining media [12, 13].

The fundamental assumption behind much of this research is that affect and cognition are inextricably bound and fundamental to learning. This assumption is reasonable if one realizes that learning inevitably involves failure and a host of affective responses. Negative emotions (e.g., confusion, irritation, frustration, anger, and sometimes rage) are ordinarily associated with making mistakes, diagnosing what went wrong, and struggling with impasses. Positive emotions (e.g., engagement, flow, delight, excitement, and eureka) are experienced when tasks are completed, challenges are conquered, and major discoveries are made.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 171–178, 2011.

Importantly, the relationship between affect and learning is more complex than a simple model which posits that positive emotions facilitate learning while negative emotions hinder learning. Perhaps one of the most significant and counterintuitive findings pertains to the role of confusion in promoting deep learning. Confusion occurs when students get stuck; are confronted with a contradiction, anomaly, or system breakdown; and are uncertain about what to do next. Confusion provides an opportunity for learning because it triggers active problem solving and reasoning, a view that is consistent with impasse-driven theories of learning [14-16].

Evidence for impasse-driven learning can be found in early work on skill acquisition and learning [14-16]. For example, in an analysis of over 100 hours of human-human tutorial dialogues, VanLehn et al. [16] reported that comprehension of physics concepts was rare when students did not reach an impasse, irrespective of the quality of explanations provided by the tutor. There is also some evidence that confusion is positively correlated with learning due to the activities associated with its resolution (i.e., effortful elaboration and causal reasoning during problem solving) [17, 18]. These activities involve desirable difficulties [19], which inspire greater depth of processing, more durable memory representations, and more successful retrieval [20].

In our view, the complex interplay between events that trigger confusion coupled with effortful impasse-resolution processes is the key to promoting deep learning. Learning is presumably not directly caused by confusion, but rather by the cognitive activities that accompany its experience. The benefits of impasses and confusion can only be leveraged in a learning environment (LE) if three conditions are met: (1) the LE has events that *induce* confusion; (2) the LE can detect and *track* the associated confusion; and (3) the LE *regulates* confusion in a way that maximizes learning. The focus of this paper is to systematically explore methods to induce confusion in the learner so this paper will mainly focus on research activities to advance this goal.

We describe a study in which confusion was experimentally induced in an LE with two pedagogical agents that engaged in a trialogue with the human learner. The two agents served as the medium through which confusion is induced over the course of learning critical thinking and scientific reasoning skills such as designing and evaluating research studies. We focused on two research questions. First, can confusion be induced when the agents contradict each other and ask the human learner to intervene? More specifically, will confusion be induced if one agent presents accurate information and the other presents inaccurate information? Second, what are the indicators of the induced confusion?

2 Method

2.1 Manipulation

We experimentally induced confusion with a contradictory information manipulation over the course of learning concepts in critical thinking (e.g., random assignment, experimenter bias). This is achieved by having the tutor and student agents stage a disagreement on an idea and eventually invite the human to intervene (note that student agent refers to an animated agent, the actual learner is referred to as participant or learner). The contradiction is expected to trigger conflict and force the participant to reflect, deliberate, and decide which opinion has more scientific merit.

Contradictions were introduced during trialogues identifying flaws in sample research studies. Some studies had subtle flaws while others were flawless. There were four contradictory information conditions. In the *true-true* condition, the tutor agent presented a correct opinion and the student agent agreed with the tutor; this is the no contradiction control. In the *true-false* condition, the tutor presented a correct opinion and the student agent disagreed by presenting an incorrect opinion. In contrast, it was the student agent who provided the correct opinion and the tutor agent who disagreed with an incorrect opinion in the *false-true* condition. Finally in the *false-false* condition, the tutor agent provided an incorrect opinion and the student agent agreed. It should be noted that all misleading information was corrected over the course of the trialogues and participants were fully debriefed at the end of the experiment.

The excerpt in Table 1 is an example trialogue between the two agents and the human learner. This is an excerpt from the *true-false* condition, where the tutor agent (Dr. Williams) and the student agent (Chris) are discussing a flawed study with Bob (the human learner).

Table 1. Excerpt of trialogue from *true-false* condition

Turn	Speaker	Dialogue Move
There was experiment done at a top University where students got the same grade whether they used the textbook or not. In the fall, science students were told that textbooks were optional. For the same class in the spring, students were told that reading the textbook was required. The researchers found no differences on the final exams. So there is no need to by textbooks <*Description of study*>		
1	**Dr. Williams**	**So Chris and I talked while you were reading and Chris thinks that there wasn't anything problematic about this study, but I think there was. <*Introduce contradiction*>**
2	Dr. Williams	How about you Bob? Would you not buy textbooks next semester based on this study? Please type buy or not buy. <*Forced-choice question*>
3	Bob	Not buy. <*Response*>
4	Dr. Williams	We are going to go over our thinking for this study before we come to any final decisions. <*Advance dialogue*>
5	**Chris**	**Well, I think how the participants were put into each condition was good, so that's not a problem. <*Assert information*>**
6	**Dr. Williams**	**It was problematic. <*Contradict with Chris*>**
7	Dr. Williams	Looks like we disagree. Bob, do you think there's a problem with how the participants were put into each group? Please type problem or no problem. <*Contradiction & forced-choice question*>
8	Bob	Problem. <*Response*>

2.2 Participants and Design

Participants were 32 undergraduate students from a mid south university in the US and participated for course credit. Data from one participant was discarded due to experimenter error. The experiment had a within-subjects design with four conditions (*true-true, true-false, false-true, false-false*). Participants completed two learning sessions in each of the four conditions with a different critical thinking topic in each session (8 in all). Order of conditions and topics and assignment of topics to conditions was counterbalanced across participants with a Graeco-Latin Square.

2.3 Procedure

The experiment occurred over two phases: (1) knowledge assessments and learning sessions and (2) a retrospective affect judgment protocol.

Knowledge Tests. Critical thinking knowledge was tested before and after learning sessions (pretest and posttest, respectively). Each test had 24 multiple-choice questions, three questions per concept (control group, construct validity, correlational studies, experimenter bias, generalizability, measure quality, random assignment, replication). There were three types of test items: definition, function, and example. Random assignment, for example, was assessed with the following questions: "Random assignment refers to __" (definition), "Random assignment is important because __" (function), and "Which study most likely did not use random assignment." (example). There were two alternate test versions and assignment was counterbalanced across participants for pretest and posttest.

Learning Sessions. First, participants signed an informed consent and then completed the pretest. Next, participants read a short introduction to critical thinking topics to familiarize them with the terms that would be discussed. Participants then began the first of eight learning sessions. A webcam and a commercially available screen capture program (Camtasia Studio^{TM}) recorded participants' face and screen, respectively, during the learning sessions.

Each learning session began with a description of a sample research study. Participants read the study and then began a trialogue with the agents. The discussion of each study involved four trials. For example, in Table 1 dialogue turns five through eight represent one trial. Each trial consisted of the student (turn 5) and tutor (turn 6) agents asserting opinions, prompting participants to intervene (turn 7), and obtaining participants' responses (turn 8).

This cycle was repeated in each trial, with each trial becoming increasingly more specific about the scientific merits of the study. The trialogue in Table 1 discusses a study that does not properly use random assignment. Trial 1 broadly asks if students would change their behavior based on the results of the study (turns 1-3), while Trial 2 addressed whether or not a problem is present (turns 5-8). Trial 3 began to specifically address the problematic part of the study, "Do the experimenters know that the two groups were equivalent?". Finally, Trial 4 directly addressed the use of random assignment, "Should the experimenters have used random assignment here?". Participants then completed the posttest after discussing the eight studies.

Retrospective Affect Judgment Protocol. Participants then completed a retrospective affect judgment protocol [21]. Videos of participants' face and screen were synchronized and participants made affect ratings while viewing these videos. Participants were provided with a list of affective states (anxiety, boredom, confusion, curiosity, delight, engagement/flow, frustration, surprise, and neutral) with definitions. Affect judgments occurred at 13 pre-specified points (e.g., after contradiction presentation, after forced-choice question, after learner response) in each learning session (104 in all). In addition to these pre-specified points, participants were able to manually pause the videos and provide affect judgments at any time.

3 Results and Discussion

We hypothesized that contradictory information would induce confusion in learners. To investigate this hypothesis, the experimental conditions (*true-false*, *false-true*, and *false-false*) were compared to the no-contradiction control condition (*true-true*) in two analyses: (1) self-reported levels of confusion and (2) responses to forced-choice questions. In addition, learning gains in experimental conditions were compared to the control condition.

3.1 Retrospective Self-report Confusion Ratings

Although a total of eight affective states were tracked, the present analysis only focuses on confusion because this is the primary dependent measure of interest. The analyses proceeded by computing proportional scores for self-reported confusion ratings in each condition. Paired sample t-tests indicated that there was significantly more confusion in the *true-false* condition (M = .06, SD = .10) than the *true-true* condition (M = .04, SD = .06), $t(30) = 2.02$, $p = .03$. However, the other experimental conditions (*false-true* and *false-false*) were not associated with significantly higher levels of confusion than the control (M = .04, SD = .06 and M = .05, SD = .08, respectively). These findings suggest that contradiction between agents can induce some confusion in learners. The success of contradiction, however, does appear to be tempered by who (tutor vs. student) takes the correct vs. incorrect position.

3.2 Tracking Uncertainty via Performance on Forced-Choice Questions

Self-reports are one viable method to track confusion. However, this measure is limited by the learner's sensitivity and willingness to report their confusion levels. A more subtle and promising measure of confusion and uncertainty is to assess learner responses to forced-choice questions following contradictions by the animated agents (see turns 3 and 8 in Table 1). Since these questions adopted a two-alternative multiple-choice format, random guessing would yield a score of 0.5. One-sample t-tests comparing learner responses to a chance value of 0.5 revealed the following pattern of performance: (a) *true-true* (M = .76, SD = .19) and *true-false* (M = .60, SD = .19) conditions were significantly greater than chance, (b) *false-true* (M = .45, SD = .26) was statistically indistinguishable from chance, and (c) *false-false* (M = .35, SD = .31) was significantly lower than chance. An ANOVA revealed the following pattern of response correctness across conditions: *true-true* > *true-false* > *false-true* > *false-false,* $F(3,90) = 16.9$, $Mse = .059$, $p < .001$, partial-eta squared = .39.

These results suggest that contradictions successfully evoked uncertainty. The magnitude of uncertainty was dependent upon the source and severity of the contradiction. Uncertainty is low when both agents are correct and there is no contradiction (*true-true*), but increases when one agent is incorrect. Uncertainty is greater when the tutor is incorrect (*false-true*) compared to when the tutor is correct (*true-false*), presumably because this challenges conventional norms. Finally, uncertainty is greatest when both agents are incorrect, even without a contradiction (*false-false*). Hence, uncertainty is maximized when learners detect a clash between their knowledge and the agents' responses. This uncertainty is a likely opportunity to scaffold deep comprehension by forcing learners to stop and think.

3.3 Learning Gains

Paired sample one-tail *t*-tests comparing the proportional learning gains in experimental conditions to the control condition were separately conducted for each question type (i.e., definition, function, example). Pretest and posttest scores were computed as the proportion of questions answered correctly. Proportional learning gains were computed as (posttest – pretest)/(1-pretest).

The results indicated that contradictions differentially impacted shallow and deep learning gains. For definition questions, the most shallow level, learning gains were marginally higher in the *true-true* condition (M = .24, SD = .59) than the *false-true* condition (M = .12, SD = .44), $t(30)$ = 1.87, p = .08, d = .22. However, this pattern was reversed for example questions that assess understanding at deeper levels. The *false-true* condition was marginally higher (M = .24, SD = .60) than the *true-true* condition (M = .00, SD = .64), $t(30)$ = 1.84, p = .08, d = .39. There were no significant learning gain differences for functional questions and with the other experimental conditions (*true-false*, *false-false*).

4 General Discussion

While recent research has identified a set of affective states that are very relevant to learning (e.g., boredom, engagement/flow, confusion, frustration, anxiety, curiosity), the question still remains of how to coordinate affective and cognitive processes to increase learning gains. The strategy we have adopted involves inducing particular affective states and subsequently helping learners regulate these affective states over the course of the session. The present paper reported on one such effort, specifically, on confusion induction during learning. Through the presentation of contradictory information, we were able to successfully induce confusion in learners. Both self-reports of confusion and learner responses to forced-choice questions showed that conditions with a contradiction induced more confusion than the no-contradiction control condition. Learner responses, however, may serve as a more effective and unbiased method to track confusion and uncertainty because learners might be hesitant to report that they are confused or might not be consciously aware of their confusion.

We did not expect impressive learning gains because confusion was only induced and not appropriately scaffolded in this preliminary study. Nevertheless, there were modest improvements in learning deeper content (example questions) in the *false-true* condition. This *false-true* condition was associated with chance-level responses to prompts (*intermediate confusion*), while responses were above chance for the *true-false* condition (*insufficient confusion*) and below chance for the *false-false condition* (*hopeless confusion*). Hence, the *false-true* condition which is associated with just the right level of confusion appears to be the most promising avenue for future research.

Since we have had some success in inducing confusion and uncertainty, the next step is to implement interventions that will make use of these learning opportunities. A learning environment (LE) that detects learner confusion has a variety of paths to pursue. The LE might want to keep the learner confused (i.e. in a state of cognitive disequilibrium) and leave it to the learner to actively deliberate and reflect on how to restore equilibrium. This view is consistent with a Piagetian theory [22] that stipulates that students need to experience cognitive disequilibrium for a sufficient amount of time before they adequately deliberate and reflect via self-regulation. If so, the LE should give indirect hints and generic pumps to get the student to do the talking when floundering. Alternatively, Vygotskian theory [23] suggests that it is not productive to have low ability students spend a long time experiencing negative affect in the face of failure. If so, the LE should give more direct hints and explanations. Another promising strategy to manage confusion is one recommended by VanLehn in his research on impasses during learning [16]. This strategy takes effect when confusion is detected and it entails: (a) prompting the student to reason and arrive at a solution, (b) prompting the student to explain their solution, and (c) providing the solution with an explanation only if the student fails to arrive at an answer. Further research will be required to compare the effectiveness of these interventions that aim to promote learning by inducing and intelligently managing confusion.

Acknowledgments. We thank our research colleagues in the Emotive Computing Group at the University of Memphis (http://emotion.autotutor.org). This research was supported by the National Science Foundation (REC 0106965, ITR 0325428, HCC 0834847). Any opinions, findings and conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of NSF.

References

1. Csikszetmihalyi, M.: Flow: The Psychology of Optimal Experience. Harper and Row, New York (1990)
2. Dweck, C.: Messages that Motivate: How Praise Molds Students' Beliefs, Motivation, and Performance (In Surprising Ways). In: Aronson, J. (ed.) Improving Academic Achievement: Impact of Psychological Factors on Education, pp. 38–61. Academic Press, Orlando (2002)
3. Stein, N., Hernandez, M., Trabasso, T.: Advances in Modeling Emotions and Thought: The Importance of Developmental, Online, and Multilevel Analysis. In: Lewis, M., Haviland-Jones, J.M., Barrett, L.F. (eds.) Handbook of Emotions, 3rd edn., pp. 574–586. Guilford Press, New York (2008)
4. Lepper, M., Woolverton, M.: The Wisdom of Practice: Lessons Learned from the Study of Highly Effective Tutors. In: Aronson, J. (ed.) Improving Academic Achievement: Impact of Psychological Factors on Education, pp. 135–158. Academic Press, Orlando (2002)
5. Meyer, D., Turner, J.: Re-Conceptualizing emotion and motivation to learn in classroom contexts. Educational Psychology Review 18(4), 377–390 (2006)
6. Schultz, P., Pekrun, R. (eds.): Emotion in Education. Academic Press, San Diego (2007)
7. Immordino-Yang, M.H., Damasio, A.: We Feel, Therefore We Learn: The Relevance of Affective and Social Neuroscience to Education. Mind, Brain and Education 1(1), 3–10 (2007)

8. 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, pp. 17–24. IOS Press, Amsterdam (2009)
9. 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)
10. Forbes-Riley, K., Litman, D.: Adapting to Student Uncertainty Improves Tutoring Dialogues. In: Dimitrova, V., Mizoguchi, R., Du Boulay, B., Graesser, A. (eds.) Proceedings of 14th International Conference on Artificial Intelligence in Education, pp. 33–40. IOS Press, Amsterdam (2009)
11. Robison, J., McQuiggan, S., Lester, J.: Evaluating the Consequences of Affective Feedback in Intelligent Tutoring Systems. In: Muhl, C., Heylen, D., Nijholt, A. (eds.) Proceedings of International Conference on Affective Computing & Intelligent Interaction, pp. 37–42. IEEE Computer Society Press, Los Alamitos (2009)
12. Graesser, A., Jeon, M., Dufty, D.: Agent Technologies designed to Facilitate Interactive Knowledge Construction. Discourse Processes 45(4-5), 298–322 (2008)
13. Litman, D., Silliman, S.: ITSPOKE: An Intelligent Tutoring Spoken Dialogue System. Paper presented at the Human Language Technology Conference: 4th Meeting of the North American Chapter of the Association for Computational Linguistics, Boston, MA (2004)
14. Brown, J., VanLehn, K.: Repair Theory: A Generative Theory of Bugs in Procedural Skills. Cognitive Science 4, 379–426 (1980)
15. Carroll, J., Kay, D.: Prompting, Feedback and Error Correction in the Design of a Scenario Machine. International Journal of Man-Machine Studies 28(1), 11–27 (1988)
16. VanLehn, K., Siler, S., Murray, C., Yamauchi, T., Baggett, W.: Why Do Only Some Events Cause Learning during Human Tutoring? Cognition and Instruction 21(3), 209–249 (2003)
17. D'Mello, S., Graesser, A.: Inducing and Tracking Confusion and Cognitive Disequilibrium with Breakdown Scenarios. Memory and Cognition (in press)
18. Graesser, A., Chipman, P., King, B., McDaniel, B., D'Mello, S.: Emotions and Learning with AutoTutor. In: Luckin, R., Koedinger, K., Greer, J. (eds.) 13th International Conference on Artificial Intelligence in Education, pp. 569–571. IOS Press, Amsterdam (2007)
19. Bjork, R.A., Linn, M.C.: The Science of Learning and the Learning of Science: Introducing Desirable Difficulties. American Psychological Society Observer 19, 3 (2006)
20. Craik, F.I., Lockhart, R.S.: Levels of Processing: A Framework for Memory Research. J. of Verbal Learning & Verbal Behavior 11(6), 671–684 (1972)
21. Graesser, A., McDaniel, B., Chipman, P., Witherspoon, A., D'Mello, S., Gholson, B.: Detection of Emotions during Learning with AutoTutor. Paper Presented at the 28th Annual Conference of the Cognitive Science Society, Vancouver, Canada (2006)
22. Piaget, J.: The origins of intelligence. International University Press, New York (1952)
23. Vygotsky, L.: Mind in society: The development of higher psychological processes. Harvard University Press, Cambridge (1978)

Students' Understanding of Their Student Model

Yanjin Long and Vincent Aleven

Human Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
{ylong,aleven}@cs.cmu.edu

Abstract. Open Learner Models (OLM) are believed to facilitate students' metacognitive activities in learning. Inspectable student models are a simple but very common form of OLM that grant students opportunities to get feedback on their knowledge and reflect on it. This paper uses individualized surveys and interviews with high school students who have at least three years experience learning with the Cognitive Tutor regarding the inspectable student model in the Tutor. We also interviewed a teacher. We found that: i) students pay close attention to the OLM and report that seeing it change encourages them to learn; ii) there is a significant discrepancy between the students' self-assessment and the system's assessment; iii) students generally rely on the OLM to make judgments of their learning progress without much active reflection. We discuss potential revisions to the student model based on the findings, which aim to enhance students' reflection on and self-assessment of their own learning.

Keywords: Open learner model, student model, self-assessment, Cognitive Tutor.

1 Introduction

Recently, many Intelligent Tutoring Systems (ITSs) researchers have studied the potential benefits of an Open Learner Model (OLM), in particular, whether it can help to improve students' metacognitive skills [5]. An OLM is a model accessible to the students that displays details of the student's learning status, such as their knowledge, difficulties, misconceptions, etc. [4]. Bull summarizes four primary OLM types: inspectable, co-operative, editable, and negotiated models [3]. The current work focuses on the first type, inspectable student models, which are the least sophisticated but probably the most common, as we argue below. As Bull and Kay [4] point out, a key purpose of an OLM is to support metacognitive activities such as reflection, planning and self-assessment. The model provides feedback with respect to students' learning and knowledge and it may trigger and facilitate metacognitive activities.

There has been only a limited amount of empirical work that supports the notion that OLMs can facilitate metacognition. In a survey study by Bull regarding college students' attitudes toward potential OLMs [3], most students expressed interest in accessing the models for the purpose of planning their learning and reflecting on it. The OLM was also viewed as a useful navigation aid. However, this survey was conducted before students actually used the tutor. A small number of investigations concentrated on students' field experience with student models. Three such studies suggest that even relatively simple inspectable student models can foster useful

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 179–186, 2011.

reflection by students and can enhance their domain-level learning and motivation. Arroyo *et al.* conducted an experiment to investigate the effects of an OLM that presented simple statistics about the given student's recent domain-level performance, together with metacognitive tips [2]. They found that students in the OLM group achieved greater learning gains and exhibited higher engagement than students who learned without the OLM. By contrast, metacognitive tips alone, without the accompanying simple OLM, were ineffective. A study by Mitrovic and Martin [8] with the SQL tutor investigated the effect of a simple inspectable student model that displayed (in the form of skill bars) students' progress in learning key concepts. They found that this OLM enhanced students' self-assessment and domain-level learning, especially for the less-able students. Finally, a study by Walonoski and Heffernan showed that an inspectable OLM can help reduce behaviors that reflect poor metacognition [10]. They designed an OLM for the purpose of counteracting students' "gaming the system" behaviors. The model plots a graphical trace of student actions with the system, in which gaming behaviors are easily visible. They found that the graphical feedback led to reduced gaming, perhaps due to greater reflection on the part of students, or because the display results in social pressure not to engage in gaming behaviors. However, no significant advantage on learning was found.

Although these studies highlight interesting connections between metacognition and OLMs and some tantalizing evidence about a potential positive influence of OLMs on metacognitive processes, little is known about whether and how OLMs might enhance the accuracy of students' self-assessment of their mastery of *specific* skills and concepts targeted in the instruction. Self-assessment has been recognized as a crucial metacognitive skill in self-regulated learning [11]. Accurate self-assessment can help students be aware of their difficulties and misconceptions, allocate attention to the proper learning topics, and even assist them in making learning plans [7].

We investigate relations between self-assessment and inspectable OLMs in the context of Cognitive Tutor, an ITS developed at Carnegie Mellon University since the early 1980s. This ITS is being used as part of the regular mathematics instruction in many US schools, and therefore provides an opportunity to study relations between self-assessment and OLMs in a real educational context with students who use the tutor over extended periods of time. In the current Cognitive Tutors, a skillometer (Fig.1) serves as an inspectable student model. It displays probabilities of skill mastery for the skills targeted in the current section of the tutor curriculum. Although the skillometer is a simple inspectable OLM, this type is in widespread use, not only in Cognitive Tutors, but also in constraint-based tutors, as mentioned above. The probabilities in the skillometer are calculated using a knowledge-tracing algorithm [6]. The skill bars gradually "grow" as students progress in the tutor and finally turn gold when the skill is fully mastered. The skillometer was added to the Cognitive Tutor to give students a sense of progress, and to help them understand how close they are to finishing a section of the tutor curriculum. An important assumption in Cognitive Tutors is that the skills in the tutor's cognitive model (and displayed in the skillometer) correspond closely to students' psychological reality. This assumption finds support both in Anderson's ACT-R theory [1] and in educational data mining results which show that the particular cognitive models used in tutors accurately account for student performance change over time [9].

Anecdotal reports from Cognitive Tutor classrooms indicate that students tend to pay close attention to their skillometers, perhaps affirming that they indeed serve as useful progress indicators. One might expect that the skillometer would also afford students opportunities to get feedback on the state of their knowledge and reflect on it, such as, for example: "Why have I not mastered this skill yet?" However, little prior work has investigated how students actually use the skillometers and whether this use facilitates students' self-assessment and reflection on their own skill mastery.

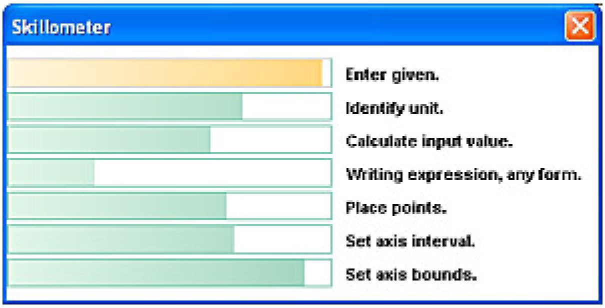


Fig. 1. Screenshot of the Skillometer

The current study uses data from an individualized survey to find out whether an inspectable model can influence students' self-assessment. Specifically, we compared students' self-assessment against the system's assessment of their skill mastery, as displayed in the skillometer. We also investigated whether students were more likely to reflect on their own skill mastery when they *disagree* with the skillometer, which Bull and Kay suggest may be a key advantage of an inspectable student model [4]. Finally, we conducted interviews with students and a teacher to supplement the findings from the survey with detailed observations and explanations.

2 Survey with Cognitive Tutor Students

The purpose of the survey is to find out i) to what extent students' self-assessment of their skill mastery agrees with the system's student model (which, as mentioned, reflects the probability of mastery of each skill, as inferred from their performance over a range of problems) and ii) the relation between students' disagreement with the student model and their reflective activities.

2.1 Participants, Materials and Procedure

The survey was conducted in a high school in a school district near Pittsburgh. A total of 47 students completed the survey. All the students were enrolled in Cognitive Tutor classes with the same teacher, including Algebra I, Algebra II and Geometry. The age ranged from 15 to 18 years old, and all the students have been in Cognitive Tutor classes for at least three years.

In order to investigate relations between students' assessment of their own skills and the system's assessment, *individualized* survey forms were created, as follows: For each student, a "high skill" and a "low skill" were identified just prior to administering the survey, using automated reports provided by the tutoring software. A high skill had a probability of mastery above 0.6 (according to the tutor's knowledge-tracing algorithm), a low skill a probability lower than 0.4. Individualized survey forms were then put together with three groups of questions, the first two of which varied by the individual student: (1) questions about the high skill (2) questions about the low skill and (3) general questions about the skillometer. For both skills, the participants were asked to rate their overall mastery of the skill on a 7-point Likert scale. They were also asked to self-rate various additional aspects of their mastery and understanding of the skill, such as whether they are good at using this skill, whether they can give an example of a problem in which the skill would be used, and whether they feel they need more practice with the skill. Due to technical problems, we did not have skill levels available for all students at the time we designed the surveys, so we also created a generic version of the survey, which was the same as the individualized version, except that the skills referred to in the first two sections were randomly picked. Only the third sections of these generic surveys were analyzed; the first two parts were added only to make all surveys look equivalent to the participants.

All the surveys were handed out during the students' Cognitive Tutor class time and each took less than 10 minutes to finish. The students were not logged in to the tutor at the moment the surveys were taken, so they could not look at the OLM.

2.2 Results

A total of 47 students participated in the survey, of whom 35 completed an individualized version and 12 completed the generic one.

Agreement between Self-Assessment and System-Assessment. The 35 individualized surveys were analyzed to test whether students' self-assessment of their skills agrees with the system's assessment, as captured in the student model. Specifically, we tested whether the survey scores for the high skill are higher than those for the low skill. As mentioned, students rated their skill mastery on a scale from 1 to 7, where 7 represents greatest level of mastery. For the high skill, the average rating was 4.969 (*SD*= 1.402), and for the low skill, the average rating was 5.156 (*SD*: 1.629); this difference is not statistically significant ($t(30)=-1.329$, $p = 0.194$).

Table 1. Participants' Responses to Other Self-Assessment Questions

		High Skill	Low Skill
Good at the Skill or Not?	Yes	24	25
	No	8	6
	Not Sure	3	4
Give an Example of the Skill	Yes	8	9
	No	27	26
More Practice on this Skill?	Yes	23	24
	No	7	8
	Not Sure	5	3

Additionally, Table 1 summarizes results from the other three self-assessment questions for both the high and low skills. We see that students' answers to the three questions do not differ much between the high and low skills. For example, 24 and 25 participants rated they were good at using the high and low skills, respectively. The results indicate a discrepancy between the students' perception of their skill mastery and the system's OLM. This discrepancy may be due to inaccurate self-assessment on the part of the students regarding their skill levels. Additionally, it is possible that the descriptions of the skills as they occur in the skillometer are not meaningful or understandable to the students. In the survey, the skills were described using the same short phrases that appear in the skillometer, illustrated in Fig. 1.

The question asking the students to give an example of a mathematics problem that involves the given skill was included mainly to test students' understanding of the skills displayed in the OLM. Two raters independently evaluated the answers. Not surprisingly, given the challenging nature of the question, only 8 (22.9%) participants gave examples for the high skill and 9 (25.7%) for the low skill. The examples given by the 17 students were mostly correct and were in the same format as they were presented in the Cognitive Tutor. We also found that the majority of students (23 for high skill, and 24 for low skill) preferred more practice on the skills. This preference for more practice is quite interesting. Again it is striking that there is no difference between the high skill and low skill questions, which may be evidence that students have difficulty in assessing their own skill.

Relation between Disagreement and Reflection. The results came from the third part of the survey, and all 47 participants' answers were analyzed.

Table 2. Cross-Table of Disagreement and Reflections

		Disagreement		
		Yes	No	Total
Reflection	Yes	19	13	32
	No	11	4	15
	Total	30	17	47

Table 2 presents results from Question 1 "Do you sometimes disagree with the skillbar?" and Question 4 "Do you reflect on what you have learned in the tutor when you finish each section?" A majority of participants indicated that they sometimes disagreed with the skillometer (30 participants, 63.8%) and reflected on their learning (32, 68.1%). The relationship between students' disagreement and reflection is not statistically significant (*chi*(1)=.862, *p*=.353). Thus, our study finds no strong support for Bull and Kay's hypothesis [4] that disagreement with the OLM leads to reflection.

For Question 3 "Does the skillbar accurately describe what you know and what you don't know in the tutor?", students' answers varied considerably. 23 students (48.9%) answered yes, 15 (31.9%) answered no, 2 (4.3%) answered "sometimes" and 7 (14.9%) indicated "not sure". In response to the question "How often do you look at the skillbar in your tutor?" 28 (59.6%) participants reported they look at the skillometer each time they finish a problem and 9 (19.1%) that they refer to it several times per session. These findings confirm that the majority of the students pay close attention to the skillometer, as we had heard in anecdotal reports from the classroom.

2.3 Discussion

It is notable that there is a significant discrepancy between students' self-assessment and the system's assessment. It is reasonable to assume that the tutor's knowledge-tracing algorithm is accurate and that the skills in the tutor's cognitive model (which are displayed in the skillometer) accurately represent the knowledge components that students are actually learning, given the amount of research and development effort that has been invested in this area [1][6][9].Therefore, the discrepancy between the student's and system's assessment may indicate inaccurate self-assessment abilities of the students. It is possible also that the students have trouble understanding the skill names used in the skillometer, especially outside the Tutor.

In an inspectable student model, the students are simply viewing the model. Even if they sometimes disagree with the model, they cannot express this disagreement or "argue" with the model. Results from the survey suggest the need for negotiation with the students to some extent, since more than 60% students expressed disagreement with the skillometer. One of the goals of the interview portion of our study, therefore, was to hear students' viewpoints with respect to a possible negotiable student model.

3 Interview

Individual interviews were conducted to further investigate students' understanding about the skillometer, as well as to clarify some issues that emerged from the surveys.

3.1 Participants, Materials and Procedure

Five male students from the same teacher's Cognitive Tutor classes volunteered to participate in the interview. The interview was conducted individually in a conference room at the school All interviews were audio recorded with consent from both the students and parents. Each interview took 15 to 20 minutes. The students answered 15 questions regarding their perception and understanding of the skillometer. The 15 questions addressed the following themes: 1) how well do the students understand the skillometer? 2) how often do they trust/disagree with the skillometer? And 3) how much control do they prefer to have in the Tutoring system?

In order to gain a perspective from an instructor, a follow-up interview was conducted through email with the Cognitive Tutor teacher.

3.2 Results and Discussion

All five participants claimed that they paid close attention to the skillometer when they were using the Tutor. They also said that seeing the skill bars change encouraged them to learn in the system. As one student said "It keeps you wanting to go. If it goes down, you get mad. If it goes up, that makes you want to work better."

Understanding of the Skillometer. In general, the participants understand how the skillometer changes in response to their interactions with the tutor, although some misunderstandings exist as well. For example, three participants indicated that the bars would keep going down when they asked for further hint levels, which is not accurate. On the other hand, the teacher stated that some of the skill names were confusing even to her, and the students would ask her for explanations for the skill names from time to time. In future designs of the skillometer, we need to ensure that all skill names can be easily understood, or that other means are used to communicate what the names mean (e.g., examples linked to the skillometer).

Need for Negotiation and Control. All participants stated that they sometimes disagreed with the skillometer, and they would be upset if the system did not allow them to progress to the next section when all skills were mastered. It might be an interesting idea to let them choose their own problems when working with the Cognitive Tutor. However, none of the participants actually prefer to pick their own problems instead of letting the system choose. One of the students said "that could be useful, but I can see … how it could be abused, just like you just choose problems that were easier for you to do." In general, these findings suggest that there is interest in negotiating with the system about the content of the student model. At the same time, the students do not seem to want strong control over their learning process. They trust the system and find it convenient to rely on it. So it is still an open question how much control a negotiable student model should give to students.

Lack of Reflection and Self-Assessment. The students rely heavily on the skillometer to decide what they know and what they still need to learn, in other words, the students do not usually reflect on or try to assess their own mastery of the skills targeted in the tutor. The perspective from the teacher confirms this observation. She wrote "I do not think students have good self assessment of their own skill levels. I feel they are just concerned with getting their bars yellow, but are not too concerned with what the bars mean or say." Also "I do not think that most of my students take time to reflect. Unfortunately, they just want to get it done and move on." These results bring up an essential question. The inspectable student model supports students in telling what they have mastered and what they have yet to master, and thus gives them clues as to what they should still work on. However, such convenience may hinder their thinking and reflection during the learning process, and reinforces a simplified notion of progress as only the changing of the skill bars. Perhaps prompting the students to assess their own skills first, before comparing with the skillometer, can be a better way of facilitating reflection on the part of students.

4 Future Work and Conclusion

In sum, this study confirms that students generally pay close attention to the skillometer. They stated that seeing the skill bars change encourages them to learn. We also find a significant discrepancy between students' self-assessment and the system's, which indicates perhaps that the student model is not fully understandable, but also that there is room for improvement in students' self-assessment abilities.

The long-term goal of the current project is to investigate how a student model can assist students in productive reflection and better self-assessment of skill mastery. Specifically, an interactive negotiable student model that prompts students to reflect may result in more advanced self-assessment abilities, combined with support for comparing with the information in the inspectable student model. It will be interesting to investigate how much control a negotiable student model should give to students in order to achieve the best learning outcome. Another interesting future topic might be showing students indicators of their improvement in the skillometer, analogous to Arroyo et al.'s simple progress indicators [2]. Finally, more in-depth qualitative methods like think-aloud protocols can be used in investigations to find out more information regarding students' understanding of the skillometer and motivation.

Acknowledgments. We would like to thank Gail Kusbit, Matthew Bernacki and Tristan Nixon for their kind help with this work. We would also like to thank the participating teacher and students. This work was funded by an NSF grant to the Pittsburgh Science of Learning Center (NSF Award SBE0354420).

References

1. Anderson, J.R.: Rules of the Mind. Erlbaum, Hillsdale (1993)
2. Arroyo, I., Ferguson, K., Johns, J., Dragon, T., Mehranian, H., Fisher, D., Barto, A., Mahadevan, S., Woolf, B.: Repairing Disengagement with Non Invasive Interventions. In: International Conference on Artificial Intelligence in Education, Marina del Rey, CA, pp. 195–202 (2007)
3. Bull, S.: Supporting Learning with Open Learner Models. In: 4th Hellenic Conference: Information and Communication Technologies in Education, Athens (2004)
4. Bull, S., Kay, J.: Student Models that Invite the Learner In: The SMILI Open Learner Modeling Framework. International Journal of Artificial Intelligence in Education 17(2), 89–120 (2007)
5. Bull, S., Kay, J.: Metacognition and Open Learner Models. In: Proceedings of Third Workshop Meta-Cognition and Self-Regulated Learning in Educational Technologies, ITS 2008 (2008)
6. Corbett, A., McLaughlin, M., Scarpinatto, K.: Modeling Student Knowledge: Cognitive Tutor in High School & College. User Modeling and User-Adapted Interaction 10, 81–108 (2000)
7. Mitrović, A.: Investigating students' self-assessment skills. In: Bauer, M., Gmytrasiewicz, P.J., Vassileva, J. (eds.) UM 2001. LNCS (LNAI), vol. 2109, pp. 247–250. Springer, Heidelberg (2001)
8. Mitrovic, A., Martin, B.: Evaluating the Effects of Open Student Models on Learning. In: de Bra, P., Brusilovsky, P., Conejo, R. (eds.) Proceedings of 2nd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems, pp. 296–305. Springer, Heidelberg (2002)
9. Ritter, S., Harris, T., Nixon, T., Dickison, D., Murray, C., Towle, B.: Reducing the Knowledge Tracing Space. In: Barnes, Desmarais, Romero, Ventura (eds.) Proceedings of the 2nd International Conference on Educational Data Mining, Cordoba, Spain, pp. 151–160 (2009)
10. Walonoski, J.A., Heffernan, N.T.: Prevention 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. 722–724. Springer, Heidelberg (2006)
11. Winne, P.H., Hadwin, A.F.: Studying as Self-Regulated Learning. In: Hacker, D.J., Dunlosky, J., Graesser, A.C. (eds.) Metacognition in Educational Theory and Practice, pp. 279–306. Erlbaum, Hillsdale (1998)

Workflow-Based Assessment of Student Online Activities with Topic and Dialogue Role Classification

Jun Ma, Jeon-Hyung Kang, Erin Shaw, and Jihie Kim

Information Science Institute, University of Southern California
4676 Admiralty Way, Marina del Rey CA 90292, United States
{junma,jeonhyuk,shaw,jihie}@isi.edu

Abstract. The Pedagogical Assessment Workflow System (PAWS) is a new workflow-based pedagogical assessment framework that enables the efficient and robust integration of diverse datasets for the purposes of student assessment. The paper highlights two particular e-learning workflows supported by PAWS. The first workflow correlates student performance, as measured by project grades, with different dialogue roles, *information seeker* and *information provider*, that students take on in project-based discussion forums. The second workflow identifies the distribution of question topics within student discussions. Both workflows employ state of the art natural language processing techniques and machine learning algorithms for dialogue classification tasks. Workflow results were reviewed with a course instructor and feedback regarding the analysis and its fidelity are reported.

Keywords: Discourse analysis, workflow technology, discussion assessment.

1 Introduction

Online discussion forums are now an integral component of the virtual learning environments that are centrally supported by many colleges and universities, and have become an essential tool for student-student and student-instructor communication beyond the walls of the classroom. Course discussion forums contain rich information about student understanding of course concepts and assignments, and the resulting information provides invaluable feedback for instructors, allowing them to respond formatively to student concerns. However, even when instructors do participate in forums, they often do so question-by-question. In heavily used forums, patterns of participation can be impossible to discern, and patterns of discussion difficult to connect with course concepts. With better models for forum assessment it may be possible to better to identify misunderstanding and predict course performance.

As the use of online forums and other collaborative virtual learning technologies increase, the resulting heavier interactions introduce a considerable burden for teachers who wish to support their students' online activities. The Pedagogical Workflows project has developed a scalable e-learning framework to support efficient and robust integration of diverse datasets for the purposes of student assessment. The Pedagogical Assessment Workflow System (PAWS) employs the same computational workflow technologies that support scientific applications in the fields of seismology

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 187–195, 2011.

and astronomy [1]. PAWS facilitates the efficient processing and robust analysis of large amounts of data. A grid computing service acts as the backend of the system [2]. These existing workflow generation and execution approaches are applied to make online assessment accessible to instructors. Workflow results are used to answer questions and provide formative feedback to instructors to facilitate "just in time" instructional adaptation to students learning and needs.

Recent work on integrating state of the art topic modeling and dialogue role classification techniques into PAWS is presented in this paper. The resulting classification results were correlated with other types of data, including questionnaire responses and project grades, through the workflows. Initial feedback on the resulting analysis was collected from a course instructor whose student discussions were fed on-demand into PAWS through a data collection service. The goal was that instructors would directly benefit from these new text tools.

2 Topic and Dialog Role Classification

The following sections describe the classification techniques used in PAWS.

2.1 SVM Classification Models for Online Discussion Threads

Support vector machine (SVM) is a widely used model in computer science and machine learning to perform classification tasks. PAWS uses SVM to classify both types of messages, i.e., *question* or *answer*, and types of users, i.e., *information seeker* or *information provider*, with respect to their dialogue roles in discussion forums. These classifications are important for the following reasons:

1. A student's dialogue role indicates whether the student is asking for help or providing help to others. One cannot assume, for example, that every response provides an answer to a question; e.g., students with similar problems will sometimes join threads once initiated.
2. Knowing whether a piece of discussion text is a question or an answer (or neither) supports modeling of the types of discussions students engage in.

Analyzing individual messages with respect to their true *information seeking* or *information providing* roles is challenging. Standard surface-level grammatical forms are not enough to distinguish questions from answers. Surface-level features such as *wh* words such as what, where, when and how, or punctuation, such as question marks, are not sufficient. For example, some answers are commonly provided in a form of a question, e.g., "Have you checked the Nachos Manual section 4.3?", and sometimes questions are posted to provide help rather than to seek it. So the same text can play different roles depending on context.

To train the SVM model, a labeled dataset that had been constructed by human annotators was used. Questions and answers within individual messages were marked. For user roles within a thread, the annotator marked the role of each participant as *information provider* or *information seeker*. The annotation scheme was developed over three years by multiple annotators (>6) until sufficient agreement on the data was reached. The annotators shared and compared their annotations while they were developing the scheme. The data used in this work was marked by two annotators using the final annotation scheme. Table 1 shows the Kappa values for inter-annotator agreement on a data subset that consisted of 30 discussion threads with 99 messages. Kappa values were computed with independent datasets. For all categories the annotators show a high level of agreement (> 0.8). Then, a collection of feature templates was designed based on Kang et al. [3]. The features included word-based features such as uni-gram, bi-gram and tri-gram phrases, and discussion context features such as the position of current post in the thread. We also apply feature selection [3] to remove the noise and improve the performance.

Table 1. Test Set Results on Question, Answer, and Information User Role

Classifier	Precision	Recall	F-Score	Kappa
Question	0.88	0.88	0.88	0.93
Answer	0.83	0.80	0.83	0.96
Information Seeker/Provider	0.84	0.84	0.84	0.99

For this test, 240 discussion threads (904 messages) were randomly divided into two datasets: 180 discussion threads (634 messages) were used for training and 60 discussion threads (270 messages) were used for testing. Table 1 shows the model accuracy compared to the annotated target value. Results accuracy was almost 90%.

Table 2. Number of seeker/provider user roles in different settings

Role	Number initial and reply posts	SVM Classifier results for all discussion participants	SVM Classifier results for enrolled students only
Seeker	275	506	477
Provider	739	508	125

The use of SVM classifier for student dialogue roles is obvious. In an initial implementation, a simple approach assigned the initial poster the role of seeker and all reply posters the role of provider. The approach was not accurate because students commonly seek information in the middle of a discussion thread. Table 2 shows the difference between the initial approach and the SVM approach. In the last column, we show the number of information seekers and providers for only those who received a course grade, which excluded the instructor and assistants. The results clearly show how much the instructor and assistants acted as information providers.

2.2 Topic Analysis on Student Online Discussion Text

Earlier interviews with instructors indicated that instructors were quite interested in topic-related discussion assessments [2], such as the topics of questions raised in the forum and their classification using topic categories from the course syllabus. As one or our objectives was to develop an approach that could be easily applied to different courses, supervised approaches requiring a large amount of labeled data were not appropriate. And because discussion datasets are noisy we needed a model that could capture semantic meanings behind the words rather than words themselves. Latent

Dirichlet Allocation (LDA) [4] enables the capture of underlying semantic meaning without requiring large amounts labeled data, however, the original unsupervised LDA model was unsuitable because the topics learned by LDA are usually clusters of co-occurring terms that are not necessarily linked to real course topics. A semi-supervised model that could make use of course materials, such as syllabi and assignments was needed. The Labeled LDA model [5] was found to be appropriate.

The Stanford Part of Speech (POS) tagger was used first to extract nouns, since nouns in discussion sentences are the main indicators of the topics. Common words were then filtered out using a course-term dictionary that was semi-automatically generated from the words in the assignment documents. Using Labeled LDA, each topic was profiled using a bag of words model, and then labels were assigned to discussion posts according to the topic bag of words. The labels act as a prior of topic distribution and thus affect the topics learned. For experiment and illustration, the Labeled LDA model was run using ten semesters of online discussion data and course materials. Fifteen topics were extracted. Table 3 shows five of the extracted course topics and their top N term lists.

Table 3. Extracted course topics with their top N term list

Course topics	Most frequent words
Nachos Issue	function, call, line, class, type, code, nacho, thread, code, kernel,
Simulation	thread, custom, line, manager, clerk, number, switch, loop, problem,
Locks & Condition	lock, thread, condition, queue, wait, code, class, custom, variable, test,
Programming Issue	server, message, request, time, lock, system, error, code, array, char,
File System Call	file, page, swap, swap file, memory, bit, dirty, problem, size, swapfile

Although the Kappa value for agreement between two annotators was 0.96, the accuracies for the initial classifiers were low. Upon examination, several problems were found with the processing of student discussion data. First, the POS tagger did not generate correct results, especially because the system often failed to parse the noisy informal sentences that students wrote. It was also found that many irrelevant terms often misled the topic distribution process because LDA and Labeled LDA models regard each word/term in the document/thread equally when calculating the topic distribution of documents. The adoption of a domain ontology that is semi-automatically induced from a textbook glossary [6], to represent documents (discussion threads), might ameliorate these problems.

3 Assessment Workflows with Text Classification Components

3.1 Computational Workflows for Student Learning Assessment

The workflow user interface layer, or PAWS portal, is shown in Figure 1. Steps 1-4 show how the system is used to run a sample assessment workflow and how the results are accessed. In Step 1, the user selects a student assessment workflow (template). In Step 2, the user specifies the resources (datasets) that will be bound to the workflow run instance. In Step 3, the workflow instance is submitted for remote execution [2]. In Step 4, the user views the results.

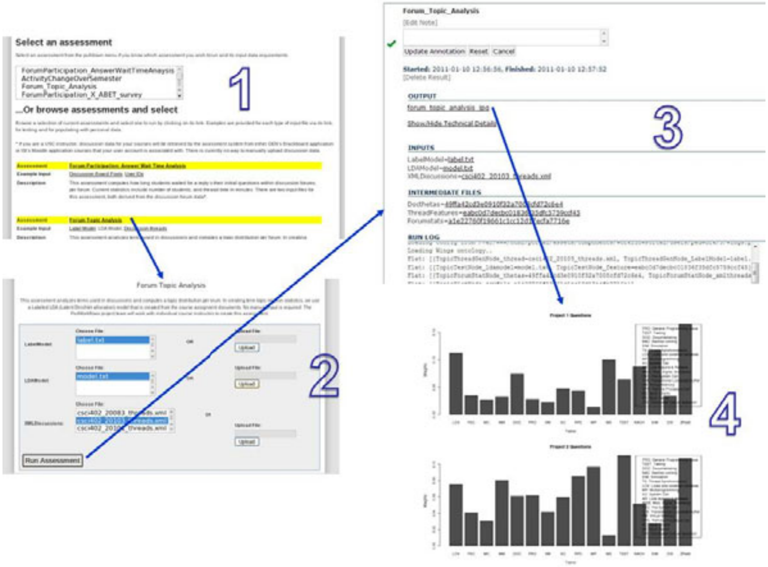

Select an assessment
ForumParticipation_AnswerWaitTimeAnaysis
ActivityChangeOverSemester
Forum_Topic_Analysis
ForumParticipation_X_ABET_survey
...Or browse assessments and select
1
Forum Topic Analysis
label.txt
model.txt
csci402_20083_threads.xml
csci402_20103_threads.xml
csci402_20103_threads.xml
Run Assessment
2
Forum_Topic_Analysis
[Edit Note]
Update Annotation
Reset
Cancel
Started: 2011-01-10 12:56:56, Finished: 2011-01-10 12:57:52
[Delete Result]
OUTPUT
forum_topic_analysis.jpg
Show/Hide Technical Details
INPUTS
LabelModel=label.txt
LDAModel=model.txt
XMLDiscussions=csci402_20103_threads.xml
INTERMEDIATE FILES
RUN LOG
3
4

Fig. 1. PAWS portal: The workflow user interface layer

3.2 Relating Information Roles with Grades

A diagram of the *Role-Grade Analysis* workflow is shown in Figure 2. There are four *components* (data processing steps, shown in yellow) in the system.

1. DiscussionClassifier: Performs the SVM Classification on discussion text. The input resources include the discussion data, trained SVM model and n-gram feature model. The output is the classified text specifying student role per thread.
2. LinkGrades: Translates the instructor's XLS grade data into an internal format and links the IDs of graded students to the IDs of discussion participants.
3. RelateRolesWithGrades: Links dialogue roles and grades. The input datasets are ClassifiedRole (output of DiscussionClassifier) and XMLUserGrades (output of LinkGrades). The ouput is the RoleGradeTable, which specifies role and grade weights.
4. MultiBoxPlot: Presents the results as multiple box plots (Figure 3).

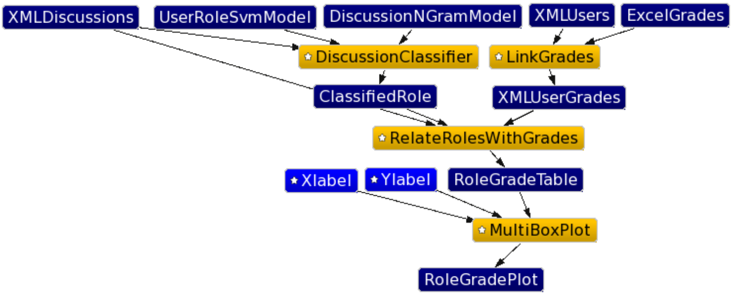


Fig. 2. Diagram of the *Role-Grade Analysis* workflow

The data flow is represented by the workflow diagram. Once the five input resources are selected by the instructor (or selected automatically by the system if there is only one matching dataset in the system, such as for the trained SVM models and n-gram feature models), a workflow instance is generated by Wings [1] and will be sent to the Pegasus [1] execution environment to run. Figure 3 shows a run result that used authentic data from an undergraduate computer science course.

The RelateRolesWithGrades component automatically rescales the grade level into five discrete levels and the BoxPlot component plots a box for each grade level. The box represents five values of the role weight distribution within the grade level: the starting point, the ¼ point, the median, the ¾ point and the end point. The level is a parameter of the workflow template so that instructors can change it for each run instance. The resulting graph shows a small trend: Students who perform better are more likely to be information seekers.

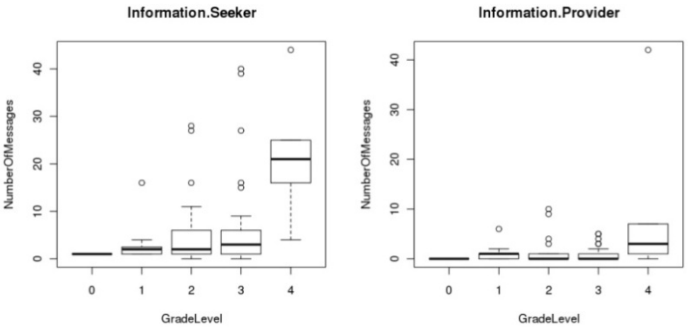


Fig. 3. Result of Role-Grade Analysis workflow

3.3 Relating Questions to Topic Categories

The second workflow integrates topic analysis and question-answer classification to identify the topic category that students ask the most questions about. This workflow directly addresses an assessment question that many instructors were interested in. The workflow consists of five components.

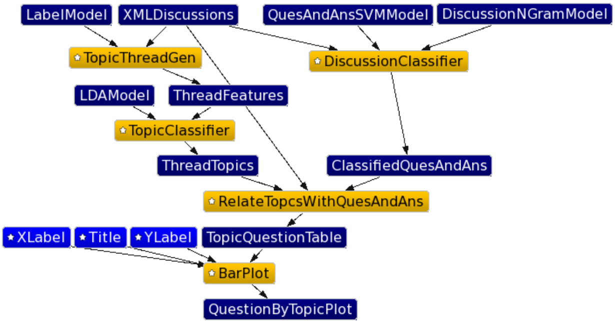


Fig. 4. Diagram of QuestionByTopic workflow

1. TopicThreadGen: Generates a feature vector for the Topic Classifier. The feature vector is the bag-of-words n-gram in the discussion text. Because an Labeled LDA Topic Model is used, it also assigns labels to each discussion thread via LabelModel.
2. DiscussionClassifier: This is the same component described in section 4.1 but performs Q/A rather than user role classification. The input SVM model is a trained Q/A SVM.
3. TopicClassifier: Determines the topic distribution given input discussion threads. The input is the trained LDA model.
4. RelateTopicsWithQuesAndAns: Links the discussion topics with discussion speech acts. Only questions raised are considered, so the output is the topic question table.
5. BarPlot: Presents the results as two bar graphs.

The resulting plots are shown in Figure 5. The top graph shows the number of questions raised in each topic category during the semester, while the bottom graph shows the number of distinct users raising questions in each topic category. The number of questions raised in each category is clearly different. During this semester, students asked questions about “programming assignment testing” and “memory management”. This graph is of great importance to instructors for assessment purposes. The accuracy of the results will improve with the accuracy of the classifiers**.**

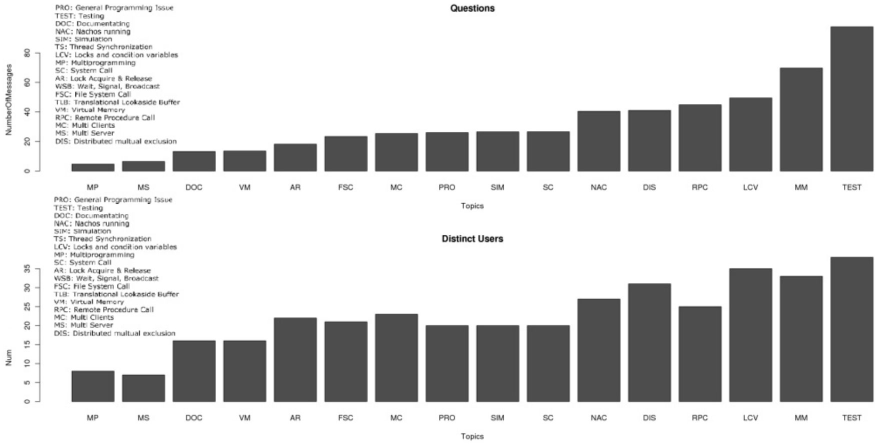


Fig. 5. Results of the QuestionByTopic workflow

4 Instructor Feedback

To collect feedback, the course instructor was given a description of the graphs and asked the following questions: 1) *Are the results understandable*?, 2) *How might you make use of the results*?, 3) *At what point during the course might it be helpful to have these results*?, and 4) *Do you have any suggestions for presenting the results*?

Regarding the role analysis, the instructor was able to understand the box plot and whiskers graphs but asked if real grades could be used instead of normalized grade levels. This would require that the instructor upload actual grades to the workflow instead of the absolute scores (0-40) used currently. The results confirmed for him that the best students were the most active and were not shy about asking questions when they had difficulties; and also that the providers understood these problems and had enough confidence to provide answers. He requested statistics about reading posts, venturing that "the top students read almost all, if not all, postings". As far as making use of the results, he said that he could inform the class of these results, although he discounted the effect it might have.

Regarding the topic-based analysis, the instructor suggested that the results would be more useful if they a) reported why students posted questions, b) the topics were constrained to individual projects. The first comment indicates that a greater context will be necessary for assessment purposes. Regarding the second, although each project is assigned its own forum, the forums were aggregated for bettering machine learning results. The workflow can be modified to process results per project (i.e., per forum), but the results should be studied to ensure that no fidelity is lost.

5 Related Work

Researchers working on non-traditional, qualitative assessment of instructional discourse include [7]. As new assessments are developed and codified, they may be readily incorporated as components into the workflow system. Longitudinal studies of student performance [8] are also relevant and might be represented as workflows to electronically track student performance across courses.

6 Summary

This paper has demonstrated a new approach to processing and analyzing student information, especially data from online discussions, for the purpose of student assessment. Combined with traditional cognitive assessment methods such as assignment and exam grades, the workflow-based approach can be powerful tool for assessing impact of online learning. The approach utilizes NLP and machine learning techniques within the context of workflow, making both processing and analysis, both efficient and robust. Handling noisy student data and modeling subject topics were found to be very challenging tasks, primarily because existing NLP tools often failed to process discussion data correctly. To reduce variance, representing data using semi-automatically induced domain terms is currently being investigated. To increase accessibility of the assessment results, a weekly report of the workflow-processed results is being sent to the instructors.

Acknowledgement

The authors thank Senior Lecturer Dr. Michael Crowley for his support. This work is supported by a U.S. National Science Foundation CISE IIS award (#0917328).

References

1. Gil, Y., Ratnakar, V., Kim, J., Gonzales-Calero, P., Groth, P., Moody, J., Deelman, E., et al.: WINGS: Intelligent Workflow-Based Design of Computational Experiments. IEEE Intelligent Systems (2010)
2. Ma, J., Shaw, E., Kim, J.: Computational workflows for assessing student learning. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 188–197. Springer, Heidelberg (2010)
3. Kang, J., Kim, J., Shaw, E.: A network analysis of student groups in threaded discussions. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 359–361. Springer, Heidelberg (2010)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. The Journal of Machine Learning Research 3, 993–1022 (2003)
5. Ramage, D., Hall, D., Nallapati, R., Manning, C.D.: Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (2009)
6. Feng, D., Kim, J., Shaw, E., Hovy, E.: Towards Modeling Threaded Discussions using Induced Ontology Knowledge. In: Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006), pp. 1289–1294 (2006)
7. 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 Int'l Conf. on Artificial Intelligence in Education (2007)
8. Reed-Rhoads, C16 – Tools for Assessing Learning in Engineering. Presentation on Inventions and Impact 2: Building Excellence in Undergraduate STEM Education (2008)

Modelling and Identifying Collaborative Situations in a Collocated Multi-display Groupware Setting

Roberto Martinez[1], James R. Wallace[2], Judy Kay[1], and Kalina Yacef[1]

[1] School of Information Technologies, University of Sydney, NSW 2006, Australia
{roberto,judy,kalina}@it.usyd.edu.au
[2] Department of Systems Design Engineering, University of Waterloo, ON, Canada
jrwallac@uwaterloo.ca

Abstract. Detecting the presence or absence of collaboration during group work is important for providing help and feedback during sessions. We propose an approach which automatically distinguishes between the times when a co-located group of learners, using a problem solving computer-based environment, is engaged in collaborative, non-collaborative or somewhat collaborative behaviour. We exploit the available data, audio and application log traces, to automatically infer useful aspects of the group collaboration and propose a set of features to code them. We then use a set of classifiers and evaluate whether their results accurately match the observations made on video-recordings. Results show up to 69.4% accuracy (depending on the classifier) and that the error rate for extreme misclassification (e.g. when a collaborative episode is classified as non-collaborative, or vice-versa) is less than 7.6%. We argue that this technique can be used to show the teacher and the learners an overview of the extent of their collaboration so they can become aware of it.

Keywords: Data Mining, Group Modelling, Collaborative Learning.

1 Introduction and Related Work

There are significant learning benefits of collaboration when students work in small groups [1]. However, in practical classroom settings, it is challenging for the teacher to be aware of the level of collaboration in each small group within their class. Emerging uses of technology offer the possibility of automatically capturing data that then can be used to detect the level of collaboration of a group. There are several ways in which such models of collaboration might be used, including mirroring information of the group to the learners and their teachers or improving the provision of adequate support in computer-supported collaborative learning systems [2]. In the latter case, these environments are sometimes designed to encourage learners to collaborate or present a structured task that forces collaboration and participation awareness. However, a general issue in applying these strategies is that different types of supportive actions can have different effects on the learning processes [3]. Specifically in collaborative learning environments, it has been shown that help is more effective if delivered just when it is needed [4]. Otherwise, well functioning groups may be distracted by unnecessary system interventions. Meanwhile, groups who experience problems and do not collaborate may benefit from such interventions.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 196–204, 2011.

The goal of our work is to explore ways to exploit readily available data to determine the level and nature of collaboration. This paper proposes an approach to infer whether group members are involved in collaborative behaviour or not. We make use of two forms of data. One is the presence of speech, based on an audio feed from each learner, without analysis of what is said. We call this a *simple audio trace*. The second source of data comes from the application log traces. From these two sources, we automatically infer key aspects of collaboration and propose a set of features to encode them. These features are then evaluated with a range of classifiers.

A given situation can be considered as "collaborative" in a learning context, if there are particular forms of interaction among the group members. For example, learning mechanisms such as explanation, negotiation, disagreement or elicitation [1]. However, even if the conditions under which these special interactions are present, there is no guarantee that learning will occur. We hypothesise that *it is possible to automatically infer whether a group of learners is engaged in a collaborative situation, from the application and audio traces of interaction with a reasonable level of accuracy.*

A number of research projects have analysed the interactions between learners to improve instructional support for collaboration using machine learning and user modelling techniques. In [5] the authors presented a fuzzy model for predicting forms of collaboration regarding the quality of the final group solution. Sequence pattern mining and clustering techniques were used to extract patterns and gain insights into the key factors that distinguish successful teams [6]. Additionally, supervised and unsupervised learning techniques have been used for grouping students according to their collaboration, assigning a value to each student to support comparison of students' behaviour [7]. The work in this paper breaks new ground as it focuses on mining patterns from simple audio and logs of interaction to match qualitative observations of the presence or absence of collaboration in a collocated group.

In the next section, we present related work. In Section 3, we introduce the collaborative learning context of our work, describing data collection and preparation. Section 4 presents our feature model, followed by the results of a number of learning approaches and we conclude with reflections and further work.

2 Context of the Study and Data Exploration

The purpose of this research study was to explore whether it is possible to infer with a reasonable level of accuracy the level of collaboration within small groups of learners. We first present the environment in which our data was collected.

Data Collection. A previous study explored the impact of alternative shared displays on group processes [8]. Data was collected from 13 groups, each with 3 students, for a total of 39 students (Figure 1, right). The participants were students predominantly enrolled in university Maths, Science or Engineering courses and aged 18-27 years. Groups were asked to perform the Job Shop Scheduling (JSS) task, an optimisation problem specifically designed for evaluating interactions within groups of learners. Participants were asked to optimise the scheduling of six *jobs*, each composed of six ordered operations. These operations require the use of six resources that can only be in use by one operation at a time. Participants modify the interface by dragging *resource pieces* into position with the shared goal of scheduling the completion of all six jobs in a minimal amount of time (see Figure 1, left).

In addition to a large, shared display projected on a nearby wall, participants were provided with laptops and external mice through which they could perform individual actions. The interface visible on the personal laptops provided a personally tailored view of the workspace, where the resources that the owner could interact with were presented as more salient than the others. The large, shared display provided an overview of the group's task progress.

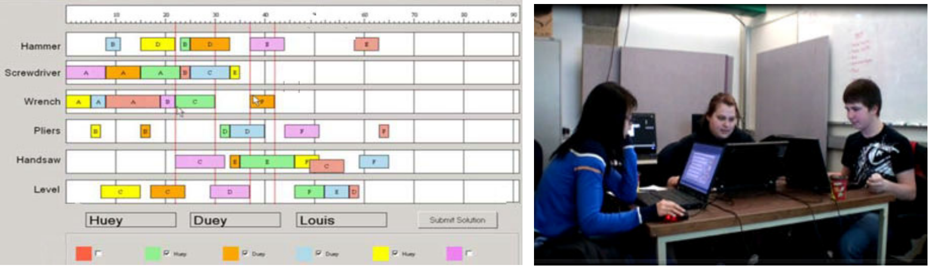


Fig. 1. Left: Application screenshot. Right: Group of students solving a problem

Each group was required to develop solutions for the JSS task 2 or 3 times. Data from 29 trials were collected and coded. Groups spent 17 minutes per trial on average and executed between 100 and 600 *physical actions* per solution, for a total of 9,800 recorded mouse click or drag operations within the JSS software. In addition to the application logs, we also transcribed verbal utterances for each trial's video recording. Each complete unit of speech in spoken language produced by a learner was considered as a verbal participation. In general, most of groups' speech was on-task. These transcripts included a total of 4,836 *verbal participations* which, combined with the physical action data, formed a dataset of more than 14,636 physical and verbal interactions (Table 1 illustrates example logs of this dataset).

Table 1. Samples from the JSS combined dataset

Verbal Participation Log				Physical Action Log			
User	Start	End	Log	User	Start	End	Log
C	15:18	15:19	I'll take care of the a's	A	01:57	01:59	- Move resource A-
C	15:21	15:22	you do the c and the d's	C	01:58	01:59	- Move resource D -
A	15:22	15:23	Yea	B	02:02	02:04	- Move resource B-

Data exploration. Before any data mining technique was performed, the data was examined to see whether any simple statistics could distinguish interesting differences between groups. Firstly, we calculated the total number of utterances, clicks and talking time for each group. Figure 2 shows the participation sequence diagrams of three sample groups. The top of each diagram shows the verbal participation and the lower parts represent the physical actions. The horizontal lines and rectangles represent actions or sets of actions (rectangles) performed by each author. The directed arrows indicate the relative sequence of the actions. From these diagrams, we observe that Group “K” was generally collaborative but participants A and C were more active. Group “L” did not have much verbal interaction, and from the diagram of physical actions, we observe they did not do much neither. Group “M” presents asymmetrical group activity: Student C has just three verbal actions, far less than the others in the group, but he performed most of the physical actions. These diagrams illustrate significant differences that exist between groups. These observations were confirmed by analysing the video recordings of the sessions.

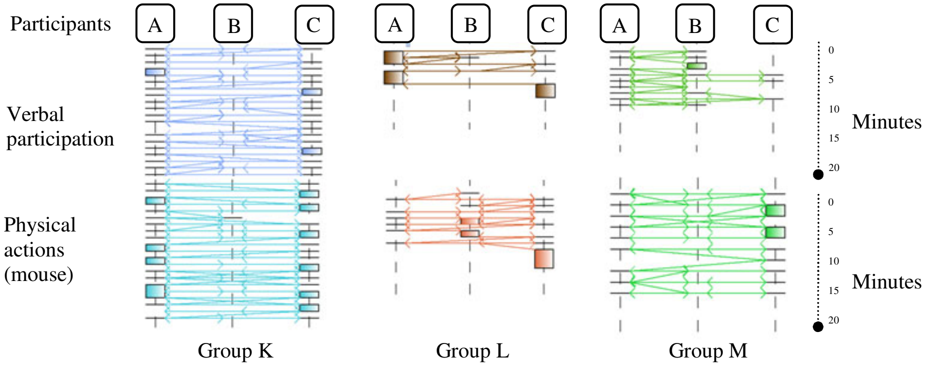


Fig. 2. Representation of the verbal and physical participation of three groups. A participative group (left), a non-communicative group (centre) and an asymmetric group (right). Diagrams created using the Process Mining Framework [9].

3 Learning Collaborative Behaviour

We now describe the rest of our approach, which, after collecting the logs of activity consists of annotating the data, constructing a set of features to learn these labels and applying different classifiers. Results are presented in the next section.

Data annotation. Dillenbourg [10] describes a situation as collaborative when participants are at the same level, can perform the same actions, have a common goal and work together. Building on these criteria, qualitative observations were made to assess whether each group was collaborating. Videos of each group’s sessions were observed. Groups’ activity was coded every 30 seconds based on the perception of collaboration for that block of time (as if a teacher was observing the group).

Each block of activity was coded as matching one of three possible values, the highest being a *collaborative moment* (C), based on Dillenbourg’s definition of collaboration [10] described above. If all participants participated to some extent or they were aware of their peers’ actions, then, that 30 seconds block of activity was tagged as “collaborative”. A moment was tagged as *somewhat collaborative* (SC) if one or two members were unaware of their peer’s actions, or if the group failed to communicate but they still tried to collaborate at some level. The last possible value, *non-collaborative moment* (NC), was assigned if the group split the task, working separately, or if just one participant did all the work. A label was assigned to each 30 seconds block of activity for each group. Most of the observations were carried out by a single observer. Two different raters, including a domain expert, tagged a sample of 15% of the sessions. Inter-rater reliability was reasonably acceptable – Cohen's k = 0.69. All groups had the same time to solve the problem (20 minutes) but they were free to decide when to stop. Figure 3 (left) depicts examples of the coding of some sessions. A row with many blue blocks (C), some in orange (SC) and few in light yellow (NC) corresponds to a collaborative sessions.

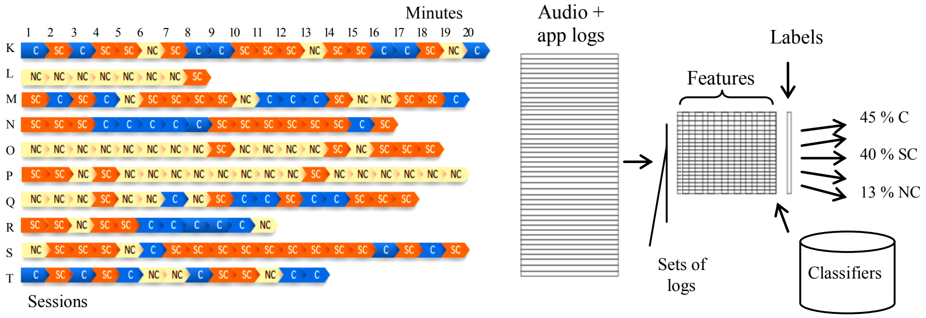

Minutes
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
K C SC C SC SC NC SC C C SC SC SC NC SC SC C C SC NC C
L NC NC NC NC NC NC NC SC
M SC C SC C NC SC SC SC SC NC C C C SC NC NC SC SC C
N SC SC SC C C C C C SC SC SC SC SC SC C SC
O NC NC NC NC NC NC NC NC SC NC NC NC NC SC NC SC SC SC
P SC SC NC SC NC NC NC NC NC NC NC NC SC NC NC NC NC NC NC
Q NC NC NC SC NC NC C NC SC C C SC C C SC SC SC
R SC SC NC SC SC C C C C C NC
S NC SC SC SC NC C SC SC SC SC SC SC SC SC SC C SC C SC
T C SC C SC C NC NC C SC SC NC C C
Sessions
Audio + app logs
Sets of logs
Features
Labels
45 % C
40 % SC
13 % NC
Classifiers

Fig. 3. Left: Dot plot representations of the coding of some analysed sessions. Each 30 seconds of group work can be tagged as "collaborative" (blue), somewhat collaborative (orange) or Non-collaborative (yellow). Diagram created using the Process Mining Framework [9]. Right: The architecture of the collaborative model.

Then, the audio and application log lines were grouped forming sets of log lines (Figure 3, right). The grouping was done using three different block sizes: 30, 60 and 90 seconds. We chose these time frame sizes based on the observations made on the videos of the sessions. In a period of 30 seconds, we can observe complete dialogues related to a solution issue so we chose it as our minimal granularity. However, the conversations can last more than 30 seconds, so we also investigated the use of longer time-frames (60 and 90 seconds). For these, the label was obtained by implementing 60 and 90 second sliding windows with steps of 30 seconds and joining the underlying labelled blocks using the following rules when the labels were not uniform across the blocks: For 60 seconds: C+SC=C, SC+NC=NC,C+NC=SC. For 90 seconds: SC+SC+C= C, C+SC+NC= SC, NC+NC+*= NC, C+C+*=C, etc. Using this process, we obtained three datasets of similar size (700 samples in average).

Feature selection. Weinberger and Fischer [11] defined that two dimensions of the collaborative learning work that can be measured quantitatively are the amount and the heterogeneity of participation. Drawing on this, a number of features were calculated for each block. We propose a feature model that includes: quantity of physical and verbal participation (features 1, 2 and 3 in Table 2), number of active participants (feature 4) and the degree of dispersion of the participation among (features 5, 6 and 7) them. In this way, we obtained three different datasets in which each instance corresponds to one block of log lines grouped in 30, 60 or 90 seconds blocks. Speech recognition was *not* used in the analysis. If there were reliable recognition of speech, this might be fed into our approach. We used the Gini coefficient as an indicator of dispersion of participation as it has been successfully

Table 2. Diagnosis features and six examples of 30 seconds blocks of collaborative (C1, C2), somewhat collaborative (SC1, SC2) and non-collaborative (NC1, NC2) activity

Feature	Metric	C1	C2	SC1	SC2	NC1	NC2
1-Physical participation	Scalar	7	12	15	15	10	15
2- Number of utterances	Scalar	28	9	4	5	0	4
3-Talking time	Seconds	19.4	17	7.5	8	0	5
4-Number of talking participants	0, 1, 2 or 3	3	3	2	3	0	1
5-Talking time dispersion	Gini coeff.	.510	.284	.747	.60	1	1
6-Verbal participation dispersion	Gini coeff.	.357	.143	.75	.614	1	1
7-Physical participation dispersion	Gini coeff.	.875	.583	.533	.40	.2	.8

used to measure equity of participation in face-to-face collaborative settings [12]. For this coefficient, a value of zero means total equality and a value of one indicates maximal inequality.

4 Evaluation

We created classification models based on the three datasets described above. We used the Best-First tree, C4.5 decision tree, Bayes-Net and naïve Bayes algorithms. Similar techniques have been successfully applied in learning contexts for detecting behaviour patterns [13]. The models were evaluated using two methodologies: 10x 10-fold Cross Validation (CV) and Leave-one-group-out CV. The 10 runs of 10-fold CV were performed on each of the 3 datasets for each algorithm. This is equivalent to breaking the data into 10 sets of same size, training on 9 of them and testing on the 10th, repeating this 10 times (folds) and repeating the whole process also 10 times. We used a standard baseline for comparing the performance of the classifiers. The *baseline classifier* simply takes account of the distribution of the frequency of the three possible label values.

We obtained results that are significantly higher than the standard baseline (Table 3). In general, even when the accuracy of the models is above our baseline we obtained sub-optimal performance with all the algorithms to predict *somewhat collaborative situations* (SC row). The training dataset formed by blocks of 30 seconds produced some of the higher performance rates across the datasets (68% for naïve Bayes, 66% for Best-First tree and Bayes-Net of F-score), and it is more balanced in the prediction of the 3 possible values. For the second dataset, we got lower rates of performance compared with the others. The third dataset produced also high rates of correct predictions (F-score above .68 for the decision trees). However, an additional metric for gaining insights on the accuracy of the models was calculated. We call it *extreme misclassifications* (EX). This measures the proportion of incorrect classifications in which the *non-collaborative* blocks were misclassified as *collaborative* and vice versa. In educational terms, a *collaborative* block misclassified as *somewhat collaborative* is still giving information about the group activity. The proportion of extreme misclassifications for the 30 seconds dataset, for all the classifiers, stayed below 7.6%; therefore the results of these models are highly acceptable. For the 90 seconds dataset, even when the accuracy levels are comparable to the first dataset, it does not perform well with the extreme misclassifications (highlighted row). The row OP (Optimistic accuracy) shows what the accuracy levels would be if only extreme misclassifications are counted as errors. Whilst this is not ideal, it shows that the classifier model is very reliable. The algorithms which produce simpler models are the decision trees. Based on these we found that the features that define the most of the classification are the number of utterances produced (at least 10 for a collaborative situation, 30 sec. dataset), low rates of verbal participation dispersion (Gini coefficient less than .40) and the absence of long periods of silence.

Table 3. Results of the 10-fold cross validation. F1=Balanced F-score, C= F-measure of the algorithm in classifying "collaborative" SC=somewhat collaborative, or NC= non-collaborative situations. EX= extreme misclassifications accuracy, OP= optimistic accuracy. BL=baseline, BNet= Bayesian Network, NB= naïve Bayes, BFT= Best-first tree, C4.5= C4.5 tree.

	Log sets of 30 seconds					Log sets of 60 seconds					Log sets of 90 seconds				
	BL	**BNet**	**NB**	**C4.5**	**BFT**	BL	BNet	NB	C4.5	BFT	BL	BNet	NB	C4.5	BFT
F1	.340	**.656**	**.683**	**.625**	**.659**	.360	.659	.668	.638	.646	.360	.666	.624	.686	.682
C	.310	**.701**	**.709**	**.652**	**.659**	.460	.827	.860	.771	.821	.340	.771	.826	.656	.587
SC	.330	**.558**	**.576**	**.630**	**.572**	.220	.178	.223	.369	.248	.430	.496	.412	.695	.724
NC	.370	**.725**	**.787**	**.739**	**.720**	.330	.771	.722	.650	.691	.250	.808	.707	.713	.737
AC		.654	**.687**	.628	**.657**		.666	.716	.648	.695		.759	.719	.746	.732
EX	.280	**.045**	**.076**	**.072**	**.057**	.340	.452	.429	.472	.510	.270	.520	.538	.640	.646
OP	.820	**.984**	**.976**	**.973**	**.981**	.790	.846	.855	.829	.819	.830	.826	.798	.799	.794

Table 4 summarises the results of the Leave-one group out CV. This analysis shows similar accuracy levels for each algorithm compared to the 10-fold CV but it also generates additional information regarding the performance of the models for each group. We noted above that the classifier algorithms produce more equilibrated results for the classification (C/SC/NC) grounding on the 30 seconds blocks dataset. However, using a Leave one out approach on this dataset, we can notice how the accuracy falls or rises depending on the group that is being tested each run. The Bayesian algorithms (at least the Bayesian network) have less oscillation in the classifications in the 30 seconds dataset (std = 9.9%) compared with the decision trees algorithms (std = 14% and 13.5%). We analysed the correlation between the proportion of *collaborative moments* and how well each model performs (accuracy). We expect not to have high correlation between the accuracy and the proportion of collaborative moments. In table 4 we can see that the negative correlation increases for the SC blocks (below -.550 for Bayes-Net and naïve Bayes in all datasets). In other words, both Bayesian algorithms are good when groups clearly behave as very collaborative or non-collaborative and decrease their power for somewhat collaborative groups. In this same respect the trees performs "better" (corr. of -.34 and -.38 for the 30 sec. dataset) but their power of prediction oscillates more across groups (higher std). We can accept the hypothesis formulated initially. It is possible to infer when a group of people is in a collaborative situation laying on the application and audio traces, taking into consideration the limitations of each algorithm. Even when our model was limited to quantitative data we could get enough information to infer if the group of learners were potentially engaged in collaborative interactions.

Table 4. Results of the leave one out cross validation

	Log sets of 30 seconds				Log sets of 60 seconds				Log sets of 90 seconds			
	BNet	NB	C4.5	**BFT**	BNet	NB	C4.5	BFT	BNet	NB	C4.5	BFT
Accuracy	.667	.688	.661	**.645**	0.660	0.694	0.656	0.673	0.648	0.642	0.605	0.606
Standard deviation	.099	.119	.140	**.135**	0.178	0.179	0.199	0.167	0.172	0.163	0.194	0.167
Correlation(C)	-0.095	-0.276	-0.206	**-0.246**	0.029	-0.143	-0.101	0.160	-0.324	-0.090	-0.298	-0.193
Correlation (SC)	**-0.61**	**-0.79**	**-0.343**	**-0.382**	**-0.623**	**-0.688**	**-0.423**	**-0.301**	**-0.695**	**-0.550**	**-0.775**	**-0.553**
Correlation (NC)	0.413	0.649	0.34	**0.391**	0.334	0.491	0.312	0.050	0.623	0.401	0.650	0.450

5 Conclusions

We presented an overview of our work to infer the extent of collaboration within groups of learners building on the foundation of collaborative learning theories [10] and data mining techniques. Our aim is to explore the intersection between the quantitative traces of peers' interactions and the research area of collaborative learning. Our approach does not take into account groups' performance. Indeed, we did not find any relationship between collaboration and this feature, obtaining a correlation of -0.052. We found that the main indicators of collaboration are the quantity and heterogeneity of verbal participation. However, the quantitative data does not tell the whole story of a group. The performance of our classifier is good enough to provide valuable information which is currently not automatically available. It would enable a teacher to see if an activity that was intended to be collaborative really was so. It would also give teachers and learners a good indication of how well each group was collaborating. The preliminary results of this study are promising and further research must be done to assess if they apply to other domains.

References

1. Stahl, G.: Collaborative learning through practices of group cognition. In: Proc.CSCL 2009, International Society of the Learning Sciences, Rhodes, Greece, pp. 33–42 (2009)
2. Soller, A., Martinez, A., Jermann, P., Muehlenbrock, M.: From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. JAIED 15(4), 261–290 (2005)
3. Hattie, J., Timperley, H.: The Power of Feedback. Review of Educational Research 77(1), 81–112 (2007)
4. Chaudhuri, S., Kumar, R., Howley, I., Rose, C.: Engaging Collaborative Learners with Helping Agents. In: Proc. AIED 2009, pp. 365–372. IOS Press, Amsterdam (2009)
5. Duque, R., Bravo, C.: A Method to Classify Collaboration in CSCL Systems. In: Adaptive and Natural Computing Algorithms, pp. 649–656. Springer, Heidelberg (2007)
6. Perera, D., Kay, J., Koprinsca, I., Yacef, K., Zaiane, O.: Clustering and Sequential Pattern Mining of Online Collaborative Learning Data. In: IEEE TKDE 2009, vol. 21, pp. 759–772 (2009)
7. Anaya, A., Boticario, J.: Application of machine learning techniques to analyse student interactions and improve the collaboration process. J. Expert Systems with Applications 38(2), 1171–1181 (2011)
8. Wallace, J., Scott, S., Stutz, T., Enns, T., Inkpen, K.: Investigating teamwork and taskwork in single-and multi-display groupware systems. Personal and Ubiquitous Computing 13(8), 569–581 (2009)

9. van Dongen, B.F., de Medeiros, A.K., Verbeek, H., Weijters, A., van der Aalst, W.M.: The ProM Framework: A New Era in Process Mining Tool Support, pp. 444–454 (2005)
10. Dillenbourg, P.: What do you mean by 'collaborative learning'? In: Collaborative Learning: Cognitive and Computational Approaches, pp. 1–19. Elsevier Science, Amsterdam (1998)
11. Weinberger, A., Fischer, F.: A framework to analyze argumentative knowledge construction in CSCL. Computers & Education 46(1), 71–95 (2006)
12. Harris, A., Rick, J., Bonnett, V., Yuill, N., Fleck, R., Marshall, P., Rogers, Y.: Around the table: are multiple-touch surfaces better than single-touch for children's collaborative interactions? In: Proc. CSCL. 2009, Rhodes, Greece, pp. 335–344 (2009)
13. Bousbia, N., Labat, J., Balla, A., Rebai, I.: Analyzing Learning Styles using Behavioral Indicators in Web based Learning Environments. In: Proc. EDM 2010 (2010)

Adapted Feedback Supported by Interactions of Blended-Learning Actors: A Proposal

Maite Martín, Ainhoa Álvarez, Isabel Fernández-Castro, and Maite Urretavizcaya

Department of Languages and Computer Systems
University of the Basque Country Apdo. 649, E-20080, Spain
{maite.martinr,ainhoa.alvarez,isabel.fernandez,
maite.urretavizcaya}@ehu.es

Abstract. SIgBLE is a general framework devoted to providing adaptable feedback for the three kinds of actors involved in a blended-learning process: teachers, students, and learning environments. Its general objectives are to automatically detect visible signs of failure or success among data coming from the actors' interactions and provide relevant feedback adapted to each situation and target actor. This paper focuses on SIgBLE's general structure and main analysis process. In addition, it presents SIgMa, a specific implementation oriented toward the teacher in the context of the MAgAdI environment, along with some evaluation results.

Keywords: interaction analysis, feedback, blended-learning.

1 Introduction

Blended-learning (b-learning) is the term used for those scenarios that combine face-to-face (F2F) and computer-mediated instruction. Most current learning experiences, such as those related to higher education, usually tend to employ this combination, where students split their work between on-line and off-line sites. In such scenarios, tight integration between traditional classrooms and on-line learning environments is needed [1], that is, the blend of such learning or teaching styles must be developed in a thoughtful way so that "*face-to-face oral and online written communication are optimally integrated such that the strengths of each are blended into a unique learning experience congruent with the context and intended learning experience*" [2]. Given this aim, we point out that the *study of interactions among all the educational actors is useful in order to discover difficulties, desires and strategies that, if suitably treated, can improve the overall learning process*. Thus, with this *working hypothesis* we propose to develop tools for studying teachers' and students' interactions arising from blended learning scenarios and provide feedback to each actor in order to promote a real synergy between both learning styles.

A main result of this hypothesis is the SIgBLE architecture, whose objective is to automate the analysis of the blended-learning actors' interactions in order to provide notices that inform them about *what is going on with the learning process*. SIgBLE has been implemented and the feedback for the teacher is currently being tested in the blended learning environment supported by the MAgAdI [3] system.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 205–212, 2011.

This paper is organized as follows: Section 2 presents the SIgBLE general proposal; Section 3 focuses on its architecture; Section 4 describes SIgMa as a specific implementation of SIgBLE coordinated with the MAgAdI learning environment; finally, in Section 5 a comparison with related work is made and some conclusions are drawn.

2 Actors and Interactions in the Current Learning Landscape

Fig. 1 shows the actors typically involved in blended environments─student(s), teacher(s) and the learning environment (LE)─, their interaction flow lines (i1, i2 and i3) and demanded feedback (f1, f2 and f3). Line i1 represents the student-environment interactions during on-line sessions; line i2 represents the teacher-environment interactions during authoring, teaching/coaching activities or inspecting processes; and line i3 represents classic and complex human face-to-face interaction (F2F).

A b-learning solution must allow for information flow between the on-line and off-line environments. That is, the results of the on-line activities are to be taken into account in the planning and development of the off-line activities, and vice versa [4]. For example, the results of on-line activities completed by the student in an LE ─such as wrong answers or solutions to exercises, questions or doubts─ should be available to the teacher in order to prepare the following off-line F2F session, whereas the results of the F2F session should be fed into the LE so they may be taken into account in subsequent on-line interactions.

Several kinds of information can be useful for improving the effectiveness of each teaching-learning actor. In a way similar to [5], we have indentified different needs for each actor and represented them by means of the feedback arrows in Fig. 1:

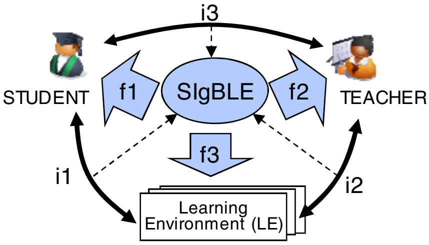


Fig. 1. SIgBLE interaction lines and feedback

- **f1** reflects the information demanded by the student regarding his or her learning progress (such as the teacher's or LE's beliefs about his or her acquired knowledge, strong and weak topics, and so forth), recommendations on the best resources to use, or even the most suitable order in which to work on a set of mandatory activities. Information to meet these needs can be derived from the student's behaviour and interactions with teachers and LEs.
- **f2** represents information that is useful for teachers so they can improve their general teaching activities, such as evaluating course content, finding activities that are more or less effective, organizing content efficiently to aid the progress of the

learner, providing personalized help or feedback for individuals or groups and planning subsequent F2F lessons. Information to meet these needs can be derived from the students' results and the success or failure of the presented activities.

- **f3** describes the information needed by the LE to adapt courses in different ways or for different types of user (individuals, groups, groups with special needs), or to provide accurate recommendations or feedback. Information to meet these needs can be derived from students' behaviour and results or even from information about F2F interactions.

However, gathering, reviewing and analyzing the huge amount of raw interaction data in order to discover meaningful information —useful for meeting the uncovered feedback needs— can be a difficult process which increases the work load of each actor considerably. Therefore, providing supporting tools to obtain and exploit the knowledge underlying the interactions among teaching-learning actors is a must.

3 SIgBLE – A General Framework

Considering the above stated current learning landscape, we propose the SIgBLE Framework (*Suggestions for ImprovinG educational aspects in a Blended Learning Environment).* Its main aim consists of *gathering* relevant data from interactions between the actors involved in the blended learning process (i1, i2 & i3 in Fig.1), *analyzing* them and *providing* pertinent feedback to each actor about ways to improve the educational experience (f1, f2 & f3 in Fig. 1). In this way SIgBLE helps the teaching-learning actors to enhance the b-learning experience without the drawback of the work load increase.

In order to define a generic proposal applicable to a wide range of learning environments, it must be based on the main areas of knowledge that are recorded in every learning environment: students (e.g. Student Model), domain topics (e.g. Domain Model), learning activities and the relationships among them (e.g. Pedagogical Domain, Strategies or Instructional Objectives). Thus, the proposed framework meets three main requisites:

- it adapts the analysis to each actor, analysis objectives and circumstances;
- it generates appropriate notices/suggestions in order to highlight critical situations regarding any of the areas of knowledge considered (i.e. Student, Learning Activity and Domain Topic);
- it analyzes and interprets each area of knowledge (i.e. Student, Learning activity and Domain Topic) by identifying data correlations and behaviour patterns.

SIgBLE provides a flexible mechanism for data analysis that can be configured in its different stages. This framework is composed of an Automatic Analysis Module (AAM), a Configuration Module (CM) and a Communication Interface, which enables it to be connected to the learning environment (LE). Interaction data to be analyzed comes from the corresponding LE knowledge bases, where the information of the blended-learning environment is recorded.

The general analysis process carried out by the AAM takes information from the knowledge bases of the three mentioned areas (Students, Domain Topics and Learning Activities) and seeks behaviour patterns on which to generate notices with relevant information; afterwards, notices with similar characteristics are clustered. In addition, this three-stage analysis process is customized by means of the Configuration Module to a desired *analysis profile* (see Analysis Profile KB section).

Fig. 2 shows an in-depth view of the general analysis process in which modules appear in light grey and data in white. It involves the three main components of the AAM which are detailed in the next sections: Data Processing Component, Notice Component and Clustering Component.

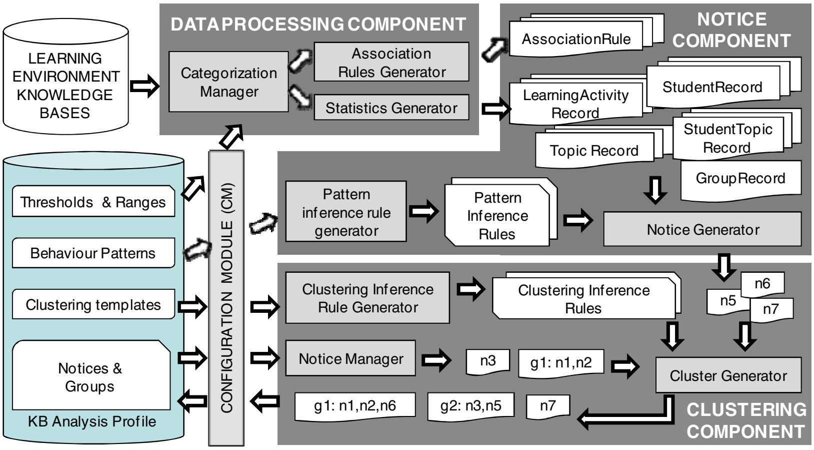
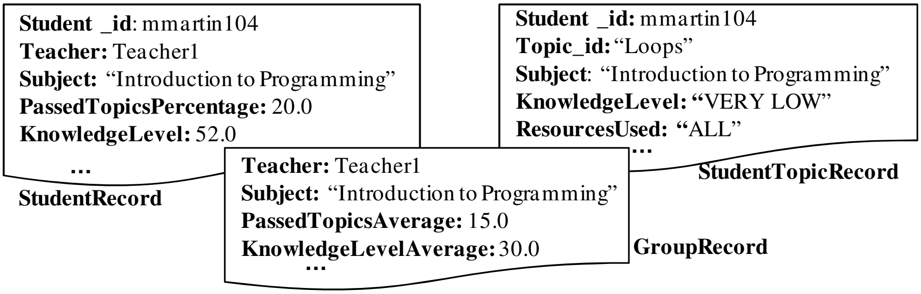


Fig. 2. Notice generation process

Data Processing Component. Its main goal is to collect and format the information from the LE Knowledge Bases in order to produce a generalized and LE-independent structure. First, the Categorization Manager collects the information about the Students, Domain Topics and Learning Activities according to the required *analysis profile*, expressed by a set of configuration parameters (*ranges* for data selecting), as indicated by the Configuration Module. Then, it cleans the abstract raw data and categorizes it (using the *analysis profile thresholds*), creating 5 types of *Records*:

- *StudentRecord* presents general information about the student learning results; it is individualized for each student (see Fig. 3, top left).
- *TopicRecord* summarizes the results of a set of students concerning a specific Domain Topic; there is one record for each topic in the domain.
- *StudentTopicRecord* sums up information about the results for one student on a Domain Topic; one record for each topic visited by each student (Fig. 3, top right).
- *LearningActivityRecord* summarizes information about the results of a set of students concerning a Learning Activity; one record for each activity.
- *GroupRecord* abstracts information about a group of students. A student belongs to a unique group-class established at the beginning of the learning process but can also be temporally grouped randomly and occasionally with other students; every group has its own record (Fig. 3, bottom).



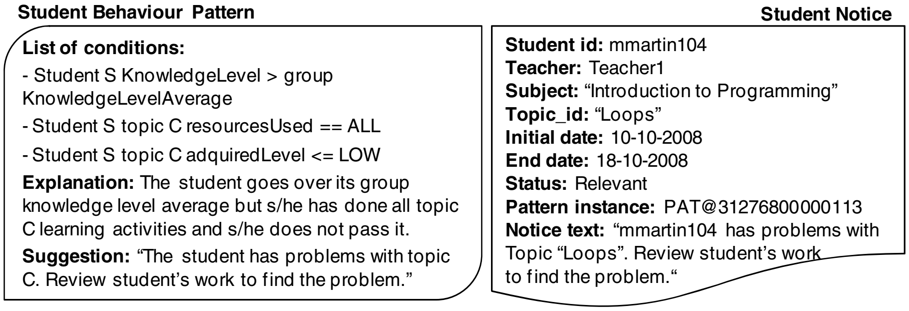
Student _id: mmartin104
Teacher: Teacher1
Subject: "Introduction to Programming"
PassedTopicsPercentage: 20.0
KnowledgeLevel: 52.0
...

Student _id: mmartin104
Topic_id: "Loops"
Subject: "Introduction to Programming"
KnowledgeLevel: "VERY LOW"
ResourcesUsed: "ALL"
...

Teacher: Teacher1
Subject: "Introduction to Programming"
PassedTopicsAverage: 15.0
KnowledgeLevelAverage: 30.0
...

Fig. 3. *Student Record*, *Student Topic Record* and *Group Record* examples

Once these records have been created, the Statistic Generator completes them with the statistical values corresponding to each type of record. Then the Association Rules Generator searches association rules describing relationships between the categorized data (more details about the process can be found in [6]).



List of conditions:
- Student S KnowledgeLevel > group KnowledgeLevelAverage
- Student S topic C resourcesUsed == ALL
- Student S topic C adquiredLevel <= LOW

Explanation: The student goes over its group knowledge level average but s/he has done all topic C learning activities and s/he does not pass it.
Suggestion: "The student has problems with topic C. Review student's work to find the problem."

Student id: mmartin104
Teacher: Teacher1
Subject: "Introduction to Programming"
Topic_id: "Loops"
Initial date: 10-10-2008
End date: 18-10-2008
Status: Relevant
Pattern instance: PAT@31276800000113
Notice text: "mmartin104 has problems with Topic "Loops". Review student's work to find the problem."

Fig. 4. Pattern model and Student notice

Notice Component. This component looks for patterns of specific behaviours exploring the *record set* created by the Data Processing Component. A *Behaviour Pattern* identifies visible signs of the learning process possibly related to its failure or success; it is composed of a set of conditions and possesses a specific meaning (see example Fig. 4 left). Behaviour Patterns are specialized for the Student, Domain Topic and Learning Activity areas. The Configuration Module selects those *behaviour patterns* to be used depending on the *analysis profile* (Fig.3, bottom left)*;* then, the Pattern Inference Rule Generator translates them into a set of operative rules. These rules, together with the *record set* and *association rules,* allow the Notice Generator to infer *notices.* A *notice* is produced when there is a set of related *records* satisfying a *pattern inference rule.* Fig. 4, left shows an example of a *student behaviour pattern* able to identify students who usually have good results but are experiencing problems acquiring a certain Domain Topic; its corresponding *pattern inference rule* applied to the records in Fig. 3 will produce the *student notice* shown on the right side of Fig. 4.

Clustering Component. This component reduces the amount of *notices* produced by means of a clustering process that gives *Notice Clusters*. Again, the Configuration Module selects the *clustering templates* belonging to the *analysis profile* in use; then, the Clustering Inference Rule Generator translates them into operative rules. These rules, together with the recently generated *notices* (e.g. n5, n6, n7 in Fig. 3) and a selection of previous *notices* and *notices clusters* (e.g. n3, g1.n1, n2 in Fig. 3) provided by the Notice Manager, allow the Clusters Generator to create new *Notice Clusters* and also add notices to those that already exist.

KB Analysis Profile. Since learning actors have their own analysis needs and preferences that might even be different depending on external circumstances or the particular point in the course the actor is in, diverse analysis profiles can be defined for each situation. They are defined in the Analysis Profile Knowledge Base; an *analysis profile* is composed of *Ranges*, *Thresholds*, *Behaviour Patterns*, *Clustering Templates* and the *Notices* and *Notices Clusters* generated during previous analysis. *Ranges* and *thresholds* values can be modified, and collections of *behaviour patterns* and *clustering templates* can be updated and extended any time. SIgBLE has a default *analysis profile* for each learning role.

4 SIgMa: SIgBLE and MAgAdI

The SIgBLE current prototype has been implemented in the context of the MAgAdI environment using Java, the Jess rule engine, WEKA and mathematical libraries. This has given rise to the SIgMa system which at the moment focuses only on the teachers' area (Fig. 1, f2 arrow). MAgAdI is a learning environment designed to be the technological component for a b-learning solution. It provides three related workspaces, one for each teaching-learning role, and a shared background composed of several data bases: student, domain and pedagogical.

Following Fig. 1, data from student-MAgAdI interactions (i1) are collected when on-line learning sessions are in progress (e.g. the student performs learning activities). Information about i2 is gathered during the authoring and reviewing sessions, i.e. when modifications to the student model are being carried out. Finally, the relevant information regarding i3 is recorded directly and explicitly by teachers. All interaction data is stored in the MAgAdI MySQL databases.

The current SIgMa prototype provides a default Teacher *Profile* with 20 defined *behaviour patterns*: 4 about domain topics, 5 about students and 11 about learning activities. The Communication Interface allows teachers to visualize the generated notices and also the information about their students, such as their learning progresses, on-line behaviours, and the comparative study of students' results.

The SIgMa & MAgAdI learning environment is currently being used and tested at the University of the Basque Country. Two main studies have been undertaken with Computer Science undergraduate students. The first one involved 12 students during the first semester of 2010. Students used MAgAdI to work on 16 activities covering 4 topics from "Introduction to Programming". The student-MAgAdI interaction data was analyzed by SIgMa at the end of the semester. The analysis process resulted in 36 *notices* before the clustering stage –5 notices on learning activities, 4 notices on domain topics and 27 notices on students. Afterwards, 8 *notice clusters* were created and 8 *notices* were not clustered. The notices generated were manually verified according to the information in the MAgAdI DB with satisfactory results.

The second study began last December and will be carried out during the 2010-2011 academic year for the subject “Data Base Development”. This study centres on the use of SIgMa by teachers in their day-to-day work. So far, 14 students have been working on 4 topics within 21 learning activities (more activities will be added). In the very first analysis (January 2011) we obtained 15 notices before the clustering stage –9 notices on learning activities, 4 notices on domain topics and 2 notices on students. After that, 1 notice cluster was created and 10 notices remained independent. This preliminary result has been evaluated with the subject teacher, who has remarked on the interest and utility of the generated *notices* with encouraging results.

5 Discussion and Conclusions

This paper presented some results from our two studies, with the aim of developing tools to improve the learning process in blended-learning environments, taking as a primary basis the set of interactions among all the learning actors. Thus, the general SIgBLE architecture was defined, implemented and tested; it is able to collect and interpret those interactions in light of a personalized perspective and generate relevant feedback according to the target receiver. It is able to detect specific learning situations from the actors' interaction data. Personalization in the data analysis process is achieved in all its different analysis stages according to configurable profiles by means of parameterization, behaviour patterns and clustering templates.

The aim of helping teachers to know *what their students are doing* is shared by several other systems. For example, the system developed by Zaiane & Luo [7] applies several data mining techniques to discover potentially useful patterns in web access logs; LOCO-Analyst [8] is a Semantic Web application that provides feedback to the teacher based just on the learning context without parameterization options.

Additionally, Teacher Advisor [9] applies fuzzy techniques to analyze student interactions with the learning course in order to recognise situations where students may need feedback. Classroom Sentinel [10] is a web service that provides teachers with timely and fine grained patterns of students behaviour in classrooms. Both systems focus on helping teachers with their day-to-day activities by sending them alerts when certain predefined situations are detected.

Unlike Zaiane & Luo's system, in which the teacher has to control and guide the analysis, or LOCO-Analyst where the teacher has to find useful information among the analysis results, SIgBLE automatically analyzes its results to offer concrete suggestions and explanations. Moreover,, our framework involves all learning actors in such a way that all of them receive suggestions and explanations according to their role. In addition, SIgBLE proposes a personalization mechanism based on a dynamic and extensible Profile Knowledge Base and a Configuration Module. Thus, a *profile* includes parameterization variables, behaviour patterns and clustering templates, which can be modified at any time, providing the framework with a high flexibility not available in other systems – see [9] and [10].

SIgMa is the current and partial implementation of SIgBLE for the teachers’ area, the implementation for the other two areas is in its initial stage. The teachers’ area is being tested in a real learning context at the University of the Basque Country. Currently, the Automatic Analysis is completely implemented and a teacher-oriented interface that allows them to visualize notices as well as student models and activities results has been provided. The default analysis profile used on SIgMa has been defined empirically taking into account the desires and objectives of three voluntary teacher participants combined with results from other studies as [11]. Testing experiences of SIgMa have validated our proposal, and the amount and correctness of the notices generated together with the interest awakened in the teachers involved support the validity of the analysis results. Therefore, during next months the experimental use of SIgMa will be widened to new subjects, teachers and students.

Acknowledgements

This work has been partially supported by: Basque Gov. UE09+/32, Spanish MEC TIN2009-14380, and DFG 157/2009. The authors thank the EHU students and teachers involved in SIgMa MAGADI testing.

References

1. Graham, C.R.: Blended Learning Systems: Definition, Current Trends, and Future Directions. In: Bonk, C.J., Graham, C.R., Cross, J., Moore, M.G. (eds.) The Handbook of Blended Learning: Global Perspectives, Local Designs, pp. 3–21. Wiley and Sons, Chichester (2006)
2. Garrison, D.R., Vaughan, N.D.: Blended Learning in Higher Education. In: Jossey-Bass, ed. Framework, Principles, and Guidelines (2008)
3. Álvarez, A.: MAgAdI, a proposal for a multi-agent adaptive framework for blended learning. PhD. Thesis, University of the Basque Country (2010)
4. Howard, L., Remenyi, Z., Pap, G.: Adaptive Blended Learning Environments. In: Int. Conf. on Engineering Education (ICEE 2006), San Juan, Puerto Rico, pp. T3K 11–16 (2006)
5. Romero, C., Ventura, S.: Educational data mining: A survey from 1995 to 2005. Expert Syst. Appl. 33(1), 135–146 (2007)
6. Martín, M., Álvarez, A., Fernández-Castro, I., Urretavizcaya, M.: Generating teacher adapted suggestions for improving distance educational systems with Sigma. In: Int. Conf. on Advanced Learning Technologies, Spain, pp. 449–453. IEEE, Santander (2008)
7. Zaıane, O.R., Luo, J.: Towards Evaluating Learners' Behaviour in a Web-Based Distance Learning Environment. In: IEEE Int'l Conference on Advanced Learning Technologies, Madison, USA, pp. 357–360 (2001)
8. Jovanović, J., Gašević, D., Brooks, C.H., Devedžić, V., Hatala, M.: LOCO-Analyst: A Tool for Raising Teachers' Awareness in Online Learning Environments. In: Duval, E., Klamma, R., Wolpers, M. (eds.) EC-TEL 2007. LNCS, vol. 4753, pp. 112–126. Springer, Heidelberg (2007)
9. Kosba, E., Dimitrova, V., Boyle, R.: The evaluation of an Intelligent Teacher Advisor for Web Distance Environments. In: Artificial Intelligence in Education (AIED), The Netherlands, pp. 370–377. IOS Press, Amsterdam (2005)
10. Singley, M.K., Lam, R.B.: The Classroom Sentinel: Supporting Data-Driven Decision-Making in the Classroom. In: 3rd WWW Conference, Chiba, Japan, pp. 315–322 (2005)
11. Zinn, C., Scheuer, O.: Getting to know your student in distance learning contexts. In: Nejdl, W., Tochtermann, K. (eds.) EC-TEL 2006. LNCS, vol. 4227, pp. 437–451. Springer, Heidelberg (2006)

Learning by Teaching SimStudent – An Initial Classroom Baseline Study Comparing with Cognitive Tutor

Noboru Matsuda[1], Evelyn Yarzebinski[2], Victoria Keiser[1], Rohan Raizada[1], Gabriel J. Stylianides[3], William W. Cohen[1], and Kenneth R. Koedinger[1]

[1] School of Computer Science, Carnegie Mellon University
5000 Forbes St. Pittsburgh PA 15213 USA
[2] University of Pittsburgh
[3] Department of Education, University of Oxford

Abstract. This paper describes an application of a machine-learning agent, *SimStudent*, as a teachable peer learner that allows a student to learn by teaching. SimStudent has been integrated into APLUS (Artificial Peer Learning environment Using SimStudent), an on-line game-like learning environment. The first classroom study was conducted in local public high schools to test the effectiveness of APLUS for learning linear algebra equations. In the study, learning by teaching (i.e., APLUS) was compared with learning by tutored-problem solving (i.e., Cognitive Tutor). The results show that the prior knowledge has a strong influence on tutor learning – for students with insufficient training on the target problems, learning by teaching may have limited benefits compared to learning by tutored problem solving. It was also found that students often use inappropriate problems to tutor SimStudent that did not effectively facilitate the tutor learning.

Keywords: Learning by teaching, teachable agent, SimStudent, machine learning, inductive logic programming.

1 Introduction

The goal of our current project is to investigate cognitive and social theories of the *effect of tutor learning* [1]. Although it is well known that students learn when they teach others, little is known about the underlying cognitive principles. Part of the difficulties of studying the effect of tutor learning is its cost and human factors. For example, to conduct an empirical study on learning by teaching in an authentic classroom setting, students must switch their role (tutor vs. tutee). To overcome this challenge, we developed a synthetic pedagogical agent (called *SimStudent*) that acts as a peer learner [2]. We then developed an on-line game-like learning environment (called APLUS) where students learn algebra equations by teaching SimStudent.

The aim of this paper is to first introduce SimStudent and APLUS. The paper then describes a classroom study, in which the effectiveness of learning by teaching SimStudent was evaluated by comparing APLUS/SimStudent with Cognitive Tutor.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 213–221, 2011.

2 SimStudent and the APLUS Learning Environment

2.1 SimStudent: A Synthetic Peer Learner

SimStudent learns procedural skills from examples. In the current context, individual students *interactively* tutor SimStudent. Namely, the examples are given as a combination of feedback and hint in the context of tutored-problem solving.

When tutoring SimStudent, the student poses a problem for SimStudent to solve. SimStudent then attempts to perform one step at a time and asks the student about its correctness. If the student provides negative feedback, SimStudent attempts an alternate action. If SimStudent cannot perform a step "correctly," it asks the student for a hint. The student then *demonstrates* the step as a hint. SimStudent inductively generalizes the examples using *background knowledge*, and generates a set of production rules that represent learned skills.

One of the unique characteristics of SimStudent as a teachable agent is its ability to *model human learning*. We are particularly interested in modeling errors that human students make from inappropriate inductions [3]. We hypothesize that students make such errors when they rely on shallow problem solving features instead of domain principles. One example is to identify '3' in '3x' as a *number* instead of a *coefficient*, as when students "divide both sides by 3 for $3x = 6$." A student who perceives such a shallow feature would likely divide both sides of $3/x = 6$ by 3 as well, which is one of the most frequently observed student errors. To model this type of learning, we modified SimStudent's background knowledge by dropping the concept of coefficient and adding more perceptually grounded background knowledge (e.g., "get a number before a variable"). This particular functionality provides us with the opportunity to investigate the impact of differences in the tutee's competency during tutor learning.

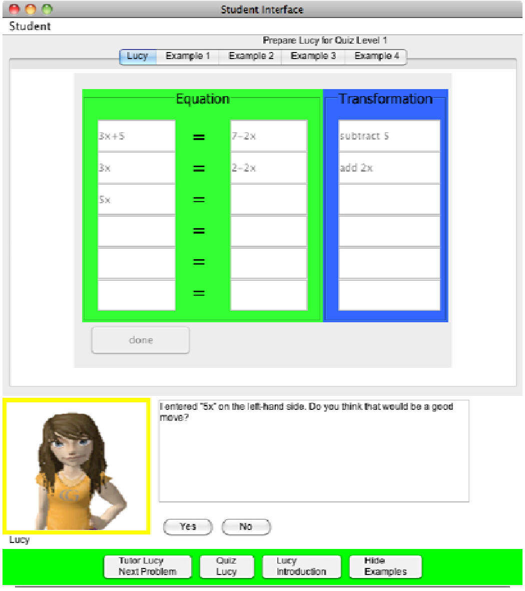
2.2 APLUS: An On-line Learning by Teaching Environment

SimStudent is embedded into an online, game-like learning environment, called APLUS (Artificial Peer Learning environment Using SimStudent). Fig. 1 shows a screenshot of APLUS. SimStudent is visualized at the lower left corner and, in this example, is named Lucy. There is a *Tutoring Interface* taken from a Cognitive Tutor that allows the student and SimStudent to collaboratively solve problems. In the figure, SimStudent entered "5x" and is asking the student if it is correct or not. The student responds by clicking on the [Yes/No] button. Because the student is also learning how to solve equations, he/she may get stuck. In such a situation, students are encouraged to review examples provided in the [Example *n*] tab shown on top of the screen.

The student is told that his/her goal is to have Lucy pass the quiz. When the student clicks on the [Quiz Lucy] button, Lucy takes a quiz without feedback from the student. The summary of the quiz results appears in a separate window (see Fig. 2).

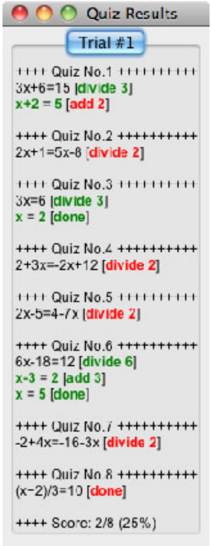
3 Related Works on Teachable Agent

Using a pedagogical agent as a peer learner to study the effect of tutor learning is not a new idea [4-8]. Such a pedagogical agent is often called a *teachable agent*.



Equation			Transformation
3x+5	=	7−2x	subtract 5
3x	=	2−2x	add 2x
5x	=		
	=		
	=		
	=		

I entered "5x" on the left-hand side. Do you think that would be a good move?



```
++++ Quiz No.1 ++++++++++
3x+6=15 [divide 3]
x+2 = 5 [add 2]

++++ Quiz No.2 ++++++++++
2x+1=5x-8 [divide 2]

++++ Quiz No.3 ++++++++++
3x=6 [divide 3]
x = 2 [done]

++++ Quiz No.4 ++++++++++
2+3x=-2x+12 [divide 2]

++++ Quiz No.5 ++++++++++
2x-5=4-7x [divide 2]

++++ Quiz No.6 ++++++++++
6x-18=12 [divide 6]
x-3 = 2 [add 3]
x = 5 [done]

++++ Quiz No.7 ++++++++++
-2+4x=-16-3x [divide 2]

++++ Quiz No.8 ++++++++++
(x−2)/3=10 [done]

++++ Score: 2/8 (25%)
```

Fig. 1. A screenshot of APLUS – Artificial Peer Learning environment Using SimStudent. There are four examples available in the [Example] tabs to review. The [Quiz Lucy] button is to initiate a quiz. In this figure, Lucy just entered "5*x*" on the left-hand side and asked the student for feedback on the correctness.

Fig. 2. A summary of the quiz. SimStudent is pre-trained with inappropriate background knowledge hence may make some common human errors such as adding 2 to *x*+2=5.

Some teachable agents employ a machine-learning algorithm to carry out genuine learning [5], whereas other systems do not require sophisticated machine-learning modules [4, 6, 7]. For some agents, the knowledge is directly transferred from the student to the teachable agent using a shared knowledge representation [4, 6, 7]. Conversely, other agents rely on an indirect knowledge transfer, meaning the agent's knowledge is not visible to the student [5]. Some teachable agents are capable of interacting with the student in a similar way as humans do – learning by tutored-problem solving [5, 8]. Other teachable agents "learn" knowledge in a different mode than students do, which somewhat limits the tutoring interactions [4].

SimStudent deploys inductive logic programming to interactively learn problem-solving skills. Because of this genuine learning mechanism, SimStudent may learn incorrect rules much like human students do (details follow in the next section) when its background knowledge is controlled as described in Section 2.1. Since the skills learned by SimStudent are not directly visible, the student must reason about the competency of SimStudent from its behavior. These characteristics provide more ecological validity for our purposes of studying theories of learning by teaching, bacause the students are able to teach SimStudent interactively in the same way as they teach their friends.

4 Evaluation Study

4.1 Methods

Two high schools in a rural area near Pittsburgh, PA, participated in the study under the supervision of Pittsburgh Science of Learning Center (www.learnlab.org). In both schools, the Algebra I Cognitive Tutor [9] (referred to as the Cognitive Tutor or CT hereafter) is intensively used. There are two Algebra I classes in one school (N=40), and two Algebra I (N=30) and two Algebra II (N=34) classes in another school. A total of 104 students participated from the six algebra classes.

The study was a randomized control trial in each class. We used the Cognitive Tutor as a control condition. Since its effectiveness is well known [9], we conjectured that this comparison would provide a good sense of the effectiveness of learning by teaching relative to tutored-problem solving. We targeted a unit of the Cognitive Tutor where students learn to solve *equations with variables on both sides*. The students in the experimental condition (the SimStudent condition, or SS, for short) were asked to tutor Lucy equations with variables on both sides. The quiz for Lucy was designed to measure the competency at this level.

There were two days for the intervention where students used either APLUS or Cognitive Tutor for one full class period (40 minutes with an exception of 54 minutes in one Algebra I class). The students' and SimStudents' activities during the intervention were all logged automatically by the software, including problems tutored, feedback provided, steps performed, examples reviewed, hints requested, and quiz attempts. The expert-model module taken from the Cognitive Tutor was embedded into APLUS, but was only used to automatically assess the student's and SimStudent's actions for the logging purposes (i.e., the students did not receive any feedback on the correctness of their tutoring activities).

Pre- and post-tests were performed immediately before and after the intervention to measure students' competency in algebra equation solving as described in the next section.

4.2 Tests

Three versions of isomorphic online tests were used to counterbalance the pre- and post-test.[1] All three versions showed decent reliability scores; Cronbach's alpha for Test A = 0.83, B = 0.76, and C = 0.84.

The test has five subsections: (1) six equation-solving items (EQ) – students were asked to show their work on a piece of paper. (2) Effective next step (EFFECT) – 12 yes/no multiple-choice items to identify if a given operation is appropriate for a given equation. (3) Demonstration of errors (DEMO) – five items with a mixture of multiple-choice and free response to identify and explain an incorrect step in an *incorrect* solution for a given equation. (4) 38 yes/no multiple-choice items. Identify constant and/or variable terms in given expressions, and identify if two given expressions are like terms. (5) Equivalent expressions – 10 yes/no multiple-choice items to identify if a pair of expressions are equal.

[1] There was a delayed-test implemented 2 weeks after the intervention hence the three versions. We have yet to analyze the delayed test scores, thus their exclusion from this paper.

In the following analysis, average scores of subsections are used. We also used the overall score, which is the average of the five subsection averages.

4.3 Results

Although the total of 104 students participated to the study, only 74 and 79 students took the pre- and post-test respectively. Of those, only 57 students took both pre- and post-tests. In this section, we only include students who took both pre- and post-test. However, we further excluded five students from the analysis for apparent patterns of "gaming" on the EQ section (e.g., entering a sequence of numbers as the answers for the six equation items). As a consequence, 52 students (25 in SS and 27 in CT) were included in the current analysis.

4.3.1 Overall Test Scores

We first ran a regression analysis predicting the post-test score with the condition as a fixed factor and the pre-test score as a covariate. The adjusted post-test scores for each condition are CT = 0.44 + 0.37 * pre-test, and SS = 0.27 + 0.56 * pre-test. The condition is not the main effect on the adjusted mean post-test score: SS = 0.65 vs. CT = 0.67; F (1, 50) = 0.97, $p<0.34$.

4.3.2 Individual Section Scores

A mixed-design ANOVA revealed that there is a trend of interaction between the condition and the test on the EQ score; F(1, 50) = 3.48, $p<0.07$. There is a trend of a main effect on the test only for the CT condition: pre-test EQ mean = 0.54, post-test EQ mean = 0.64; t=1.71, $p<0.1$.

There is a significant aptitude-treatment interaction on the EFFECT score, F(1, 48)=4.43, $p<.05$, although the test is not a main effect. The centered polynomial regression on the EFFECT score shown in Fig. **3** confirmed that the difference between the condition intercepts is not significant.[2] This implies that *the students who scored on and above the average at the pre-test EFFECT score performed equally well on the post-test EFFECT score, regardless of the type of intervention (CT vs. SS), but those who scored below the average on the pre-test EFFECT score performed worse on the post-test EFFECT when assigned to the learning by teaching condition.*

For all other subsections, there was no significant condition effect or the test effect.

4.3.3 Learning Curve

How did students improve the accuracy in applying knowledge components during the intervention? To answer this question, we analyzed the learning curves.

The knowledge component used for this analysis was determined based on the features of the equation. There are four knowledge components defined: addsub-pos, addsub-neg, muldiv-pos, and muldiv-neg. They represent skills for adding/subtracting a term to/from both sides, and multiplying/dividing both sides with/by a term. The postfix, pos and neg, is determined by *a term in a given equation* that motivated a

[2] There are two apparent outliers, one at the top left corner and one at the lower right corner). Their absolute z-scores are higher than three; hence it is probable that they are outliers. When these two points are removed, however, the significance of the interaction remains intact.

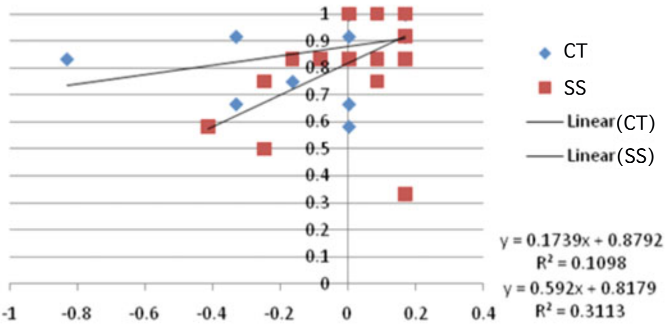


Fig. 3. A centered polynomial regression on the EFFECT score. The covariate (pre-test score) is normalized as a difference from the population mean.

particular operation. For example, "subtracting 2 from both sides of 3*x*+2=5*x*-1" is coded as addsub-pos, because it is arguably the *positive* term "+2" on the left-hand side that triggered this operation.

To compute a learning curve, we first coded the individual student's accuracy in applying knowledge components. For the Cognitive Tutor condition, for each step in solving an equation, the correctness of the application of a knowledge component is coded (correct or incorrect) based on the *first attempt* at a step. For the SimStudent condition, the correctness of the application of a knowledge component is coded based on the correctness of the student's feedback on a step performed by SimStudent or the correctness of the step performed by the student as a hint.

The learning curves shown in Fig. 4 are plotted by using the following regression model: $p_{iT} = \alpha + \beta K_i + \gamma K_T * T$ where p_{iT} represents the probability of making an *error* to apply knowledge component K_i on the T^{th} opportunity to apply K_i.

The significant decline of the probability of making errors shows that the students in both conditions improved the accuracy of applying knowledge components over

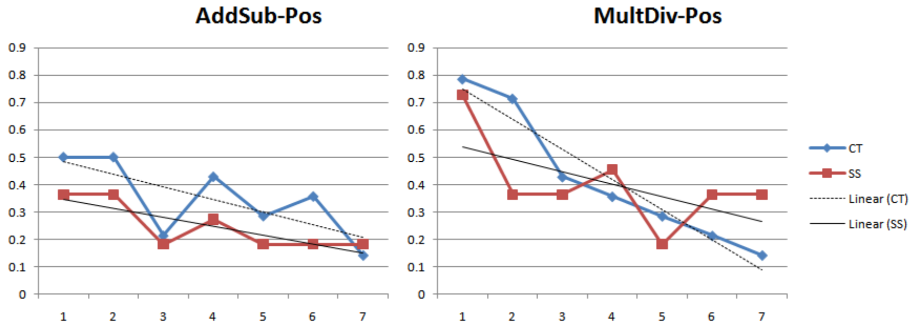


Fig. 4. A learning curve of student performance during the tutoring session. The x-axis shows the number of times the knowledge component was practiced to apply. The y-axis shows the probability making an *error* to apply the knowledge component.

time. This observation implies that the students could show better performance on the post-test than the pre-test, especially on the procedural test items (i.e., EQ, EFFECT, and DEMO; see 4.2). Yet, the data do not confirm such an improvement. The next section provides one hypothesis on this issue for the SimStudent condition.

4.3.4 Problems Used for Tutoring

For the SimStudent condition, one possible account for not seeing improvement on the test scores, even though the learning curves indicate improvement on the accuracy of applying knowledge components, is what might be called a *biased rehearsal effect*. Namely, *when tutoring SimStudent, the students might have repeatedly used only similar and perhaps "easy" problems, thus the overall accuracy of solving the problems improved as the tutoring session advanced.*

To test the hypothesis of this biased rehearsal effect, we analyzed the session log data to categorize the type of problems the students used for tutoring. There were 108 (17%) one-step equations, 259 (41%) two-step equations, and 256 (40%) equations with variable on both sides. A dominating number of problems (58%) were either one-step or two-step equations, which are not typical of the target for this current study. This observation provides positive support for the biased rehearsal effect.

5 Discussion

The current study is the first classroom experiment using APLUS and SimStudent as a teachable agent. Thus, the technical immaturity in the system may have interfered with the results. Despite intensive pilot and usability studies, there were still technical glitches observed during the study. Since the study, we have been running more usability tests and the system has been revised. In a more recent classroom study where the effect of self-explanation for tutor learning was tested, the students in the control condition that used the same APLUS system as in the current study showed significant improvement from pre- to post-test (will be reported elsewhere).

The significant aptitude-treatment interaction (ATI) on the EFFECT score (to identify an effective next step for a given equation) is a particularly important finding. The impact of learner's readiness for learning is one of the central issues in the sciences of learning [10]. The current study suggests that the prior knowledge significantly influences the effect of *tutor learning*, and (more importantly) that when the student is not trained beyond a certain threshold, he/she might receive more benefit from tutored-problem solving than learning by teaching. This makes sense because no one would likely be able to teach without a certain amount of knowledge about the subject. The current study provides an indication of what that threshold might be and opens the space for further investigation on this area.

Despite a favorable trend on the effect of the teachable agent for tutor learning [1], the current study did not confirm such an effect. One potential account for the lack of reproduction of the positive effect is the difference in domains. Betty's Brain is an example of a recent study that showed a positive effect of tutor learning, but that target knowledge is a declarative causal knowledge. The current study investigated the tutor learning effect in algebra equation solving that has more *procedural* skills involved in nature. Notably, Walker et al. [11] found no significant benefit of peer tutoring (within students alternating between tutoring and being tutoring) over being tutored in a procedural domain. The presence of the ATI suggests that there might be more requirements for prior knowledge when teaching procedural skills than when teaching declarative knowledge. If so, then, *it would be worth testing the hypothesis that learning by teaching has more effect for declarative domains than procedural domains*.

Our hypothesis concerning the biased rehearsal effect must also be tested. One way to avoid the bias on the problem selection is to have SimStudent express boredom on solving too many similar problems. Another idea is to embed a meta-tutor into APLUS to provide feedback on student's problem selection. In our upcoming new study, a bank of problems is available for students to review. The problem bank shows a wide variety of problem types used by the students in the previous studies with difficulty levels reflecting a ratio of successful vs. unsuccessful attempts made by SimStudents.

Acknowledgement

The research reported here was supported by National Science Foundation Award No. DRL-0910176 and the Institute of Education Sciences, U.S. Department of Education, through Grant R305A090519 to Carnegie Mellon University. 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.

References

1. Roscoe, R.D., Chi, M.T.H.: Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors' explanations and questions. Review of Educational Research 77(4), 534–574 (2007)
2. Matsuda, N., Keiser, V., Raizada, R., Tu, A., Stylianides, G., Cohen, W.W., Koedinger, K.R.: Learning by Teaching SimStudent: Technical Accomplishments and an Initial Use with Students. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 317–326. Springer, Heidelberg (2010)
3. Matsuda, N., et al.: A Computational Model of How Learner Errors Arise from Weak Prior Knowledge. In: Taatgen, N., van Rijn, H. (eds.) Proceedings of the Annual Conference of the Cognitive Science Society, pp. 1288–1293. Cognitive Science Society, Austin (2009)
4. Biswas, G., et al.: Learning by teaching: a new agent paradigm for educational software. Journal Applied Artificial Intelligence 19(3&4), 363–392 (2005)
5. Michie, D., Paterson, A., Hayes, J.E.: Learning by teaching. In: Proc. of Second Scandinabian Conference on Artificial Intelligence, Tampere, Finland, pp. 413–436 (1989)
6. Bredeweg, B., et al.: DynaLearn - Engaging and Informed Tools for Learning Conceptual System Knowledge. In: Azevedo, P.R.R., Biswas, G. (eds.) AAAI Fall Symposium, Cognitive and Metacognitive Educational Systems (MCES 2009), pp. 46–51. AAAI Press, Arlington (2009)
7. Nichols, D.M.: Intelligent Student Systems: an Application of Viewpoints to Intelligent Learning Environments. Lancaster University, Lancaster (1993)

8. Palthepu, S., Greer, J., McCalla, G.: Learning by teaching. In: Proceedings of the International Conference on Learning Sciences, Illinois (1991)
9. Ritter, S., et al.: Cognitive tutor: Applied research in mathematics education. Psychonomic Bulletin & Review 14(2), 249–255 (2007)
10. Koedinger, K.R., Aleven, V.: Exploring the Assistance Dilemma in Experiments with Cognitive Tutors. Educational Psychology Review 19(3), 239–264 (2007)
11. Walker, E., Rummel, N., Koedinger, K.R.: To Tutor the Tutor: Adaptive Domain Support for Peer Tutoring. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S., et al. (eds.) ITS 2008. LNCS, vol. 5091, pp. 626–635. Springer, Heidelberg (2008)

When Is It Best to Learn with All Worked Examples?

Bruce M. McLaren and Seiji Isotani

Carnegie Mellon University, Human Computer Interaction Institute,
5000 Forbes Avenue, Pittsburgh, Pennsylvania
{bmclaren,sisotani}@cs.cmu.edu

Abstract. Worked examples have repeatedly demonstrated learning benefits in a range of studies, particularly with low prior knowledge students and when the examples are presented in alternating fashion with problems to solve. Recently, worked examples alternating with intelligently-tutored problems have been shown to provide at least as much learning benefit to students as all tutored problems, with the advantage of taking significantly less learning time (i.e., more efficiency) than all tutored problems. Given prior findings, together with the prevailing belief that students should be prompted to actively solve problems after studying examples, rarely have *all* worked examples been tried as a learning intervention. To test the conventional wisdom, as well as to explore an understudied approach, a study was conducted with 145 high school students in the domain of chemistry to compare alternating worked examples / tutored problems, all tutored problems, and all worked examples. It was hypothesized that the alternating condition would lead to better results (i.e., better learning and/or learning efficiency) than either all examples or all tutored problems. However, the hypothesis was not confirmed: While all three conditions learned roughly the same amount, the all worked examples condition took significantly less time and was a more efficient learning treatment than either alternating examples/tutored problems or all tutored problems. This paper posits an explanation for why this (seemingly) surprising result was found.

Keywords: worked examples, intelligent tutors.

1 Introduction

The learning benefits of worked examples have been thoroughly researched and well documented [1]. A key theoretical reason often cited for the benefits of worked examples is cognitive load theory [2]. In particular, compared to problem solving, worked examples are believed to lessen *extraneous* load, which refers to the use of cognitive resources for mental processes, such as search. While search methods such as means-ends analysis are often critical to solving problems, such approaches exhaust the cognitive resources of students that could be used for learning. By providing learners with a worked-out solution to study, which worked examples do, the need for search is avoided and students can concentrate on building cognitive schemas, so they can more readily solve similar problems in the future.

Many studies have demonstrated the learning advantages of alternating worked examples with problems to solve (e.g., [3, 4, 5]). The learning benefits observed in

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 222–229, 2011.

these studies appear to leverage a two-step learning process in line with cognitive load theory. First, it is helpful for a student, particularly one with low prior knowledge in the domain of interest, to review an example to lessen cognitive load and maximize initial learning. The cognitive schema created by the student while studying the example can then be used to, second, tackle an isomorphic problem to solve, i.e., one with similar structure and/or elements to the example. Instead of grappling with many new and unfamiliar details in solving the new problem, as well as searching through memory, the student can easily recall the similar, just-reviewed example while, at the same time, engage in active cognitive processing to (hopefully) strengthen their understanding of this type of problem and thus achieve deep learning [5].

More recent studies have investigated the benefits of alternating worked examples with intelligently tutored problems [6]. These empirical investigations differ from more traditional worked examples research by the inclusion of *tutored* problems to solve, which provide step-by-step guidance in the form of hints and error feedback and thus offer more scaffolding than ordinary problems. Tutored problems are a middle ground between worked examples and problem solving: they allow students, if they wish, to create worked examples from the problems (by e.g. drilling down to bottom-out hints) but, at the same time, students can actively attempt to solve problems. These recent studies also differ from earlier research in that they have mostly been conducted in the classroom, a decidedly more difficult environment to test learning interventions than the laboratory setting of most prior studies.

All of these recent studies have tested the hypothesis that replacing some tutored problems with worked examples will enhance student learning by reducing instructional time and/or increasing student learning, in terms of retention and transfer. For instance, Schwonke and colleagues [7], in two studies in the domain of geometry, found that a fading worked examples condition, one in which some tutored problems were replaced with examples that were, in turn, gradually replaced first by partially completed examples and, later, by fully-tutored problems, led to as much learning and transfer as a control condition of all tutored problems, yet in significantly less time. McLaren et al's findings [8] in three studies in the domain of chemistry corroborated the efficiency findings of Schwonke et al – more specifically, an alternating example-tutored problem condition yielded the same learning as a tutored-problems-only control, but with significantly better learning efficiency. In summary, it appears that adding worked examples to tutored problem solving helps learning, but the benefits are mostly in improving learning efficiency. Learning outcomes are generally not as significant as those reported in untutored problem solving research (e.g., [3, 4]), which may be explained, at least in part, by the tougher control condition all tutored problems presents.

Given these past findings, both in untutored and tutored problem solving research, together with the generally accepted two-step learning process discussed above and the cognitive load theory underlying it, it is not surprising that researchers have infrequently tested the benefits of presenting students with all worked examples. Why would we expect superior learning benefits with all worked examples when the active problem solving step – the second of the two steps, the step that reinforces the example and (possibly) leads to deep learning – is taken out of the equation? Because we were interested in exploring the never-tested all-worked examples condition (at least never tested within the tutored problem solving line of research) and (re-)testing the generally-accepted two-step learning process of examples followed by problems to solve, we ran a study of the all-examples condition in the context of tutored problem solving. We were skeptical that all-worked examples could produce better learning outcomes or efficiency and formulated the following hypothesis.

> *Alternating worked examples and tutored problem solving will lead to better learning (i.e., better learning retention and/or learning efficiency) than either all tutored problems or all worked examples.*

Given our own prior results [8], which showed better learning efficiency in alternating examples/tutored problems versus all tutored problem solving, as well as the preponderance of evidence supporting the advantages of the two-step learning process [3, 4, 5, 6], the first part of this hypothesis (i.e., *alternating examples/tutored problems > all tutored problems*) was already well supported. Despite some (but limited) evidence that all examples can be more effective for learning and more efficient in mental effort, at least as compared to all untutored problem solving [9, 10], our theory was that all examples might be faster than examples/tutored problems but likely at the expense of careful study and robust learning, thus hurting both learning outcome and efficiency, suggesting the second part of our hypothesis (i.e., *alternating examples/tutored problems > all worked examples*). On the other hand, an "in press" study, one that occurred more-or-less concurrently to ours (yet after our hypothesis was formulated), casts doubt on the notion that an alternating condition is better than all examples, at least with respect to regular, non-tutored problem solving. In this study all worked examples were *as good as* alternating examples/problem solving, with both conditions better than all problems, in terms of both lower cognitive load during learning and higher learning outcomes [11]. Thus, it is clear that the outcome of our study, which will now be described, was not obvious.

2 Method

Participants. One hundred and forty-five (145) high school students (67 female and 78 male) in four chemistry classes in three suburban high schools in three U.S. states (Pennsylvania, Massachusetts and New Jersey) participated. There were 11 additional students who scored 0 (or very nearly 0) on one or both of the two posttests and 3 more students who finished the delayed posttest much later than their classmates; all of these students were eliminated from consideration. The study materials were used as a replacement for normal lectures and class work on the topic of stoichiometry within the four high school classes, and the three participating teachers used the immediate and delayed posttests as class grades for their students.

Materials and Procedure. We conducted a between-subjects study with students randomly assigned to one of the three conditions shown in Table 1. Students in condition 1 (*exs/tps*) were presented with 5 alternating pairs of isomorphic examples and tutored problems, students in condition 2 (*all-tps*) were presented solely with 10 tutored problems corresponding to the same problems received in condition 1, and students in condition 3 (*all-exs*) were presented solely with 10 worked examples corresponding to the same problems received in condition 1. While the intervention materials varied by condition, all students were presented with the same consent form, pre-questionnaire, preparation videos, post-questionnaire, immediate posttest, and delayed posttest. The *n* of each condition is shown at the top of Table 1.

Table 1. Study Design with Three Conditions

Condition 1: ***exs/tps (n=45)***	**Condition 2:** ***all-tps (n=51)***	**Condition 3:** ***all-exs (n=49)***
Consent Form		
Pre-Questionnaire		
Five Preparation Videos		
Intervention Materials: 10 problems in 5 isomorphic pairs: *1st in each pair – example* *2nd in each pair – tutored problem* (3 content videos interspersed)	Intervention Materials: 10 problems in 5 isomorphic pairs, *all tutored problems* (3 content videos interspersed)	Intervention Materials: 10 problems in 5 isomorphic pairs, *all worked examples* (3 content videos interspersed)
Post-Questionnaire		
Immediate Posttest – 8 problems (4 near transfer; 4 conceptual)		
Delayed Posttest (one week later) – 8 problems (4 near transfer; 4 conceptual)		

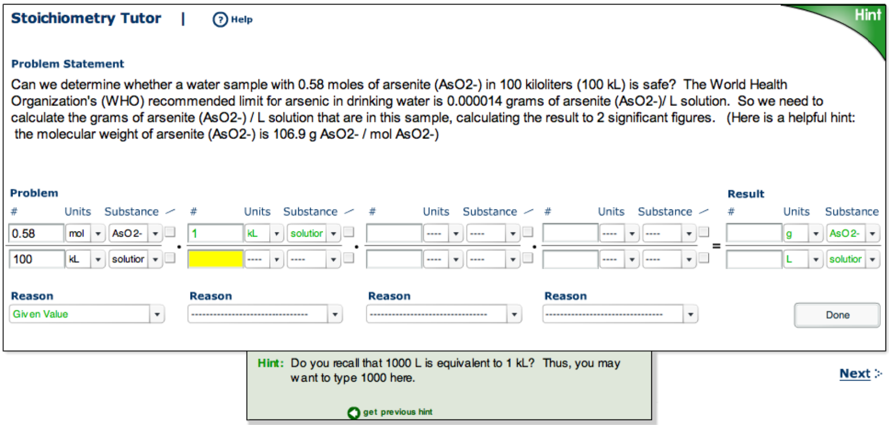
All materials were completed online, within a web browser, in the top-down order shown in Table 1. Students used school-provided computers and headphones, so they could privately listen to the videos. All participants were given user-IDs and passwords that allowed them to logoff and log back on whenever desired, including outside of the classroom.

Because of the usual difficulties in tightly controlling classroom time, the study materials of Table 1 were tackled mostly, but not exclusively, during teacher-monitored classroom time. In a few cases, due to absences or insufficient classroom time, the consent form, questionnaires, videos, and intervention materials were completed outside of classroom time, either at school or home. However, all of the posttests were taken in class. The immediate posttest was administered in the class following completion of the intervention materials and the delayed posttest was administered one week later. Each posttest took 45 to 60 minutes to complete and all of the materials in Table 1 took students between 150 and 240 minutes to complete.

The pre-questionnaire contained basic demographic questions (e.g., gender), as well as self-assessment questions about the student's prior knowledge of chemistry (e.g., "I know what the 2 stands for in H2O", "I know what Na stands for," "Rate your overall knowledge of chemistry, from 1 ('Far below average') to 5 ('Highly above average')"). The pre-knowledge questions were scored between 1 and 15 and all students who scored below the calculated mean of 9.95 were classified as "low prior knowledge learners," while all others were classified as "high prior knowledge learners." Note that we did not administer a pretest, in favor of the self-assessment questions, to avoid testing effects, the phenomenon in which a test can help students learn after they have already pre-studied material [12]. As pupils in a chemistry class, our population of subjects may have been exposed to similar or related materials.

After completing the pre-questionnaire, the students were presented with five videos to prepare them for working with the materials, including a review of how to use the online materials, a review of significant figures, and an overview of stoichiometry problem solving. Next, the students worked on the intervention materials that were specific to their condition, as shown in Table 1. Short (1 to 4 minute) videos were interspersed throughout the materials, presenting various background materials on chemistry concepts relevant to stoichiometry (e.g., molecular weight, dimensional analysis).

Fig. 1 shows the Stoichiometry Tutor, developed using the Cognitive Tutor Authoring Tools [13]. To solve the stoichiometry problems, students must understand basic chemistry concepts, such as molecular weight, and be able to solve algebraic chemistry equations. The student can request hints by selecting the “Hint” button in the upper right-hand corner of the interface. The hints the tutor gives provides progressively more information for solving the problem, with the last hint on each step providing the final answer for that step (a “bottom-out hint”, Fig. 1 is an example of such a hint). If the number typed (or unit or substance or reason selected) is correct, the typed (or selected) information appears in a green font. If it is incorrect, it appears in red. The tutor also provides context-specific error messages when the student makes a mistake during problem solving.



Problem Statement

Can we determine whether a water sample with 0.58 moles of arsenite (AsO2-) in 100 kiloliters (100 kL) is safe? The World Health Organization's (WHO) recommended limit for arsenic in drinking water is 0.000014 grams of arsenite (AsO2-)/ L solution. So we need to calculate the grams of arsenite (AsO2-) / L solution that are in this sample, calculating the result to 2 significant figures. (Here is a helpful hint: the molecular weight of arsenite (AsO2-) is 106.9 g AsO2- / mol AsO2-)

Problem **Result**

#	Units	Substance	#	Units	Substance	#	Units	Substance	#	Units	Substance	#	Units	Substance
0.58	mol	AsO2-	1	kL	solutior		----	----		----	----		g	AsO2-
100	kL	solutior		----	----		----	----		----	----		L	solutior

Reason	Reason	Reason	Reason
Given Value	--------------------------------	--------------------------------	--------------------------------

Hint: Do you recall that 1000 L is equivalent to 1 kL? Thus, you may want to type 1000 here.

Fig. 1. The Stoichiometry Tutor and an example of a bottom-out hint

The worked examples used in Conditions 1 and 3 are implemented as videos of an expert solving and narrating problems with the Stoichiometry Tutor shown in Fig. 1. To prompt reflection and review of the example, the student is asked, after watching a worked example video, to answer several (3 to 5) self-explanation (SE) questions, through multiple-choice, pull-down menus. Examples of SE questions for the worked example version of the problem in Fig. 1 are: “Our goal in this problem was to convert from moles AsO2- per kiloliters solution to grams AsO2-: (a) true, (b) false.”, “We used the unit conversion term 1 kL solution / 1000 L to convert: (a) the numerator of the given value from moles to grams, (b) the denominator of the given value from kiloliters to liters, (c) the numerator of the given value from moles to liters, (d) the denominator of the given value from liters to kiloliters.” If the student incorrectly answers an SE question, the answer turns red. The student cannot proceed to the next problem until they answer all SE questions correctly.

After completing all of the intervention videos and problems, the participants were prompted to respond to a web-based questionnaire that asked about the helpfulness and the usability of the tutor. Finally, the students took an immediate posttest and, one week later, a delayed posttest. The two posttests were isomorphic to one another (A and B), each containing eight problems, 4 of which were of the same type and had the same user interface as the intervention problems (like Fig. 1 but without hints or error feedback; near transfer problems) and 4 of which were more conceptual questions. The order of the A and B tests was counterbalanced (i.e., ½ of the participants received A as the immediate test and B as the delayed test and vice versa).

Test scores were calculated by assigning a score per problem (i.e., dividing the number of correct steps the student took on a single problem by the total number of possibly correct steps for that problem), adding the scores of all eight problems together, and dividing by 8.

3 Results and Discussion

The results comparing the three conditions according to the dependent variables (DVs) immediate and delayed posttest performance are shown in Table 2. ANCOVAs on the DVs immediate and delayed posttest performance, with prior knowledge as a covariate, were run, with post-hoc comparisons between each pair of conditions, including a calculation of effect size (Cohen's d). As can be seen, there were no significant differences in learning between the conditions on either immediate or delayed posttest performance. Separate analyses of the low and high prior knowledge students (not shown in Table 2), as defined by the 9.95 threshold discussed earlier, also did not exhibit significant effects. However, for the low prior knowledge group the ex/tps condition compared to both the all-tps and all-exs conditions reached medium, but not significant, effect sizes for both the immediate and delayed posttests. This is at least a hint toward past results that have shown, for low prior knowledge students, alternating worked examples with problems can be advantageous (e.g., [4]).

Table 2. Comparison according to the DVs imm. posttest and delayed posttest performance

Dependent Variable	C1: exs/tps mean (sd)	C2: all-tps mean (sd)	C3: all-exs mean (sd)	*p*	Cohen's *d* C1 v. C2	Cohen's *d* C1 v. C3	Cohen's *d* C2 v. C3
Imm. Posttest	0.54 (0.18)	0.53 (0.16)	0.54 (0.19)	0.84	0.07	-0.01	-0.08
Del. Posttest	0.62 (0.19)	0.61 (0.16)	0.58 (0.23)	0.27	0.05	0.15	0.11

* - significant difference, $p < 0.05$

The results comparing the three conditions according to the DVs' learning time, learning efficiency on the immediate posttest, and learning efficiency on the delayed posttest, are shown in Table 3[1]. Here, the learning time of the all-exs condition was found to be significantly shorter than either the exs/tps or all-tps conditions, with large effect sizes in both cases. The learning efficiency of the immediate posttest of the all-exs condition was also significantly higher than either of the other two conditions, with medium to high effect size. The learning efficiency of the delayed posttest of the all-exs condition was marginally significantly higher than the alternating condition and significantly higher than the all-tps condition, with medium effect sizes. Notice that the comparison of the alternating condition to the all-tps condition, unlike our prior studies with the same tutors and population [8], was not significant on any of the DVs.

[1] Learning efficiency was calculated per subject as z-score (test score) - z-score (instructional time) with z-score = (value – mean) / sd. This measure is a simplified (but mathematically sound) version of the quantitative model of efficiency first described in [14].

Table 3. Comparison according to the DVs learning time, learning efficiency (imm. posttest), and learning efficiency (del. posttest).

Dependent Variable	C1: exs/tps mean (sd)	C2: all-tps mean (sd)	C3: all-exs mean (sd)	*p*	Cohen's *d* C1 v. C2	Cohen's *d* C1 v. C3	Cohen's *d* C2 v. C3
Learning Time	59.4 (18.0)	63.9 (20.8)	43.0 (12.6)	< 0.0001	-0.23	1.06 *	1.21 *
Learning Eff. (Imm. Posttest)	-0.18 (1.44)	-0.48 (1.54)	0.66 (1.11)	< 0.0001	0.20	-0.66 *	-0.85 *
Learning Eff. (Del. Posttest)	-0.13 (1.44)	-0.41 (1.57)	0.54 (1.17)	0.003	0.18	-0.51 #	-0.68 *

* - significant difference, p < 0.05 # - marginally significant difference, 0.05 < p < 0.1

Our hypothesis was therefore not confirmed: The exs/tps condition did not lead to better learning, either in retention or efficiency, than either the all-exs or all-tps conditions. In fact, the all examples condition showed learning efficiency benefits compared to the other two in this study. Why might this have occurred? The answer may lie in the two-step learning process discussed earlier. The worked examples of this study may provide students with an opportunity to take both of the steps within the context of one example. Students first study the video example and are then encouraged to actively process the material through the follow-up SE questions, which cannot be skipped and must be answered correctly to proceed. While self-explanation is not problem solving, it is implemented in this case in a manner that may trigger a sort of "mental" problem solving. Furthermore, the isomorphic second worked example of each pair may strongly reinforce learning of particular problem types, through a second cycle of the two-step process. In short, while the all-exs condition did not lead to a better learning outcome than the other conditions, it may have promoted a learning model similar to what has worked for alternating examples and tutored problem solving, leading to as much learning yet in a much faster way.

Perhaps even more surprising is the fact that, when comparing exs/tps to all-tps, we did not replicate the significant learning time benefit achieved in 3 of 3 prior studies (effect sizes 1.02, 0.59, 0.54 vs. 0.23 in this study, i.e., viewed in positive direction, instead of the -0.23 in Table 3) or the learning efficiency benefit achieved in 2 of 3 prior studies (effect sizes 0.75, 0.39, 0.56 vs. 0.20 on immediate posttest in this study) [8], even though the same materials and general population were used. We currently have no explanation for this outcome and will continue to explore the data for clues.

4 Conclusion

Our study produced surprising results. While our hypothesis was not confirmed, including replication of our own past work, we made an interesting discovery: at least sometimes, under some conditions, students can benefit the most by learning strictly with worked examples, at least with respect to conserving their time. The worked examples of our study, however, were not static, conventional examples; rather, they were "modeling" examples [15] – live, narrated videos – that were followed by prompted self-explanation questions that had to be answered correctly by the student in order to move on. At least in the context of our domain and our materials, such a type of worked example, which trades off between example study and active problem processing, led to the best results, if not with respect to learning gains than at least with respect to learning efficiency. In other words, our study showed that learning from what might be called "interactive" worked examples may sometimes be a better choice than static worked examples, tutored problems, or problems to solve.

Acknowledgments. The Pittsburgh Science of Learning Center (PSLC), NSF Grant # 0354420, provided support for this research. Dave Yaron and Mike Karabinos assisted in developing the stoichiometry materials used in the study. Tamara van Gog reviewed an early version of this paper and provided valuable comments.

References

1. 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)
2. Sweller, J., Van Merriënboer, J.J.G., Paas, F.: Cognitive Architecture and Instructional design. Educational Psychology Review 10, 251–295 (1998)
3. Mwangi, W., Sweller, J.: Learning to Solve Compare Word Problems: The Effect of Example Format and Generating Self-Explanations. Cog. and Inst. 16, 173–199 (1998)
4. Kalyuga, S., Chandler, P., Tuovinen, J., Sweller, J.: When problem solving is superior to studying worked examples. Journal of Educational Psychology 93, 579–588 (2001)
5. Sweller, J., Cooper, G.A.: The use of Worked Examples as a Substitute for Problem Solving in Learning Algebra. Cog. and Inst. 2, 59–89 (1985)
6. Salden, R.J.C.M., Koedinger, K.R., Renkl, A., Aleven, V., McLaren, B.M.: Accounting for Beneficial Effects of Worked Examples in Tutored Problem Solving. Educational Psychology Review 22(4), 379–392 (2010)
7. Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., Salden, R.J.C.M.: The Worked-Example Effect: Not an Artefact of Lousy Control Conditions. Computers in Human Behavior 25, 258–266 (2009)
8. McLaren, B.M., Lim, S., Koedinger, K.R.: When and How Often Should Worked Examples be Given to Students? New Results and a Summary of the Current State of Research. In: Proc. of the 30th Annual Conf. of the Cog. Sci. Soc., pp. 2176–2181 (2008)
9. Van Gerven, P.W.M., Paas, F., Van Merriënboer, J.J.G., Schmidt, H.G.: Cognitive Load Theory and Aging: Effects of Worked Examples on Training Efficiency. Learning and Instruction 16, 154–164 (2002)
10. Van Gog, T., Paas, F., Van Merriënboer, J.J.G.: Effects of Process-Oriented Worked Examples on Troubleshooting Transfer Performance. Learn. and Inst. 18, 211–222 (2006)
11. Van Gog, T., Kester, L., Paas, F.: Effects of Worked examples, Example-Problem, and Problem-Example Pairs on Novices' Learning. Contemporary Ed. Psychology (in press)
12. Roediger III, H.L., Karpicke, J.D.: The Power of Testing Memory: Basic Research and Implications for Educational Practice. Perspectives on Psych. Science 1(3), 181–255 (2006)
13. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A New Paradigm for Intelligent Tutoring Systems: Example-Tracing Tutors. Int'l J. of AIED 19(2), 105–154 (2009)
14. Paas, F., Van Merriënboer, J.J.G.: The Efficiency of Instructional Conditions: An Approach to Combine Mental Effort and Performance Measures. Human Factors 35(4), 737–743 (1992)
15. Van Gog, T., Rummel, N.: Example-Based Learning: Integrating Cognitive and Social-Cognitive Research Perspectives. Educational Psychology Review 22, 155–174 (2010)

Toward Exploiting EEG Input in a Reading Tutor

Jack Mostow, Kai-min Chang, and Jessica Nelson

Project LISTEN, School of Computer Science, RI-NSH 4103, 5000 Forbes Avenue,
Carnegie Mellon University, Pittsburgh, PA 15213, USA
mostow@cs.cmu.edu,
{kaimin.chang,jessica.nelson}@gmail.com

Abstract. A new type of sensor for students' mental states is a single-channel EEG headset simple enough to use in schools. Using its signal from adults and children reading text and isolated words, both aloud and silently, we train and test classifiers to tell easy from hard sentences, and to distinguish among easy words, hard words, pseudo-words, and unpronounceable strings. We also identify which EEG components appear sensitive to which lexical features. Better-than-chance performance shows promise for tutors to use EEG at school.

Keywords: EEG, reading tutor, power spectrum, frequency band, lexical feature.

1 Introduction

The ultimate automated tutor could peer directly into students' minds to identify their mental states (knowledge, thoughts, feelings, and so forth) and decide accordingly what and how to teach at each moment. The reality, of course, is that today's automated tutors attempt instead to infer students' mental states from a thin trickle of data, typically in the form of mouse clicks and keyboard input. Some ITS researchers (e.g. Anderson, Graesser, Picard, and Woolf, in too many papers to cite here) are exploring other types of data, such as speech, eye movements, posture, heart rate, skin conductance, and mouse pressure. This paper tests a complementary source of input from as close to the brain as non-invasively possible: electroencephalogram (EEG).

The EEG signal is a voltage signal that can be measured on the surface of the scalp, arising from large areas of coordinated neural activity. This neural activity varies as a function of development, mental state, and cognitive activity, and the EEG signal can measurably detect such variation. For example, rhythmic fluctuations in the EEG signal occur within several particular frequency bands, and the relative level of activity within each frequency band has been associated with brain states such as focused attentional processing, engagement, and frustration [1-3], which in turn are important for and predictive of learning [4].

The recent availability of simple, low-cost, portable EEG monitoring devices suddenly makes it feasible to take this technology from the lab into schools. The NeuroSky "MindSet," for example, is an audio headset equipped with a single-channel EEG sensor. It measures the voltage between an electrode that rests on the forehead and electrodes in contact with the ear. Unlike the multi-channel electrode nets worn in labs, the sensor requires no gel or saline for recording, and requires no

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 230–237, 2011.

expertise to wear. Even with the limitations of recording from only a single sensor and working with untrained users, the MindSet distinguished two fairly similar mental states (neutral and attentive) with 86% accuracy [5].

The ability to record longitudinal EEG data in authentic school settings is important for several reasons. First, we can analyze longer-term learning over intervals longer than a lab experiment, in contrast to short-term memory effects. Second, we can study data generated by children's *"in vivo"* behavior at school, rather than their more constrained behavior in unfamiliar lab settings under intense adult supervision. Third, we can get enough data over a long enough time from enough students to combat the notoriously noisy nature of EEG data with the statistical power of "big data," thereby enabling us to analyze the effects of different forms of instruction and practice on student learning and moment-to-moment engagement. Finally, longitudinal recording of EEG data on a school-based tutor offers the opportunity to make student-specific models actually useful, by obtaining enough data over time to train valid models, and applying them on enough occasions to pay off in better student learning.

To assess the feasibility of collecting useful information about cognitive processing and mental state using a portable EEG monitoring device, we conducted a pilot study in which participants wore a NeuroSky Mindset while using Project LISTEN's Reading Tutor [6]. The Reading Tutor displays text, listens to the student read aloud, and logs detailed longitudinal records of its multimodal tutorial dialogue to a database [7]. We linked this data to EEG data by user ID and timestamp.

We wanted to know if MindSet data can distinguish among mental states relevant to learning to read. More specifically:

1. Can EEG detect when reading is difficult? So we presented easy and hard text.
2. Can EEG detect lexical features? So we showed isolated words, varied by type.
3. What EEG components are sensitive, to what features? So we correlated them.

We used a within-subject design to compare the EEG signal during easy vs. difficult reading, at both the passage and single item level, during both oral and silent reading. Sections 2, 3, and 4 address questions 1-3; Section 5 concludes.

2 Can EEG Detect When Reading Is Difficult?

We implemented our experimental protocol in the Reading Tutor's homegrown language for scripting interactive activities. It displayed passage excerpts to read aloud, three easy and three hard, in alternating order. The "easy" passages were from texts classified by the Common Core Standards (www.corestandards.org) at the K-1 level. The "difficult" passages came from practice materials for the Graduate Record Exam (majortests.com/gre/reading_comprehension.php) and the ACE GED test (college.cengage.com:80/devenglish/resources/reading_ace/students). Each passage was followed by a multiple-choice cloze question (formed from the next sentence in the passage) to ensure that readers were reading for meaning. The protocol then repeated these tasks in a silent reading condition, using different text. Across the read-aloud and silent reading conditions, passages ranged from 62 to 83 words long.

10 adult readers participated in our lab, and 11 nine- and ten-year-olds at school. (A few other participants user-tested the protocol or had no EEG data.) We excluded

4 adults and 2 children due to missing or poor-quality data. We analyzed data for the remaining 6 adults and 9 children both separately and pooled across all 15 readers.

2.1 Training Procedure

We trained binary logistic regression classifiers to estimate the probability that a given sentence was easy (or hard), based on EEG data. We trained separate classifiers for each condition (oral and silent reading) and group (adults and children), and also classifiers for data pooled across both conditions and groups.

We trained and tested two types of classifiers for each classification task. We trained *reader-specific* classifiers on a single reader's data from all but one stimulus (passage or word), tested on the held-out stimulus, performed this procedure for each stimulus, and averaged the results to cross-validate accuracy within readers. For stimuli (e.g., passages) with multiple successive observations (e.g., sentences), cross-validating across stimuli avoids improperly exploiting statistical dependencies – such as temporal continuity – between observations of a reader on the same stimulus. We trained *reader-independent* classifiers on the data from all but one reader, tested on the held-out reader, performed this procedure for each reader, and averaged the resulting accuracies to cross-validate across readers.

As features for logistic regression we used the streams of values the MindSet logs:

1. The raw EEG signal, sampled at 512 Hz
2. A filtered version of the raw signal, also sampled at 512 Hz
3. Proprietary "attention" and "meditation" measures reported at 1 Hz
4. A power spectrum of 1Hz bands from 1-256Hz, reported at 8 Hz
5. An indicator of signal quality, reported at 1 Hz

We averaged measures 1-4 over the time interval of each stimulus, excluding the 15% of observations where measure 5 reported poor signal quality.

One problem in training classifiers is class size imbalance. We face this issue because we have more easy sentences than hard ones and more non-words than real words. A common solution is to resample the training data to obtain equal-size sets of training data. However, "random undersampling can potentially remove certain important examples, and random oversampling can lead to overfitting" [8]. To avoid bias due to class size imbalance, we employed three different resampling methods: random oversampling of the smaller class(es), with replacement; random undersampling of the larger class(es); and directed undersampling, in our case by truncating the larger class to the temporally earliest k examples. An adaptive tutor would use such temporal truncation to train user-specific models on each user's initial data. We show results for all three resampling methods.

We computed *classification accuracy* as the percentage of cases classified correctly; chance performance is one over the number of categories. To test whether a classifier was significantly better than chance, we first computed its overall accuracy for each reader, yielding a distribution of N accuracies, where N is the number of readers. Treating this distribution as a random value, we performed a one-tailed T-test of whether its mean exceeds chance performance for the classification task in question. Counting N readers rather than observations is conservative in that it accounts for statistical dependencies among observations from the same reader. Our significance criterion was $p < .05$, without correction for multiple comparisons.

2.2 Results

To find out if our data differed by population, grain size, or modality, we trained classifiers to distinguish between children vs. adults, words vs. sentences, and silent vs. oral reading. Children's and adults' data had no significant differences, but word and sentence reading differed sharply, as did silent and oral reading.

We trained classifiers to distinguish between easy and hard sentences read aloud, silently, or both, by adults, children, or both. Table 1 shows the results; values in **bold** here and later are significantly better than chance. Depending on the resampling method used, accuracy averaged from about 43% to 69% for reader-specific classifiers and 41% to 65% for reader-independent classifiers, respectively, suggesting that imperfect transfer across readers sometimes outweighs the advantage of training on more data; classification of fMRI brain images has a similar qualitative pattern [9]. Reader-specific classification of children's oral reading was especially good, which bodes well for detecting reading struggles in the Reading Tutor.

Table 1. Accuracy in classifying sentences from easy vs. hard text

		Reader-specific			**Reader-independent**		
	condition	**over-sample**	**under-sample**	**truncate**	**over-sample**	**under-sample**	**truncate**
adult	**oral**	0.49	0.56	0.53	**0.65**	0.54	0.41
	silent	0.44	0.43	0.56	**0.63**	0.54	0.54
	both	**0.53**	0.55	0.55	0.54	**0.56**	**0.54**
child	**oral**	0.62	0.62	**0.69**	0.59	**0.59**	**0.63**
	silent	0.47	0.46	0.45	0.50	0.52	0.48
	both	**0.64**	**0.59**	**0.65**	0.47	0.46	0.48
both	**oral**	0.57	**0.60**	**0.62**	0.52	0.52	0.53
	silent	0.49	0.57	0.50	0.53	**0.58**	0.50
	both	0.56	**0.61**	**0.60**	0.47	0.52	0.50

3 Can EEG Detect Lexical Features?

Besides text, our protocol displayed 10 words and 10 pseudo-words one at a time, ordered randomly, to read aloud. Words were all 2-syllable 7-letter words; half were easy and half were hard, to see if our data reflected difficulty in word reading; prior work [10] had found distinct EEG indicators of visual-spatial, orthographic, phonological, and semantic operations in reading. We included non-words to see if we could detect when readers saw unfamiliar words. The "easy" words had a Kucera-Francis (K-F) frequency of 30 or more (mean = 84) and an age of acquisition (AOA) below 315 on a scale from 0-700 (mean = 254.4) [11]. The “hard” words had a K-F frequency below 10 (mean = 3.4) and an AOA above 450 (mean = 555.5). Pseudo-words were 3 letter pronounceable strings, chosen to vary in their number of orthographic neighbors (words that differ in spelling by only one letter), since EEG data (specifically, event related potentials) are sensitive to neighborhood size [12].

The isolated-item section also presented ten illegal 3-character strings to read silently, also with varying orthographic neighborhood sizes, also from the same study; the read-aloud condition omitted illegal strings because they are unpronounceable. We varied the orthographic neighborhood size of the pseudo-words and illegal strings from 0 neighbors to 22 neighbors, to enable (future) analysis of its effects.

We trained and evaluated classifiers just as described in Section 2.1, except that we trained multinomial logistic regression classifiers to estimate the probability that a word was easy, hard, a pseudo-word, or (in the silent condition) an illegal string. We evaluated their *rank accuracy* as the average percentile rank (normalized between 0 and 100) of the correct category if categories are ordered by the value of the regression formula; chance performance is 50%. Rank accuracy is a more sensitive criterion than classification accuracy for evaluating performance on multi-category tasks such as decoding mental states from brain data [9].

We expected it to be harder to distinguish among 3 or 4 kinds of isolated words and non-words than to tell easy from hard sentences, because reading an isolated word is so brief compared to reading a sentence. In addition, we had fewer samples of isolated words than sentences. Nevertheless, as Table 2 shows, rank accuracy averaged from about 45% to 58% for reader-specific classifiers, depending on the resampling method used, and about 39% to 59% for reader-independent classifiers.

Table 2. Rank accuracy (chance = 50%) in classifying words easy, hard, pseudo, or illegal

		Reader-specific			**Reader-independent**		
	condition	**over-sample**	**under-sample**	**truncate**	**over-sample**	**under-sample**	**truncate**
adult	**oral**	0.52	0.51	0.51	0.46	0.43	0.40
	silent	0.50	0.51	0.49	0.51	0.50	**0.59**
	both	0.51	0.53	0.49	**0.54**	**0.56**	**0.58**
child	**oral**	0.48	0.49	0.45	0.42	0.44	0.39
	silent	**0.58**	0.54	0.55	0.48	0.48	0.52
	both	0.49	0.46	0.45	0.42	0.44	0.39
both	**oral**	0.50	0.49	0.48	0.52	0.49	**0.58**
	silent	0.54	**0.56**	0.53	0.48	0.50	**0.54**
	both	0.49	0.49	0.46	0.50	0.51	**0.54**

4 What EEG Components Are Sensitive, to What Features?

To identify sensitive frequency bands, we fit 8 separate linear mixed effects models, one model for each combination of modality (oral vs. silent), item type (sentences vs. isolated words), and population (adults vs. children). A logit transform of the dependent variable predicts whether reading an item of that type in that modality is easy or hard for that population. As fixed factors we used the average value of each standard frequency band – Delta (1 to 3Hz), Theta (4 to 7 Hz), Alpha (8 to 11 Hz), Beta (12 to 29 Hz), Gamma (30 to 100 Hz), and Gamma+ (101 to 256 Hz) – averaged over the duration of the item. We included individual reader identity as a random factor to model the population of readers by allowing a separate intercept value for each reader. Linear mixed effects models are robust to missing data, so we included readers with partial data, for a total of up to 8 adults or 12 children in each model. We used the Wald Z statistic to test significance at the p < .05 level. Despite the small number of readers, we found statistically significant – but different – predictors for adult and child oral sentence reading: the beta band for adults and the gamma band for children.

Besides training classifiers to distinguish easy from hard reading, we performed a follow-up analysis to take advantage of between-sentence variance in lexical content. We took several lexical properties of words from the MRC Psycholinguistic Database [11] and computed their mean values for each sentence. The between-sentence variance of these per-sentence means provided a natural experiment on the EEG effects of lexical properties. We correlated these sentence-level values against the EEG power spectrum for each sentence. The within-sentence variance in lexical properties naturally diluted the correlations, as did EEG signal noise. Adjusting them to compensate for such variance would more accurately estimate presumably stronger true underlying correlations [13].

Table 3. Correlations of EEG power spectra to mean MRC lexical features of sentences: Concreteness CNC, imageability IMG, Mean Colorado Meaningfulness CMEAN, familiarity FAM, age of acquisition AOA, Brown verbal frequency BFRQ, Kucera and Francis written frequency KFRQ, Thorndike-Lorge frequency T-LFRQ, and # letters NLET

	Delta (1-3 Hz)	Theta (4-7 Hz)	Alpha (8-11 Hz)	Beta (12-29 Hz)	Gamma (30-100 Hz)	Gamma+ (101-256 Hz)
CNC						
IMG						
CMEAN		-0.08				-0.10
FAM						-0.08
AOA						
BFRQ	**-0.12**	**-0.13**			**-0.09**	**-0.12**
KFRQ			0.07			**0.10**
T-LFRQ						**0.09**
NLET	**0.11**	**0.13**	**0.10**	**0.10**	**0.14**	**0.16**

Table 3 shows the unadjusted correlations, with a row for each lexical feature and a column for each frequency band. It shows all correlations significant at $p < .05$ without correction for multiple comparisons, and in **bold** if significant using False Discovery Rate [14]. The table shows effects of word length (NLET) and verbal frequency (BFRQ) across multiple frequencies, and (with less confidence) effects of other features in other bands. Differences among features in which bands they correlate with would suggest that different frequency bands carry information about different word-level aspects of reading – information conceivably useful to an automated tutor.

5 Conclusions

We showed that the EEG data from a single electrode portable recording device can discriminate between reading easy and hard sentences reliably better than chance, across populations (adults and children) and modalities (oral and silent reading). We identified frequency bands sensitive to difficulty and to various lexical properties, which suggests that they can detect transient changes in cognitive task demands or specific attributes of lexical access.

Much work remains. We need to detect additional mental states. We need to improve classifier accuracy by collecting more data and by using more sophisticated training methods. Besides manipulating stimuli experimentally, we can label training data based on observable events in longitudinal data, such as improved performance.

Nevertheless, the statistically reliable relationship between reading difficulty and relatively impoverished EEG data illustrates its potential to detect mental states relevant to tutoring, such as comprehension, engagement, and learning. At the level of longitudinal data aggregated across students, such information could help generate and test hypotheses about learning, elucidate the interplay among emotion, cognition, and learning, and identify specific tutor behaviors to prefer. At the level of dynamic data about an individual student, the tutor could adapt to the student, either by responding immediately to a detected mental state, or by adapting more slowly to a cumulative student model updated over time. In summary, this pilot study gives hope that a school-deployable EEG device can capture tutorially relevant information.

Acknowledgments. This work was supported by the Institute of Education Sciences, U.S. Department of Education, through Grants R305A080157 and R305A080628 to Carnegie Mellon University. The opinions expressed are those of the authors and do not necessarily represent the views of the Institute or U.S. Department of Education. We thank the students, educators, and LISTENers who helped generate, collect, and analyze our data, Sarah Laszlo for stimuli, and the reviewers for helpful comments.

References

1. Marosi, E., Bazán, O., Yañez, G., Bernal, J., Fernández, T., Rodríguez, M., Silva, J., Reyes, A.: Narrow-band spectral measurements of EEG during emotional tasks. Int. J. Neurosci. 112(7), 871–891 (2002)
2. Lutsyuk, N., Éismont, E., Pavlenko, V.: Correlation of the characteristics of EEG potentials with the indices of attention in 12-to 13-year-old children. Neurophysiology 38(3), 209–216 (2006)

3. Berka, C., Levendowski, D.J., Lumicao, M.N., Yau, A., Davis, G., Zivkovic, V.T., Olmstead, R.E., Tremoulet, P.D., Craven, P.L.: EEG correlates of task engagement and mental workload in vigilance, learning, and memory tasks. Aviat. Space Environ. Med. 78(5 Suppl), B231–B244 (2007)
4. Baker, R., D'Mello, S., Rodrigo, M.M., Graesser, A.: Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive-affective states during interactions with three different computer-based learning environments. International Journal of Human-Computer Studies 68(4), 223–241 (2010)
5. NeuroSky: NeuroSky's eSenseTM Meters and Detection of Mental State. Neurosky, Inc. (2009)
6. Mostow, J., Beck, J.: When the Rubber Meets the Road: Lessons from the In-School Adventures of an Automated Reading Tutor that Listens. In: Schneider, B., McDonald, S.-K. (eds.) Scale-Up in Education, vol. 2, pp. 183–200. Rowman & Littlefield Publishers, Lanham, MD (2007)
7. Mostow, J., Beck, J.E.: Why, What, and How to Log? Lessons from LISTEN. In: Proceedings of the Second International Conference on Educational Data Mining, Córdoba, Spain, pp. 269–278 (2009)
8. Chawla, N.V., Japkowicz, N., Kolcz, A.: Editorial: special issue on learning from imbalanced data sets. ACM SIGKDD Explorations Newsletter 6(1), 1–6 (2004)
9. Mitchell, T., Hutchinson, R., Niculescu, R.S., Pereira, F., Wang, X., Just, M.A., Newman, S.D.: Learning to decode cognitive states from brain images. Machine Learning 57, 145–175 (2004)
10. Bizas, E., Simos, P.G., Stam, C.J., Arvanitis, S., Terzakis, D., Micheloyannis, S.: EEG Correlates of Cerebral Engagement in Reading Tasks. Brain Topography 12(2), 99–105 (1999)
11. Coltheart, M.: The MRC Psycholinguistic Database. Quarterly Journal of Experimental Psychology 33A, 497–505 (1981)
12. Laszlo, S., Federmeier, K.D.: The N400 as a snapshot of interactive processing: Evidence from regression analyses of orthographic neighbor and lexical associate effects. Psychophysiology (in press)
13. Behseta, S., Berdyyeva, T., Olson, C.R., Kass, R.E.: Bayesian Correction for Attenuation of Correlation in Multi-Trial Spike Count Data. Journal of Neurophysiology 101(4), 2186–2193 (2009)
14. Benjamini, Y., Yekutieli, D.: The control of the false discovery rate in multiple testing under dependency. The Annals of Statistics 29(4), 1165–1188 (2001)

Persistent Effects of Social Instructional Dialog in a Virtual Learning Environment

Amy Ogan[1], Vincent Aleven[1], Christopher Jones[1], and Julia Kim[2]

[1] Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA
[2] University of Southern California, 12015 Waterfront Dr., Playa Vista, CA 90094, USA
{aeo,aleven}@cs.cmu.edu, cjones@andrew.cmu.edu, kim@ict.usc.edu

Abstract. Interactions between learners and pedagogical agents may take on a more or less social tone, dependent on many possible factors. We investigate the effects of manipulating agent dialog on learner-agent interpersonal relations, which is hypothesized to promote learning. We have implemented our model of social instructional dialog (SID) in an agent in a virtual learning environment for instructing intercultural interactions. SID is designed to support students in taking a social orientation towards learning, through the use of conversational strategies that are theorized to produce positive interpersonal effects. This paper reports on the results of an empirical study (N=60) comparing SID to a corresponding task informational dialog (TID) model without these social features. We found that the SID model had significant positive effects on learners' entitativity, shared perspective, and trust with the agent. Moreover, these effects transferred to other non-SID based agents in the environment. We discuss how these findings may impact development of dialog for future agents.

Keywords: virtual agents, instructional dialog, social outcomes, trust, entitativity.

1 Introduction

Embodied conversational agents (ECAs) put a "human" touch on intelligent tutoring systems by using conversation to support learning. Traditionally, the instructional dialog of ECAs has a task orientation, in that it focuses on the instructional task. For example, in a physics tutor, the main focus of agent dialog might be to assist students in solving the next step in a momentum problem, or in better understanding a concept like force or velocity [e.g., 1,2]. Increasingly, the motivational and affective components of student-agent interactions are receiving greater attention [e.g., 3,4]. Especially when considering instruction in interpersonal domains, such as negotiation, the development of an interpersonal relationship with one's pedagogical agent may play a significant role in learning.

There is conflicting evidence regarding the ability of agents to cultivate relationships with humans. On one hand, Reeves and Nass have shown through numerous studies that interaction between humans and computers can appear to mimic facets of human-human relationships [5]. [6] has also shown that agents can engender feelings of rapport through nonverbal cues, although this manipulation did not lead to greater learning outcomes. Other research has shown that social relationships and their

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 238–246, 2011.

subsequent desired outcomes do not always result from interactions with agents. In [7], learning outcomes were increased by the mere belief that a human was generating instructional dialog rather than an agent. The hypothesized explanation for this effect is that learners believed they were taking a socially relevant action only when talking to a "human". To muddy the waters further, [8] have developed a non-embodied social agent that was not rated more highly than a task-based agent on most social outcome measures, yet produced increased learning gains on an engineering task.

Thus, more work must be done to understand how and when agent behaviors create desirable interpersonal effects with the learner. As an emerging topic of research, a number of interpersonal outcomes have been targeted in instructional dialog, and equally many social behaviors have been proposed for getting there [e.g., 6,8,9,10]. In this paper, we report on a study investigating an instructional dialog model aimed at achieving social outcomes. We use the following methodology in order to integrate literature from multiple disciplines and systematically investigate a subset of this domain. First, we chose desired social outcomes based on the instructional domain and prior tutoring research. Next, we looked to human-human communications literature for conversational strategies that are hypothesized to promote these social outcomes (resulting in our *social instructional dialog* (SID) model, fully described in [11]). Third, we compared this dialog model in an empirical study to a comparison model of task instructional dialog, finding that the SID model had significant positive effects on all three outcomes. Finally, we used statistical modeling techniques to understand how these outcomes influenced one another, as well as their effects on other non-SID based agents in the environment. We see our work contributing not only a better understanding of how properties of instructional agent dialog relate to social outcomes, but also a methodology for approaching research questions in this domain.

2 Social Informational Dialog

We first chose three interpersonal outcomes on which to focus our investigation. Our choices were influenced by the domain of our work, which takes place in BiLAT, a simulation designed to teach intercultural negotiation skills [12]. A psychosocial outcome that is tightly coupled to negotiation is *entitativity* – the feeling of working together as a team. In negotiation literature, this feeling leads to more positive affect towards negotiation partners, and also significantly better negotiation outcomes [13]. Beyond influencing negotiation outcomes, agents in all learning environments may benefit from helping learners feel that they are working together as a team to achieve educational outcomes. Numerous studies have reported the positive effects of collaboration on learning [14,15].

Additionally, agents should be able to influence the *perspective* that learners take in the interaction. In both intercultural competence and negotiation, perspective-taking has been shown to be important [13,17]. Beyond intercultural education, the ability to take a shared perspective has great value. For example, in STEM (science, technology, engineering and math) education, one central objective is for students to be able to see themselves taking on the persona of scientists or mathematicians.

Our third, domain-agnostic outcome of interest is *trust.* Trust enables people to make reliability judgments about the accuracy of the information they are receiving.

While trust literature has found considerable evidence that higher levels of trust lead to an increased willingness to listen to useful knowledge and absorb it (see e.g., [17]), this outcome has been understudied in literature on pedagogical agents, although related concepts such as believability and utility have been investigated [e.g., 18].

Figure 1 shows the SID model of the conversational strategies we hypothesize will achieve these outcomes, which is fully described in [11]. Learning objectives are delivered using *narrative,* a form of communication that increases group bonds and allows learners to leverage pre-existing schemas to acquire new information [19]. *Self-disclosure* reveals information about oneself, family, or similarly private items, with the effect of gaining reciprocal trust from the listener [20]. The final strategy our model incorporates is *affirmation*, the acknowledgement that the receiving party's perspective has been heard and understood. We then developed a corresponding task dialog (TID) that does not use these strategies, for use as a comparison condition.

This dialog was developed within the context of BiLAT, in which the learner takes the role of an officer tasked with meeting with Iraqi townspeople to accomplish peacekeeping missions. Within each scenario, the learner must negotiate with one or more ECAs in culturally appropriate ways. These agents simulate members of the Iraqi culture, e.g. a police officer or merchant. The BiLAT interface offers the learner a menu of actions and dialog choices. In general, interaction is turn-based, with each learner selection followed by an utterance from an agent (see Fig. 1). Driving these responses, as well as gestures, gaze, etc., is a model of culture and personality ([12] has a complete agent description). A key learning objective is to consider your meeting partner's interests and perspective, so as to realize a "win-win" negotiation result.

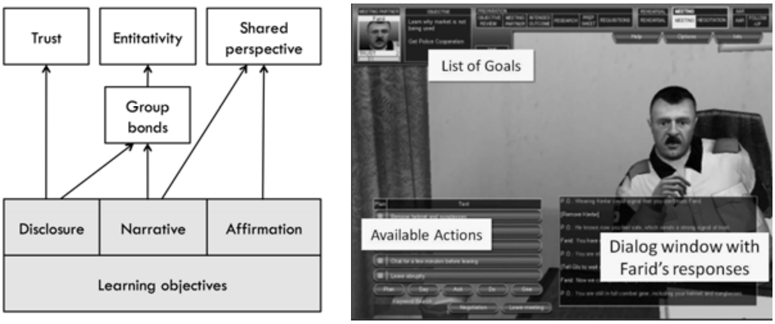


Fig. 1. Left: Model of social informational dialog and effects on interpersonal variables. Right: The BiLAT interface showing Farid, a police officer, and the set of currently available actions.

3 Study

In order to examine the relation between our selected social conversational strategies and interpersonal outcomes, we created an agent named Zahora within the BiLAT environment. Zahora was presented as an introductory character who could help students learn about Iraqi culture. She covered up to 10 learning objectives, and could be configured to use either the SID or the TID model for her dialog. We ran a randomized, controlled experiment to investigate the following individual hypotheses:

H1: A SID-based agent is perceived as more social, while a TID-based agent is perceived as more task-focused.

H2: SID agents influence learners to present an agent-centered perspective, while TID-based agents influence them to present a learner-centered perspective.

H3: Entitativity is higher with a SID-based agent.

H4: Trust is higher with a SID-based agent.

We conducted further analyses to understand how these outcomes were interrelated, and how our manipulation affected interactions in the rest of the environment.

Sixty participants (53% female) were recruited using an online subject pool from two university campuses. Requirements for participation were U.S citizenship and age between 18-25. First, participants were given a briefing describing their role as an officer and introducing the character of Zahora, an Iraqi interpreter. Participants were randomly assigned to either the task instructional dialog (TID) or the social instructional dialog (SID) condition. In both conditions the agent was introduced as an authority on local Iraqi culture. Participants interacted with this agent for as long as they wanted (on average, ten minutes). They then entered into negotiation meetings with two other characters in BiLAT, a police officer named Farid and a businessman named Hassan. These agents used the standard BiLAT dialog (rather than SID or TID) and were identical across condition. Finally, they took a post-interaction survey to rate various qualities of their interactions with the three agents.

Trust was measured through a standardized scale of trust on a seven-point scale [21]. *Entitativity* was assessed with four items on a seven-point Likert scale measuring how much participants felt like they identified with the "team" [22]. Shared *Perspective* was measured using two seven-point Likert items asking whether 1) participants felt they had attempted to express an American perspective with their dialog choices and 2) whether they tried to conform to an Iraqi perspective with their dialog choices. For all items, 1 = strongly disagree, 7 = strongly agree.

4 Results

To verify that learners found the dialog with Zahora to be different between conditions, we asked four manipulation check questions listed in Table 1. Table 1 contains the results of a between-subjects ANOVA for each of these questions with TID vs. SID as the independent variable. Confirming H1, there was a significant difference between conditions on ratings for all four questions.

Table 1. Between-condition ANOVA on manipulation check questions. 7=Strongly Agree.

Item	*ANOVA*	*TID M(SD)*	*SID M(SD)*
Zahora shared personal stories	P<.001; F(1,60)=50.37	3.40 (1.57)	5.90 (1.16)
Zahora was very social	P<.001; F(1,60)=26.34	5.10 (1.30)	6.45 (0.68)
Zahora focused on the task	P<.001; F(1,60)=13.85	4.93 (1.28)	3.61 (1.48)
Zahora did not make much smalltalk	P<.001; F(1,60)=14.92	3.63 (1.43)	2.13 (1.61)

Participants were asked to report on the perspective they believed they were trying to demonstrate in their discussion with Zahora. In the dialog, participants could choose options that reflected an American perspective on the cultural issues, or attempted to conform to the Iraqi perspective asserted by Zahora. Table 2 contains the results of a between-subjects ANOVA on each perspective with TID vs. SID as the independent variable. Confirming H2, TID participants were significantly more likely to claim they were presenting an American perspective in their dialog. SID participants were significantly more likely to state that they were presenting an Iraqi perspective.

Table 2. Between-condition ANOVA on perspective-taking with Zahora. 7=Strongly Agree.

Item	*ANOVA*	*TID M(SD)*	*SID M(SD)*
American perspective	P < .001; F(1,60)=14.78	4.53 (1.50)	3.06 (1.48)
Iraqi perspective	P = .001; F(1,60)=12.30	4.47 (1.70)	5.84 (1.34)

After meeting with the three agents, participants were asked to report how strongly they felt entitativity with each character. Table 3 contains the results of a between-subjects ANOVA comparing TID to SID. Confirming H3, participants in the social condition were significantly more likely to claim they indentified with Zahora as a team. Continuing the investigation to examine how Zahora's dialog model affected perceptions of the other agents in the environment, SID participants were also significantly more likely to state that they identified with Hassan as a team. Although means in the SID condition were higher, the difference in entitativity ratings with Farid did not reach statistical significance.

Table 3. Between-condition ANOVA on entitativity with each character. 7=Strongly Agree.

Scale	*ANOVA*	*TID M(SD)*	*SID M(SD)*
Zahora: Entitativity	P=.001; F(1,59)=11.79	4.53 (1.10)	5.53 (1.14)
Farid: Entitativity	P=.25; F(1,59)=1.35	4.32 (1.36)	4.76 (1.53)
Hassan: Entitativity	P=.04; F(1,59)=4.42	2.66 (1.04)	3.38 (1.54)

Participants were also asked to report how much they trusted each character. Table 4 contains the results of a between-subjects ANOVA comparing TID to SID. Confirming H4, participants in the SID condition had significantly higher trust ratings for Zahora. SID learners also rated their level of trust in Hassan significantly higher than those in the TID condition. However, ratings of trust in Farid did not differ.

Table 4. Between-condition ANOVA on trust with each character. 7= Strongly Agree.

Scale	*ANOVA*	*TID M(SD)*	*SID M(SD)*
Zahora: Trust	P=.016, F(1,59)=6.18	5.03 (1.30)	5.81 (1.11)
Farid: Trust	P=.792, F(1,59)=0.35	4.72 (1.62)	4.62 (1.63)
Hassan: Trust	P=.035, F(1,59)=4.68	2.10 (1.14)	2.94 (1.75)

To investigate the mechanisms by which our experimental manipulation influenced each interpersonal outcome, we used Structural Equation Models (SEM). SEM models each variable as a linear function of its immediate causes and independent

Gaussian noise [23]. We used the SEM algorithm implemented in the software package Tetrad to search for models consistent with our background knowledge. The model generated by Tetrad ($\chi\mathbf{2}(26)=35.39$, $p=.1$)[1] and shown in Fig. 2 revealed several significant relationships. Foremost, the model shows that a participant's strength of entitativity from their initial interactions with Zahora carried forward to subsequent interactions with Farid and Hassan. Thus, if a participant had a high level of entitativity with Zahora, they were likely to have a higher level of entitativity with Farid and Hassan, regardless of condition. Further, the model shows that entitativity increased trust. In other words, the more a participant felt like they are working on a team with an agent, the more trust they had in that agent. The perspective that participants took in the interaction was not seen to affect either their level of entitativity or trust.

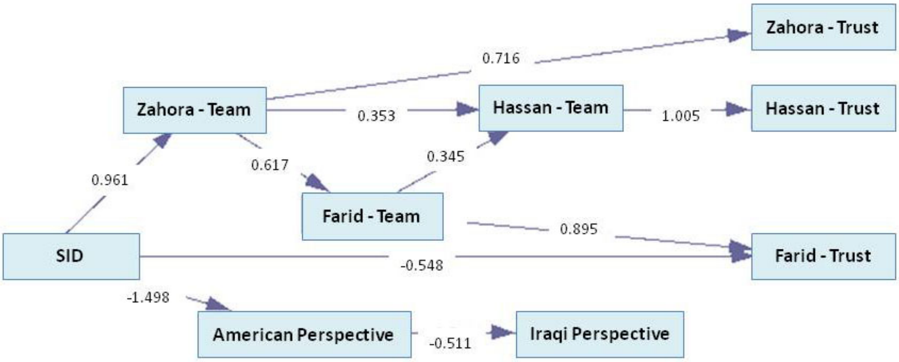


Fig. 2. SEM model showing relationships between condition and interpersonal outcomes. Numbers indicate the model parameters for the strength of the relationship.

5 Discussion

In this work, we have looked at how introducing social conversational strategies into instructional dialog affects interpersonal relations with virtual agents. Our four main hypotheses for this study were confirmed. The social informational dialog (SID) felt more social, while the task-based comparison (TID) felt task-focused. SID also had a significant effect on the desire of learners to demonstrate a shared perspective with an intercultural agent, an ability that is also highly valued in STEM domains where students should see themselves sharing the values and perspective of scientists and mathematicians. Learners in the SID condition also felt more entitativity with Zahora, which could benefit collaborative scenarios such as peer tutoring systems.

SID learners also expressed more trust in the agent. While this finding was in line with our predictions, it contradicts our previous findings on trust [11]. In the current study, SID-condition learners remarked on how useful Zahora had been in preparing them for meeting the rest of the agents, and how relaxed they felt when entering into

[1] In SEM, the p-value reflects the probability that the deviance between the implied covariance matrix (at the maximum likelihood estimate) and the observed covariance is as big or bigger than observed. Thus, p-values greater than .05 indicate that the model fits the data well [23].

difficult negotiations. On the other hand, the TID condition commented that the information contained in preparatory documents had been of greater utility than the agent. As is hypothesized in [18], we believe this may indicate that ratings of trust are tied to utility of information. In [11], our agent was evaluated outside of the full educational context, and the TID agent's more authoritative and less personal tone led students to trust her in the absence of confirmatory evidence. In the current study, the subsequent utility of the SID-based agent gave increased ratings of trust. Further exploration of the concept of trust in learner-agent interactions is warranted.

Following their encounter with Zahora, all participants interacted with two other characters, Farid and Hassan, who were identical across conditions. Learners with a SID-based Zahora felt significantly greater entitativity and trust with subsequent meeting partner Hassan. The SEM model gives further evidence that the strength of learners' perceptions of Zahora was proportional to their attitudes towards the other agents, regardless of condition. Specifically, participants with higher ratings of entitativity with Zahora reported similar higher ratings with both subsequent characters. The SEM model also shows that feeling greater entitativity with each character leads to greater trust in that character.

Although the model shows this to be true over all three characters, effects were most pronounced with Hassan, and effectively hidden in between-subject ANOVAs with Farid. We attribute this outcome to the significant difference in the "likeability" of these two characters. Hassan was intentionally designed to be more difficult, less cooperative, and less team-driven, an impression confirmed by post-study survey questions and interviews. Farid, on the other hand, was received quite positively, aligning with his intended design. We believe that for Hassan, who had the lowest reported "likability" of the three characters, the magnitude of the manipulation effect was more visible. This has interesting implications for agent design. In a dialog-based learning environment for STEM skills, all pedagogical agents might be designed with SID. However, to transfer to real-world encounters in the domain of interpersonal skills, students must learn to deal with "difficult" people. Our results suggest that priming learners with a SID manipulation can raise entitativity ratings of subsequent difficult character from negative to neutral, and may induce sufficient motivation in the learner to push through the engagement.

While each of our results constitutes a contribution to the literature, our current dialog model combines several strategies to maximize interpersonal effects, and more work will be done to tease apart which strategy does in fact cause each particular effect. We believe that using the research methodology followed in this paper to investigate further desirable social outcomes will contribute to building a generalized model explaining the underlying social phenomena. A complimentary, critical avenue of research will be to investigate the mediated relationship between conversational strategies, learner-agent interpersonal relations, and learning results. Understanding how these strategies, and in turn, the interpersonal relationships they develop, relate to learning will enable the creation of agent dialog with strong benefits for education.

Acknowledgments. The research was supported by the Institute of Education Sciences, U.S. Dept. of Ed. Grant R305B040063, and the U.S. Army Research, Development, and Engineering Command. 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.

References

1. Litman, D., Silliman, S.: ITSPOKE: An Intelligent Tutoring Spoken Dialogue System. In: Companion Proc. of the HLT/NAACL, Boston, MA (2004)
2. Jordan, P., Makatchev, M., Pappuswamy, U., VanLehn, K., Albacete, P.: A natural language tutorial dialogue system for physics. In: Sutcliffe, G., Goebel, R. (eds.) Proceedings of the 19th Intl FLAIRS Conf. AAAI Press, Menlo Park (2006)
3. D'Mello, S.K., Picard, R., Graesser, A.C.: Toward an affect-sensitive AutoTutor. IEEE Intelligent Systems 22, 53–61 (2007)
4. Kim, Y., Baylor, A.L.: A Social-Cognitive Framework for Pedagogical Agents as Learning Companions. J. Edu. Tech. Research and Dev. 54(6), 569–596 (2006)
5. Reeves, B., Nass, C.: The Media Equation: How People Treat Computers, Television, and New Media like Real People and Place. Cambridge University, Cambridge (1996)
6. Wang, N., Gratch, J.: Can a Virtual Human Build Rapport and Promote Learning? In: Proc 14th International Conference on AIED, Brighton (2009)
7. Okita, S.Y., Bailenson, J., Schwartz, D.L.: Mere Belief of Social Action Improves Complex Learning. In: Barab, S., Hay, K., Hickey, D. (eds.) ICLS 2008. Lawrence Erlbaum Associates, New Jersey (2008)
8. Kumar, R., Ai, H., Beuth, J., Rosé, C.: Socially Capable Conversational Tutors Can Be Effective in Collaborative Learning Situations. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 156–164. Springer, Heidelberg (2010)
9. Bickmore, T., Cassell, J.: Relational agents: a model and implementation of building user trust. In: Proc. of CHI, pp. 396–403. ACM, New York (2001)
10. Wang, N., Johnson, W.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. Ogan, A., Aleven, V., Kim, J., Jones, C.: Developing interpersonal relationships with virtual agents through social instructional dialog. In: Proc. of IVA. Springer, Heidelberg (2010)
12. Hill, R.W., Belanich, J., Lane, H.C., Core, M.G., Dixon, M., Forbell, E., Kim, J., Hart, J.: Pedagogically Structured Game-based Training: Development of the ELECT BiLAT Simulation. In: Proc. 25th Army Science Conf. (2006)
13. Gelfand, M., Brett, J. (eds.): The Handbook of Negotiation and Culture, Stanford (1983)
14. Johnson, D.W., Johnson, R.T.: Cooperative learning and achievement. In: Sharan, S. (ed.) Cooperative learning: Theory and Research, pp. 23–37. Praeger, NY (1990)
15. Lou, Y., Abrami, P.C., d'Apollonia, S.: Small group and individual learning with technology: A meta-analysis. Review of Educational Research 71(3), 449–521 (2001)
16. Neale, M.A., Bazerman, M.H.: The role of perspective-taking ability in negotiating under different forms of arbitration. Industrial and Labor Relations Review 36, 378–388 (1983)
17. Mayer, R.C., Davis, J.H., Schoorman, F.D.: An integration model of organizational trust. Academy of Management Review 20, 709–734 (1995)
18. Lester, J.C., Converse, S.A., Kahler, S.E., Barlow, S.T., Stone, B.A., Bhogal, R.S.: The Persona Effect: Affective Impact of Animated Pedagogical Agents. In: Proc. CHI 1997, pp. 359–366 (1997)
19. Bochner, A.P., Ellis, C., Tillmann-Healy, L.M.: Relationships as stories: Accounts, storied lives, evocative narratives. In: Dindia, K., Duck, S. (eds.) Communication and Personal Relationships, pp. 12–29. John Wiley & Sons, Ltd., Chichester (2000)

20. Cozby, P.C.: Self-disclosure: A literature review. Psychological Bulletin (1973)
21. Wheeless, L., Grotz, J.: The Measurement of Trust and Its Relationship to Self- Disclosure. Human Communication Research 3, 250–257 (1977)
22. Leach, C.W., van Zomeren, M., Zebel, S., Vliek, M.L.W., Pennekamp, S.F., Doosje, B.: Group-level self-definition and self-investment: A hierarchical (multicomponent) model of in-group identification. J. of Personality and Social Psych. 95, 144–165 (2008)
23. Spirtes, P., Glymour, G., Scheines, R.: Causation, Prediction, and Search, 2nd edn. MIT Press, Cambridge (2000)

A Teachable-Agent Arithmetic Game's Effects on Mathematics Understanding, Attitude and Self-efficacy

Lena Pareto[1], Tobias Arvemo[2], Ylva Dahl[3], Magnus Haake[4], and Agneta Gulz[5]

[1] University West, Media & Design, Sweden
lena.pareto@hv.se
[2] University West, Mathematical Statistics, Sweden
Tobias.Arvemo@hv.se
[3] Uddevalla Schools, School Development, Sweden
ylva.dahl@uddevalla.se
[4] Lund University Design Sciences, Sweden
Magnus.haake@design.lth.se
[5] Lund University Cognitive Science, Sweden
agneta.gulz@lucs.lu.se

Abstract. A teachable-agent arithmetic game is presented and evaluated in terms of student performance, attitude and self-efficacy. An experimental pre-post study design was used, enrolling 153 3rd and 5th grade students in Sweden. The playing group showed significantly larger gains in math performance and self-efficacy beliefs, but not in general attitude towards math, compared to control groups. The contributions in relation to previous work include a novel educational game being evaluated, and an emphasis on self-efficacy in the study as a strong predictor of math achievements.

Keywords: teachable agents, mathematics achievement, attitude, self-efficacy.

1 Introduction

Educational games for mathematics have documented effects on learning and motivation [1], [2], [3], [4] and [5]. Games are considered to be effective tools since they are action-based; motivational; accommodate multiple learning styles and skills; reinforce mastery skills; and provide interactive and decision making context [5]. The instructional effectiveness of a game depends both on its particular characteristics and how it is used in classroom instruction [4], [6]. The relation between game characteristics and competence promotion is not well understood [3]. One such characterization is proposed in [7] where the authors claim that technology, such as games, need to be pedagogically sound, mathematically true and cognitively defined in order to deepen understanding. Furthermore, technology should bring reasoning into the environment and allow exploration, conjecture and testing to deepen mathematical understanding [7]. Teachable Agents (TA), i.e., agents that can learn [8], have previously been used to scaffold reflection, conceptual understanding as well as motivation [2], [9]. Below, we will discuss how our Teachable Agent Arithmetic Game (hereafter TAAG) relates to the described characterization.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 247–255, 2011.

In addition to performance, affective issues need to be included in studies of cognition and instruction to have an impact on mathematics education [10]. Attitude, belief, and emotion are the major descriptors of the affective domain, and many mathematics educators consider attitude as their major concern [10]. Attitude toward mathematics refers to: students' affective responses to – their liking or disliking of – mathematics; their tendency to engage in or avoid mathematical activities; their belief in their mathematics ability (i.e., self-efficacy) and their believing that mathematics is useful or useless [11]. Lately, self-efficacy has attracted special attention since self-efficacy beliefs have been shown to be strong predictors of actual accomplishments [12], [13], and [14]. Therefore, we have chosen to study the issues of general attitude towards mathematics and self-efficacy separately.

The primary research questions addressed in this study are: Will TAAG have effects on 1) conceptual understanding of arithmetic, 2) attitudes towards mathematics and 3) self-efficacy beliefs regarding arithmetic performance? As a secondary explanatory question we will explore if achievements are effected by students' self-reported like/dislike of mathematics and/or by different levels of authenticity in the learning situation. To address the latter issue we compare the situation where the game play occur in full class lead by regular teachers (referred to as fully-authentic setup) to the situation where game play takes place in smaller groups lead by researchers as instructors (semi-authentic setup). Instructors own enthusiasm in dealing with material may affect students' absorption of values and importance [15].

2 Related Mathematic Game Studies

Criticisms have been raised towards game studies that either are non-authentic in their setup [5] or lack control groups to the treatment [4]. Therefore, we restrict related work to longer experimental studies in authentic settings where both achievement and affective measures for math are investigated and compared to controls.

Ke and Grabowski [16] used strategy games for arithmetic problem-solving and enrolled 125 voluntary 5th grade students in a 4-week study. Three conditions were investigated: cooperative, competitive and no game play. For math performance, both game playing groups performed significantly better than the control. For attitude, the cooperative game play group was significantly more positive than the other two conditions. This study relates to ours by student age, general topic, and two measures.

In a larger (N=358) similar study, Ke also investigated metacognitive awareness by a self-report questionnaire [6]. Metacognition and self-efficacy are related: while self-efficacy is a predictor of both declarative and procedural knowledge, metacognition is only related to procedural knowledge [14]. In this study, significant effect in attitude was found, but neither math performance nor metacognitive awareness showed effects. This study is similar to our, the main difference is the Teachable Agent game.

In a 18-week study with 193 9th and 10th grade students, the effects of a mission-based game for algebra was investigated [5]. The results indicated significant gain in math achievement. No significant improvement was found in the motivation of the groups.

A geometry puzzle game was used in a 10 session study enrolling 29 6th grade students with 2 play conditions: with and without level progression [17]. The authors found significant performance gains for both treatment groups compared to pre tests results and controls. The affective measure reported gains, but concerned attitude towards the game instead of towards math.

Finally, our game has been evaluated in two previous studies: a short-term pilot study [18] and a study involving a small number of special education students [19].

3 The Teachable Agent Arithmetic Game

The educational content of the game is basic arithmetic with a particular focus on conceptual understanding of base-10 and the arithmetic operations. The approach in our environment is to provide 1) an animated, graphical model simulating arithmetic behavior; 2) a set of two-player games based on the model; and 3) intelligent, teachable agents which can be taught to play the games. In the animated simulation model, square-boxes are explicitly packed/unpacked, to illustrate carrying and borrowing. The computation 48+43 = 91 is illustrated in Fig. 1:

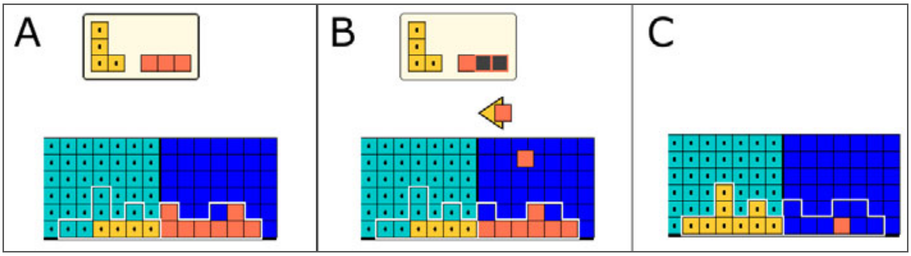
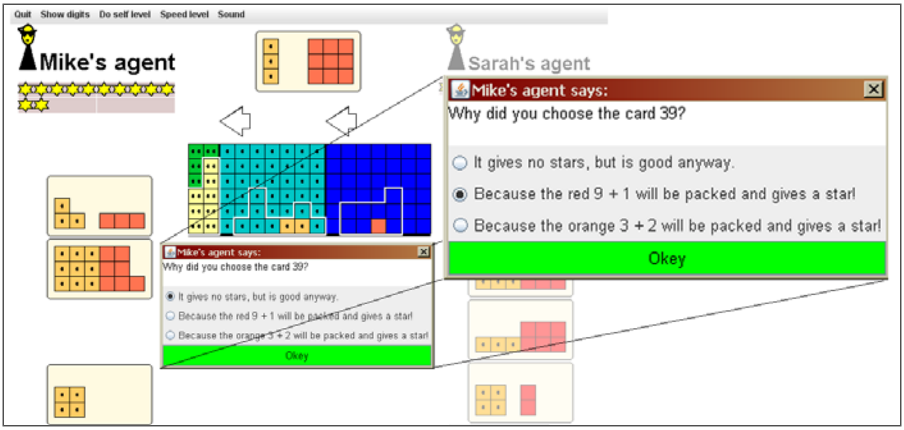


Fig. 1. The carrying operation as an animated packing of squares into a square box

In A, the number 48 is represented graphically on the game board at the bottom (4 orange one-dot squares in the left compartment, and 8 red squares in the right), and the number 43 on the card above. Addition is to put objects on the board, subtraction to remove. Picture B captures the animated packing of 10 red singleton squares into a sized 10 square box. In C, the computation 48+43 = 91 is completed.

In the games, each player acts an arithmetic operation and receives a set of cards with graphical numbers. The players take turn choosing a card until all cards are played. A game, i.e., a sequence of turns, thus constitutes a computation x_1+y_1+x_2+…. The player's task is for each turn to choose the best possible card according to various game goals, such as maximizing number of carryings or number of zeroes in the intermediate results. The task involves predicting the cards effects (i.e., the results of one-step computations), reasoning about the available choices for the current turn, and longer term strategies to maximize scoring. The target knowledge is structural properties of the base-10 system, and how numbers behave under computations.

Besides playing themselves, students can teach an agent to play the game in a master-apprentice manner. Students take on the role as teacher, which most find very engaging. Agents are taught in two ways; by showing how to play or by having the agent try making a choice according to its knowledge, which the student either accepts or corrects to a (possibly) better choice. Either way, the agent asks the student reflective questions *on why* a particular choice was made or was better. For example, in figure 2, Mike is teaching his agent in show mode, and has just chosen the card 39, instead of the other choices 33, 97 and 40. The agent, being an inquisitive learner, asks Mike why the choice 39 was good before the computation takes place and the effect is known. The system also provides plausible explanatory responses (with one correct explanation) in a multiple-choice format, for the student to choose from. The agent learns from observing and analyzing the students playing behavior, and from the question responses.



Mike's agent says:

Why did you choose the card 39?

○ It gives no stars, but is good anyway.

● Because the red 9 + 1 will be packed and gives a star!

○ Because the orange 3 + 2 will be packed and gives a star!

Okey

Mike's agent says:

Why did you choose the card 39?

● It gives no stars, but is good anyway.

○ Because the red 9 + 1 will be packed and gives a star!

○ Because the orange 3 + 2 will be packed and gives a star!

Okey

Fig. 2. Mike is teaching his agent by showing how to play and answer explanatory questions

In this way, the TA provides guidance to connect to symbolic math and stimulates reflection of game playing behavior, often required in games to help learners achieve deep understanding [20], [3]. Reflective thinking is an important condition for learning mathematical concepts [17]. The game design adheres to pedagogical fidelity by promoting reflective thinking, allowing exploration and manipulation of virtual objects; to mathematical fidelity by ensuring mathematical soundness in its behavior and by providing substantial training in reasoning and logic; and finally to cognitive fidelity by a its concrete, visual and explicit representations of numbers and operations. The essence of mathematics involves observing and investigating patterns and relationships between objects [7].

4 Method

This study used a pre-post experimental design. The objective of the present study was to evaluate the TA Arithmetic game with respect to the hypotheses that playing the game would: 1) support students' conceptual understanding of basic arithmetic as revealed in the difference in post-test vs. pre-test scores on a mathematical comprehension paper-and-pencil test; 2) scaffold more positive attitudes in students towards the topic of mathematics as revealed in the difference in post-test vs. pre-test scores on an attitudes questionnaire; and 3) scaffold better self-efficacy beliefs as revealed in the difference in post-test vs. pre-test scores on a task-specific self-efficacy questionnaire. We also explored in a secondary analysis if students' self-reported like/dislike for mathematic and/or authenticity levels affected achievements.

The study enrolled five 3rd grade classes and four 5th grade classes at two different locations, southern and west of Sweden, in total 153 students. Due to school policy and practical reasons for a longer in-class, within curriculum study, we had to split conditions at the level of class. The enrolled classes were chosen, at each location, to be as similar as possible with respect to socio-economic background, overall performance, digital competence, amount of math instruction, and pedagogical approach. One class from each location and level were assigned playing condition, the others a no-intervention control condition. Each year, all 3rd and 5th grade students in Sweden take mandatory standard tests in mathematics at a pre-determined period of some weeks, which occurred during the study period. Much attention is paid under math classes to prepare and take the tests during this time, since results are basis for nationwide quality comparisons. This ensured that the math activities during the study were as equivalent as possible between conditions, apart from the intervention.

One of the locations had a semi-authentic setting (game play in groups of 8, monitored by researchers) the other a fully authentic setting (entire class, their regular teachers). There were 68 students in the play condition, 51,5% girls and 48,5% boys.

The playing classes used the game for 9 weeks, aiming for one 40-minute session per week, instead of other activities during regular mathematics classes. Control conditions proceeded with regular instruction. Prior to the study both conditions completed a paper and pencil test in three parts: 1) arithmetic base-ten math problems, 2) questions regarding general attitudes to mathematics, and 3) questions addressing math self-efficacy. After the intervention both conditions completed a post-test with the same three parts and questions as the pre-test. Repeated measures were collected, but only as in-game progression parameters, which are yet to be analyzed.

The math test consisted of 36 items in 7 problem types, adapted to the two age-groups. Several problems were inspired by previous standard national tests, for example using alternative representations of the base-10 system such as "nature-money" where leaves, cones and stones represent ones, tens and hundreds, respectively. Tasks included translating between nature-money and integers, using nature-money for computations and judge the value of nature objects (place value). Other tasks involved deciding which of two sums is the greatest (e.g., *857+275* or *475+639*) or deciding if a sum's result will be an even ten (e.g., 361+439) by reasoning rather than performing formal computations. Students were told they did not need to calculate; some of the examples were deliberately too difficult for them to compute, and there were no room for calculations. To answer such problems by reasoning require a rather deep understanding of the base 10 system and addition.

The questionnaire for assessing attitudes towards math was inspired by Bandura's design guidelines for self-efficacy scales. For general attitude, one explicit ("*Do you think math is boring or fun*?" on a continuous scale very boring to very fun) and 4 implicit questions of math liking such as *"What do you think about learning new topics in math?* were included. The explicit and implicit questions will be correlation tested for validation. For self-efficacy 4 items such as "*How confident are you in deciding which of the sums 47+32 or 35+41 is the largest?*, which all were task-specific in the sense that concrete examples are given for confidence judgments.

Since a Likert scale format with a mark for neutral was used, affective responses were measured with a ruler on the scale (-3,3) with 0 as the neutral point, resulting in the range of (-12,12) for attitude and self-efficacy questions. The explicit like/dislike question (hereafter called math enjoyment variable) was asked only once, since such opinion is considered as a stable property [21] and used for categorization into low, medium and high positive attitude towards mathematics.

5 Results

First, we present the descriptive statistics for the pre test results in Table 1 below:

Table 1. Descriptive statistics of Pre tests for the play and the control group

		Pre Math Achievement (max 36)			**Pre Attitude** (min -12, max 12)			**Pre Self-Efficacy** (min -12, max 12)		
	N	Mean	Median	SD	Mean	Median	SD	Mean	Median	SD
Control	**85**	26,11	27,00	5,51	2,67	2,65	4,90	6,79	8,07	4,19
Play	**68**	24,86	26,00	7,02	3,97	4,00	4,25	5,50	6,16	4,39
Total	**153**	25,55	27,00	6,24	3,25	3,50	4,66	6,22	6,98	4,31

A pre-treatment test (Mann-Whitney) was conducted for between group comparisons. The results shows that there are no significant pre-treatment differences between the two condition groups, neither for math achievement, general attitude nor self-efficacy (all *p> ,05*). Neither is there any group difference of the math enjoyment indication, which correlates strongly to the attitude measure. Hence, we can compare the score gain directly, as shown in Table 2 (descriptive statistics).

Table 2. Descriptive statistics of Gain (difference pre and post test)

		Gain Math Achievement			**Gain Attitude**			**Gain Self-Efficacy**		
	N	Mean	Median	SD	Mean	Median	SD	Mean	Median	SD
Control	**85**	,93	1,00	4,38	1,50	1,83	3,82	-,23	,00	3,62
Play	**68**	3,19	3,00	5,28	,93	,50	3,09	1,44	1,20	3,52

To examine our three main hypothesis, we have conducted a Mann-Whitney between group comparison test for score gains, showing that there are significant effects for the math achievement (*p=,01*) and the self-efficacy (*p=,009*) in favor of the treatment group, but no significant effect for the attitude gain score *(p=,172).* Effect sizes are *0,47* for achievement and self-efficacy, and *-0,16* for attitude. For comparison, an ANCOVA controlling for pre-test results yields similar p-values for math achievement and self-efficacy (*p=,01* and *p=,03*) and considerably higher for attitude (*p=,67*). Hence, hypothesis 1 and 3 are supported, but not 2.

The explanatory secondary analysis of the within treatment group difference with respect to the categorization of the self-reported math enjoyment variable showed math achievement mean gain for the respective sub groups *(low=9,00; medium=3,14; high=2,35; total=3,19)*, for the general attitude measure *(low=1,23; medium=0,44; high=1,16; total=0,93)*, and for the self-efficacy measure *(low=3,29; medium=0,09; high=1,94; total=1,44)*. There are no significant differences between the subgroups in any of the three measures. For the categorization into semi- and fully authentic groups, there was a slightly larger gain for the semi-authentic group for math achievement (*3,41* compared to *2,97,n.s.*), significantly larger gain in attitude for the semi-authentic group (*1,23* compared to *0,14, p=0,026*), and finally a slightly larger gain of self-efficacy for the fully authentic group (*1,77* compared to *1,14, n.s.*).

6 Discussion and Conclusion

The results support the hypothesis that playing the game improves students' conceptual arithmetic understanding and increases students' self-efficacy beliefs, but not the hypothesis that it scaffolds more positive attitudes towards mathematics in general. For math performance, similar findings using other games are reported in [17], [16], and [5], and the present result strengthens previous indicative results [18].

Measuring attitude change in relation to game usage seems to be a more diverse issue, both in terms of used measures and what the results indicate. Three related studies showed a positive change in attitude, whereas this study and [5] did not. The overall attitude measure used in [6], where a positive change was detected, ought to be compared to both our measures attitude and self-efficacy, since we separated the issues whereas Ke included both in one measure. The attitude measured in [17] is not comparable, since it concerned attitude towards the game and not the subject mathematics. Considering the students' positive engagement and attitude towards the game (as evident from observations, teachers' and students' testimonies), the lack of attitude gain in our study may be explained by students not including the game play in general mathematics did (as indicated in post intervention interviews) and that 9 weeks is too short to change an attitude formed during several years [21].

We consider the positive change in self-efficacy beliefs to be the main contribution of this paper since it has not been studied as a separate issue in related works. Also, self-efficacy beliefs are strong predictors of future math accomplishments [12]. We suggest that the particular game design contributes to an explanation by: 1) the absence of failures; choices can be better or worse but never wrong, and 2) the TA allowing students to act the role of an expert, boosting self-esteem and confidence. A future study comparing the game with and without the TA, should shed light on 2).

The exploratory analysis of within treatment group difference with respect to math enjoyment and authenticity level were not significant but give indications for further research. (For example, a future larger study allowing multi-level analysis, e.g., hierarchical linear model analysis, should provide further insights.) The low attitude group show the largest gains on all three measures, which may suggest that an unconventional approach to math could particularly attract these students (also observed in [19]). Being at-risk and a difficult group to attract, such indications deserve further research. Finally, we can only speculate on why the significant difference in attitude gains between the semi- and fully-authentic groups (in favor of semi-authentic) appeared, but perhaps the extra attention, the authority of being researchers or their conviction of the game's value played a role.

References

1. Rieber, L.P.: Seriously considering play: Designing interactive learning environments based on the blending of microworlds, simulations, and games. Educational Technology Research & Development 44(2), 43–58 (1996)
2. Schwartz, D.L., Martin, T.: Inventing to Prepare for Learning: the Hidden Efficiency of Original Student Production in Statistics Instruction. Cognition and Instruction 22, 129–184 (2004)
3. Moreno, R., Mayer, R.E.: Role of guidance, reflection and interactivity in an agent-based multimedia game. Journal of Educational Psychology 97(1), 117–128 (2005)
4. Vogel, J.F., Vogel, D.S., Cannon-Bowers, J., Bowers, C.A., Muse, K., Wright, M.: Computer gaming and interactive simulations for learning: A meta-analysis. Journal of Educational Computing Research 34(3), 229–243 (2006)
5. Kebritchi, M., Hirumi, A., Bai, H.: The effects of modern mathematics computer games on mathematics achievement and class motivation. Comp. & Education 55, 427–443 (2010)
6. Ke, F.: Computer games application within alternative classroom goal structure: cognitive, metacognitive and affective evaluation. Educational Teachnology Research Development 56, 539–556 (2008)
7. Bos, B.: Virtual math objects with pedagogical, mathematical, and cognitive fidelity. Computers in Human Behavior 25, 521–528 (2009)
8. Biswas, G., Katzlberger, T., Brandford, J., Schwartz, D.L., TAG-V.: Extending intelligent learning environments with teachable agents to enhance learning. In: Moore, J.D., Redfield, C.L., Johnson, W.L. (eds.) Artificial Intelligence in Education, pp. 389–397 (2001)
9. Schwartz, D.L., Chase, C., Wagster, J., Okita, S., Roscoe, R., Chin, D., Biswas, G.: Interactive Metacognition: Monitoring and Regulating a Teachable Agent. In: Hacker, D.J., Dunlosky, J., Graesser, A.C. (eds.) Handbook of Metacognition in Education (2009)
10. McLeod, D.B.: Research on Affect and Mathematics Learning in the JRME: 1970 to the Present. Journal for Research in Mathematics Education 25(6), 637–647 (1994)
11. Ma, X., Kishor, N.: Attitude Toward Self, Social Factors, and Achievement in Mathematics: A Meta-Analytic Review. Educational Psychology Review 9(2), 2 (1997)
12. Pajares, F., Graham, L.: Self-Efficacy, Motivation Constructs, and Mathematics Performance of Entering Middle School Students. Contemporary Educational Psychology 24, 124–139 (1999)
13. Bandura, A.: Self-efficacy: The Foundation of Agency. L. Erlbaum, Mahwah (2000)
14. Moores, T.T., Chang, J.C.-J., Smith, D.K.: Clarifying the role of self-efficacy and metacognition as predictors of performance: construct development and test. SIGMIS Database 37(2-3), 125–132 (2006)
15. Frenzel, A.C., Pekrun, R., Goetz, T.: Perceived learning environment and students' emotional experiences: A multilevel analysis of mathematics classrooms. Learning and Instruction 17, 478–493 (2007)
16. Ke, F., Grabowski, B.: Gameplaying for maths learning: cooperative or not? British Journal of Educational Technology 38(3), 249–259 (2007)
17. Sedig, K.: Toward operationalization of 'flow' in mathematics Learnware. Computers in Human Behavior 23, 2064–2092 (2007)
18. Pareto, L., Schwartz, D.L., Svensson, L.: Learning by guiding a teachable agent to play an educational game. In: Proc. of the 14th Int. Conference on Artificial Intelligence in Education, pp. 662–664. IOS press, Amsterdam (2009)

19. Nilsson, A., Pareto, L.: The Complexity of Integrating Technology Enhanced Learning in Special Math Education – A Case Study. In: Wolpers, M., Kirschner, P.A., Scheffel, M., Lindstaedt, S., Dimitrova, V. (eds.) EC-TEL 2010. LNCS, vol. 6383, pp. 638–643. Springer, Heidelberg (2010)
20. Mayer, R.E.: Should there Be a Three-Strikes Rule Against Pure Discovery Learning? The Case for Guided Methods of Instruction. Educational Psychologist 59, 14–19 (2004)
21. Middleton, J.A., Spanias, P.A.: Motivation for Achievement in Mathematics: Findings, Generalizations, and Criticisms of the Research. Journal for Research in Mathematics Education 30(1), 65–88 (1999)

Using Contextual Factors Analysis to Explain Transfer of Least Common Multiple Skills

Philip I. Pavlik Jr., Michael Yudelson, and Kenneth R. Koedinger

Human Computer Interaction Institute,
Carnegie Mellon University, USA
{ppavlik,yudelson,koedinger}@cs.cmu.edu

Abstract. Transfer of learning to new or different contexts has always been a chief concern of education because unlike training for a specific job, education must establish skills without knowing exactly how those skills might be called upon. Research on transfer can be difficult, because it is often superficially unclear why transfer occurs or, more frequently, does not, in a particular paradigm. While initial results with Learning Factors Transfer (LiFT) analysis (a search procedure using Performance Factors Analysis, PFA) show that more predictive models can be built by paying attention to these transfer factors [1, 2], like proceeding models such as AFM (Additive Factors Model) [3], these models rely on a Q-matrix analysis that treats skills as discrete units at transfer. Because of this discrete treatment, the models are more parsimonious, but may lose resolution on aspects of component transfer. To improve understanding of this transfer, we develop new logistic regression model variants that predict learning differences as a function of the context of learning. One advantage of these models is that they allow us to disentangle learning of transferable knowledge from the actual transfer performance episodes.

Keywords: computational models of learning, educational data mining, transfer appropriate processing.

1 Introduction

Transfer of learning is often thought to be the sine qua non goal of education, and the field has now acquired more than one hundred years of experimental research in this area [4, 5]. One finding is that transfer of learning to new contexts is often a difficult feat to achieve [6]. Because of this difficulty, which is manifest in the often frequent tendency of changes in instruction to fail to result in changes to assessed performance, research that shows transfer and helps us understand its mechanisms is highly relevant to the goals of education.

Of course, different educational research approaches propose different mechanisms for transfer. One tradition uses task analysis to assigns skills to tasks within a particular domain [1, 7-10]. One main assumption of these models is that knowledge component (KC) transfer is a unitary process. This is explicit in the structure of the models. For instance, whether we use Bayesian knowledge tracing (BKT), AFM [3] or PFA [1], they all assume that when a knowledge component transfers, it transfer as

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 256–263, 2011.

a knowledge component unit that is the same for different contexts, where those contexts differ in features hypothesized to be irrelevant to the retrieval or application of the KC. So, if our model says that a certain problem step requires a KC for both least common multiple and equivalent fractions skills, this step is very literally taken to be the sum or product (depending upon the conjunction rule, i.e. additive factors (AFM) version or conjunctive factors version, CFM) of the two *unitary* skills as they function apart from particular learning or performance contexts. In other words, every KC is functionally independent of the presumed irrelevant aspects of the context of both learning and application and exists as a latent variable. This all-or-none Q-matrix skill assignment formalism is very convenient because every future performance can be considered as a simple formula of the prior experience with the KCs.

In contrast to this parsimonious assumption that each task can be cleanly categorized as involving some list of discrete KCs, we might speculate that if two problems/steps share a KC, these two problems may cause different learning of the shared KC or may react differently to the learning of the shared KC. Indeed, this idea is not completely new since different strategies causing different degrees of transferability has been shown before [11], so our main contribution here is a formal model to describe such situations. This model implies that the learning that occurs with practice cannot be simply described according to a list of the latent variables involved in the problem, but rather additional insight is gained when we assume that each item class or skill class causes more or less accumulation of latent skill strength and is affected more or less by the accumulation from other classes of items.

In this paper we explore the hypothesis that there is more to be learned about transfer than can be inferred from current discrete skill models (but see Pardos et al. [12, 13], which have related goals). To do this, we first analyze our data with the PFA model to provide a baseline, and then provide two variant models that provide deeper reflection on the transfer effects in our data by characterizing the independent effects of skills learned in different contexts. Because our intent is to understand the data rather than optimize fit to data, we do not compare our work with other models such as BKT. No doubt, if these other models were made sensitive to context (e.g., Bayesian knowledge tracing with different learning transition probabilities for different categories of future contexts), they might achieve similar explanations of the contextual transfer.

On a practical level, this modeling allows us to meticulously relate problems to see which are the most effective for creating transfer. For example, consider the task we will analyze, least common multiple (LCM). Some of the cases, like finding the LCM of 3 and 5, can be solved by providing the product, 15 (we call these problems "product" problems, or type A). Other problems cannot be solved by the product, since the LCM of 4 and 6 is not 24 but 12 (we call these problems "LCM" problems, or type B). To a student that has not clearly learned the meaning of LCM, feedback on attempts of the two types of problems may seem ambiguous, because the product knowledge component matches the answer for many problems. Our PFA model versions will primarily examine this distinction between question types.

A detailed analysis of situations like these can be very difficult using conventional experimental methods because conventional experiments tend to produce only a few data points during learning, and often are designed to contrast overall conditions rather than trial by trial transfer in the building of a cognitive skill. On the other hand, tutoring systems in the classroom typically do not deliver the sort of randomized practice needed for many of the most interesting analyses. Because of these limitations, we created an experimental design that merged the advantages of controlled design with the advantages of tutor based classroom delivery. With the help of Carnegie Learning Inc., we did this by placing our content within the Bridge to Algebra (BTA) tutoring system at Pinecrest Academy Charter Middle School.

2 Design

The data was collected in several sixth and seventh grade classes at Pinecrest Academy Charter Middle School as integrated "Warm-up" units that would come up in the course of student's normal use of the Bridge to Algebra Cognitive Tutor. These warm-ups were given at 10 separate points with different content across the 62 sections of BTA. The LCM warm-up had 16 single step problems chosen randomly from a set of 24 problems, of which 14 were type A and 10 were type B. Correct responses were indicated and incorrect responses were followed by a review of the correct answer, which was presented on the screen for 18 seconds. While there were 4 conditions of practice that included some additional information for some of the problems, we did not find any reliable differences due to these conditions (which included providing some direct instruction or an analogy), so we will just be reporting on the effects during practice as a function of the text of the problem the student needed to solve.

The report below covers the results for the 1st problem set of LCM problems. 197 subjects completed 16 trials with this 1st problem set, and another 58 subjects were also included from a condition that had only 8 single step problems for this 1st problem set. Problems texts are shown in Table 1.

Table 1. Examples of the fixed factors conditions. Problem numerals (as shown below) were matched across the story or no-story question types (of which there were 12 each).

Problem Example	Story Item	Product Item
What is the least common multiple 4 and 5?	no	yes
What is the least common multiple 8 and 12?	no	no
Sally visits her grandfather every 4 days and Molly visits him every 5 days. If they are visiting him together today, in how many days will they visit together again?	yes	yes
Sally visits her grandfather every 8 days and Molly visits him every 12 days. If they are visiting him together today, in how many days will they visit together again?	yes	no

3 Performance Factors Analysis

The PFA model has been presented previously, so the following description is abbreviated. PFA owes its origin to the AFM model and the Q-matrix method, since it uses a Q-matrix to assign prior item types (or KCs in the typical Q-matrix) data to predict the future performance for these same types of items. Because it uses a logic of item categories rather than KC categories, PFA has a single intercept parameter for each item type that describes in-coming knowledge of that type of item. Given this configuration, the PFA model uses logistic regression to estimate item-type category performance as a function of all item types (or KCs) that transfer according to the Q-matrix.

PFA's standard form is shown in Equation 1, where *m* is a logit value representing the accumulated learning for student *i* (ability captured by α parameter) practicing with an item type *k*. The prior learning for this item type is captured by the β parameters for each KC, and the benefit of correctness (γ) or failure (ρ) for prior practice is a function of the number of prior observations for student *i* with KC *j*, (*s* tracks the prior successes for the KC for the student and *f* tracks the prior failures for the KC for the student).

$$m(i, j \in \text{KCs}, k \in \text{Types}, s, f) = \beta_k + \sum_{j \in \text{KCs}} (\gamma_j s_{i,j} + \rho_j f_{i,j}) \tag{1}$$

Together, the inclusion of both correctness and incorrectness in the model make it sensitive to not only the quantity of each event, but also the relative ratio of correct to incorrect. Data for success and failures counts always refers to events prior to the predicted event, consistent with creating a model that is predictive for the effect of learning [1].

4 Model Versions and Transfer Implications

The following section shows how the two similar types of LCM problems result in significantly different benefits to transfer. All of these models include fixed effect assumptions about both the LCM vs. product fixed effect and a fixed effect for the story problems as compared to the explicit LCM problems.

Note that while we are not fitting any fixed (optimized) student parameters for any of the following models, we are fitting students as a random effect that is estimated according to standard random effects modeling (`lme4` package in R, `lmer` function). A random effect is any effect that is sampled from a population over which statistical inferences are to generalize. Because, we want our models to generalize across students in general, not just those sampled, our subjects qualify as random effects. Furthermore, we often have practical problems (with large numbers of students) with producing continuous parameter distributions when we fit subjects as a fixed effect. In contrast, we find that when we include no subject parameter at all (a third alternative and the approach used in a prior PFA paper [1]), parameter values found by the model tend to settle on values that track student ability rather than learning, as evidenced by negative ρ values. Indeed, for both the fixed and random effect subject models, we find tend to find larger, usually positive, ρ values. Considering the wide distribution of student abilities, we believe that adding subject variance in the model "purifies" the γ and ρ parameters (yields more interpretable estimates of these parameters) by minimizing their role in tracking student prior differences (the subject variance does this) and better focusing their role on tracking learning.

4.1 Analysis of PFA Result (Full Q-matrix)

Table 2 shows the standard PFA model with a full Q-matrix assumption ([1,1],[1,1])[1] already makes a highly interpretable if simplistic prediction that type B (LCM<product) items cause far more learning than type A (LCM=product) items. That is, the success learning rate for LCM items (γ_B=.30) is greater than for product items (γ_A=.07). We offer this initial example to contrast with the following examples. Because we have modeled subject prior learning as random effects with mean 0, learning rates for failures (ρ), as well as success (γ), are positive (unlike prior model with no subject terms).

Table 2. Standard PFA parameters found. Product (subscript A) refers to LCM problems solved by the product of the two numbers. LCM (subscript B) refers to LCM problems where the LCM is less than the product. Random effect of subject prior learning had an SD of 0.71.

Influences	Parameter	Estimate	Z-score Est.	p-value	Factor
A & B	intercept	-0.33	-3.40	0.000671	overall prior learning
A	β	1.25	14.80	<2.0E-16	prior learning prod
story items	β	-1.03	-12.94	<2.0E-16	prior learning story
A & B	γ_A	0.07	2.07	0.0381	successes product
A & B	γ_B	0.30	9.22	<2.0E-16	successes LCM
A & B	ρ_A	0.05	1.01	0.313	failures product
A & B	ρ_B	0.07	2.47	0.0136	failures LCM

4.2 Analysis of Contextual AFM (CAFM) Result

Rather than assume some particular Q-matrix, we now introduce the CAFM model that instantiates each cell in the Q-matrix with a parameter. While a normal Q-matrix assumes that each Q-matrix *column* is controlled by 1(CAFM) or 2 (CPFA) parameters, our new contextual models assign 1 or 2 parameters per *cell*. In this case, we fit CAFM since we wished more clear comparison with the prior model with the ρ failure learning rate. Since we have dropped 2 parameters and added 2 parameters, the complexity of PFA and CAFM is equivalent.

While the model fit is slightly worse with the CAFM model (see Table 5), confirming the importance of capturing success and failures separately, we find that the model parameters in Table 3 enrich our understanding of student transfer, while not disagreeing with the PFA result that type B practice is more effective. The basic pattern that is being revealed is one that might be described as transfer appropriate processing (TAP) [14]. In the case of practice with type A or type B, we see that learning effects are much weaker when transfer is measured with the other type.

However, the story is not so simple because the parameters do indicate a significant transfer effect from type B practice to type A performance. This result conflicts with the simple TAP result and shows that the story is more complex. Indeed, this B to A transfer is dramatic since type B problems transfer about 6 times better to type A than the reverse (.09/.015). If, following Pennington, Nicolich, Rahm

[1] Q matrix specification is in matrix notation, row by row. Columns of the matrix correspond to the items (or KCs) that influence items in rows. Thus, a Q matrix defined as ([,X,Y], [X,1,1], [Y,0,1]) says X is influenced by both X and Y, while Y only influences itself.

[15], we advocate the idea that failure to transfer entails rote procedural learning, and that success at transfer involves declarative conceptual learning, we might suppose that type B problems provide more conceptual practice. This is plausible because, based on the very different demands of type B (a relatively complex back checking procedure that checks factors of the product, or a relatively complex sequence of steps starting with prime factors), we could easily expect different declarative learning effects for items that require these strategies, since these strategies tend to build an organized understanding of the factor structure of the specific numbers in addition to a general understanding of factoring. In contrast, it seems that type A problems may mostly review multiplication procedures, since any deeper factor search on these problems is not immediately productive; students fail to learn a transferable understanding of common factors from these problems.

Table 3. Contextual AFM parameters found. Random effect of subject prior learning had an SD of 0.99.

Influences	Parameter	Estimate	Z-score Est.	p-value	Factor
A & B	intercept	-0.47	-4.04	5.28E-05	overall prior learning
A	β	1.44	9.61	<2.0E-16	prior learning prod
story items	β	-1.06	-13.06	<2.0E-16	prior learning story
A	γ_A	0.18	3.83	0.000131	drill count product
A	γ_B	0.09	2.59	0.00962	drill count LCM
B	γ_A	0.015	0.43	0.670	drill count product
B	γ_B	0.24	8.62	<2.0E-16	drill count LCM

Table 4. Contextual PFA parameters found. Random effect of subject prior learning had an SD of 0.74.

Influences	Parameter	Estimate	Z-score Est.	p-value	Factor
A & B	intercept	-0.33	-3.06	0.00223	overall prior learning
A	β	1.29	8.81	<2.0E-16	prior learning prod
story items	β	-1.07	-13.15	<2.0E-16	prior learning story
A	γ_A	0.16	3.02	0.00254	successes product
A	γ_B	0.08	1.85	0.0639	successes LCM
A	ρ_A	0.19	2.52	0.0117	failures product
A	ρ_B	0.07	1.76	0.0779	failures LCM
B	γ_A	0.02	0.40	0.692	successes product
B	γ_B	0.43	10.53	<2.0E-16	successes LCM
B	ρ_A	-0.02	-0.30	0.763	failures product
B	ρ_B	0.07	2.17	0.0301	failures LCM

4.3 Analysis of Contextual PFA (CPFA) Result

The preceding model suggests that context of learning matters, and this final model provides further detail by joining CAFM and PFA to create CPFA. An interesting pattern in the learning of students picked up by this model is the strong effect of failures on type A to type A performance, and the similarly relatively weak effect of failures of type B on type B performance. This is interpreted by the likely much greater ease with which students can infer the method from the type A solution feedback. Indeed, one can imagine that students quite often note that the method is multiplication for type A problems after they fail. In contrast, type B solution feedback might provide an anchor for the ambitious student to build a useful conceptual structure for future problems, but knowledge of the answer allows no easy inferences about method such as for type A problems. This suggests that students may be particularly benefitted by instructional scaffolding following failure for these harder type B LCM problems.

5 Conclusions

The Q-matrix method of assigning each latent variable (or KC) a single parameter (or 2 for PFA) and then overlaying a binary matrix that assigns KC's to items is more parsimonious than the CFA method, but lacked the ability to provide as rich an understanding of how transfer was occurring. Notably, the best fitting Q-matrix models (R2s see below) predict no transfer, while the CAFM and CPFA models both find significant (1-way test in the CPFA case with $p<0.10$) transfer for parameters describing the effect of LCM items on product item performance. See Tables 3 and 4.

While the primary purpose of this paper was to show how artificial intelligence methods can be used to understand complex hypotheses about educational transfer, Table 5 shows some aggregate fit statistics. These statistics support the idea that contextual logistic regression models improve fit only slightly, highlighting the importance of their interpretive value. Table 5 shows AFM and PFA models in 4 Q-matrix variants for comparison, only PFA-F was described in detail above.

Table 5. Comparison of the fit of the 4 model versions. R1 Q-matrix – ([,A,B],[A,1,1],[B,0,1]). R2 Q-matrix – ([,A,B],[A,1,0],[B,0,1]). R3 Q-matrix – ([,A,B],[A,1,0],[B,1,1]).

Model	Obs.	*LL*	*MAD*	*r*	*A'*
AFM-F	3616	-2075	0.411	0.346	0.705
AFM-R1	3616	-2047	0.404	0.376	0.722
AFM-R2	3616	-2042	0.402	0.377	0.721
AFM-R3	3616	-2052	0.404	0.367	0.719
CAFM	3616	-2038	0.401	0.380	0.724
PFA-F	3616	-2038	0.392	0.430	0.754
PFA-R1	3616	-2037	0.394	0.422	0.751
PFA-R2	3616	-2020	0.388	0.434	0.755
PFA-R3	3616	-2032	0.391	0.420	0.750
CPFA	3616	-2017	0.387	0.440	0.759

Of course, the CFA method requires a number of parameters that scales with the number of KCs squared, while Q-matrix methods only increase parameters as a linear function of KCs. This clearly indicates that more data is needed to successfully fit a CFA model relative to a Q-matrix model. This does not diminish the fact that when enough data is available, and it is properly balanced and randomized, contextual models such as described in this paper will likely provide better quantitative fits and enhance opportunities to discover unexpected transfer effects.

Acknowledgements. 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. Pavlik Jr., P.I., Cen, H., Koedinger, K.R.: Performance Factors Analysis – a New Alternative to Knowledge Tracing. In: Dimitrova, V., Mizoguchi, R. (eds.) Proceedings of the 14th International Conference on Artificial Intelligence in Education, Brighton, England (2009)
2. Pavlik Jr., P.I., Cen, H., Koedinger, K.R.: Learning Factors Transfer Analysis: Using Learning Curve Analysis to Automatically Generate Domain Models. In: Barnes, T., Desmarais, M., Romero, C., Ventura, S. (eds.) Proceedings of the the 2nd International Conference on Educational Data Mining, Cordoba, Spain, pp. 121–130 (2009)
3. Cen, H., Koedinger, K.R., 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, pp. 164–175. Springer, Heidelberg (2006)
4. Thorndike, E.L., Woodworth, R.S.: The Influence of Improvement in One Mental Function Upon the Efficiency of Other Functions (I). Psychological Review 8, 247–261 (1901)
5. Judd, C.H.: Special Training and General Intelligence. Education Review 36, 28–42 (1908)
6. Wertheimer, M.: Productive Thinking (1945)
7. Koedinger, K., McLaren, B.: Developing a Pedagogical Domain Theory of Early Algebra Problem Solving. CMU-HCII Tech. Report 02-100 (2002)
8. Kieras, D.E., Meyer, D.E.: The Role of Cognitive Task Analysis in the Application of Predictive Models of Human Performance. In: Schraagen, J.M., Chipman, S.F., Shalin, V.L. (eds.) Cognitive Task Analysis. Lawrence Erlbaum Associates Publishers, Mahwah (2000)
9. Barnes, T., Stamper, J., Madhyastha, T.: Comparative Analysis of Concept Derivation Using the Q-Matrix Method and Facets (2006)
10. Barnes, T.: The Q-Matrix Method: Mining Student Response Data for Knowledge. In: American Association for Artificial Intelligence 2005 Educational Data Mining Workshop (2005)
11. Simon, H.A.: The Functional Equivalence of Problem Solving Skills. Cognitive Psychology 7, 268–288 (1975)
12. Pardos, Z., Heffernan, N.: Detecting the Learning Value of Items in a Randomized Problem Set. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education. IOS Press, Brighton (2009)
13. Pardos, Z., Heffernan, N.: Determining the Significance of Item Order in Randomized Problem Sets. In: Proceedings of the 2nd International Conference on Educational Data Mining, Cordoba, Spain, pp. 111–120 (2009)
14. Morris, C.D., Bransford, J.D., Franks, J.J.: Levels of Processing Versus Transfer Appropriate Processing. Journal of Verbal Learning and Verbal Behavior 16, 519–533 (1977)
15. Pennington, N., Nicolich, R., Rahm, J.: Transfer of Training between Cognitive Subskills: Is Knowledge Use Specific? Cognitive Psychology 28, 175–224 (1995)

Scenario-Based Training: Director's Cut

Marieke Peeters[1,2,⋆], Karel van den Bosch[2],
John-Jules Ch. Meyer[1,2], and Mark A. Neerincx[2,3]

[1] Information and Computing Sciences, Utrecht University
[2] Training Innovations, TNO - Human Factors
mpeeters@cs.uu.nl
[3] Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Abstract. Research regarding autonomous learning shows that freeplay does not result in optimal learning. Combining scenario-based training with intelligent agent technology offers the possibility to create autonomous training enriched with automated adaptive support delivered by a director agent. We conducted an experiment to investigate whether *directing* training scenarios improves the quality of training. Six instructors rated video fragments of directed and non-directed scenarios in terms of learning value. Results show that the instructors consider directed scenarios to be considerably more effective for learning than non-directed scenarios. Implications for the design of a director agent are discussed.

Keywords: intelligent agents, autonomous training, director agent.

1 Introduction

Scenario-based training (SBT) is a powerful way to let trainees prepare, execute, and evaluate real (authentic) tasks within a simulated environment [4,12]. SBT meets the principles recognized in dominant instructional theories as described by Merrill (2002) [9]. Important benefits of training within a simulated environment are the reduction of risks and the possibilities for control over training, e.g. authoring the scenario, delivering feedback, and instructing the actors. However, this control can only be exerted when the scenario is not playing. Control *while* the scenario unfolds is problematic, if not impossible. Yet such control is also highly desirable. Research has shown that trainees need a suitable amount of support *during* training tasks [5]. For instance, if the trainee is performing well, it would be interesting to tell an actor to make a mestake. Whereas if the trainee panicks, it would be better to tell an actor to take over. During normal SBT, such adjustments are hard to accomplish. However, by using intelligent agent technology it becomes possible to wield *online control* over training in advanced practice environments, such as serious games [3]. This can be achieved by developing a *director agent* (DA) that controls the scenario as it unfolds; it monitors the course of events in the training environment, analyzes and assesses

[⋆] Corresponding author.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 264–271, 2011.

suitable ways to proceed, and instructs non-player characters (NPCs) to execute, or refrain from, particular actions. The DA uses its means of control to create meaningful and suitable experiences for the trainee.

1.1 Automated Control: The Director Agent

The issue of this paper is how to automate control over a training scenario as a means to guide and support the trainee. The idea to obtain control over a scenario while it unravels, is not new [2,16]. Within the domain of interactive narrative, there are interesting publications on this subject. In several papers the concept of a director agent (DA) is mentioned, and whereas some researchers merely describe an architecture [7], others actually implemented a framework or built a prototype [8,10,14]. Within the mentioned paradigm, the reason for an intervention is a narrative discrepancy, e.g. the player (Little Red Riding Hood) decides to visit her grandmother by bicycle, therefore, the DA intervenes to hold on to the original storyline by giving the player a flat tire.

The current paper will focus on a different reason for intervening, i.e. to create learning opportunities for the trainee that lie within the zone of proximal development [11,15]. Such opportunities are challenging, yet not confusing [13], but most certainly not boring [1]. This paper focuses on such *pedagogical* interventions. During SBT, instructors use their experience and intuition to intervene; they recognize that a trainee seems lost, overwhelmed or bored and decide to adjust the scenario to attune it to the trainee's needs. To be able to automate these interventions, we need to turn such implicit notions into explicit ones, for instance by defining behavioral cues and events that accompany confusion or boredom, e.g. a lack of activity, the amount of mestakes, posture, etc.

Pedagogical interventions can be divided into two types: supportive and challenging interventions. Supportive interventions are needed when the trainee is performing actions leading him to a situation that is too complex. The trainee receives support to get through some overly complex situation, while leading him to a less demanding situation. Challenging interventions are executed when the trainee is performing all the right actions, but is not being sufficiently challenged. The trainee is motivated to take the training to a higher level. Interventions can consist of adjustments of the complexity level, the availability of information, the salience of certain cues or the amount of learning goals addressed simultaneously.

But even if we define such explicit cues for interventions, the question still remains how effective such interventions are. Clearly, the goal of the interventions is to improve the quality of learning. We argue that a learning situation offers optimal learning opportunities if a trainee is able to cope with the demands, while still being challenged to learn new things [11]. The proximity of a training situation to this optimum can be expressed as the *learning value*. If a training situation has a low learning value, this means that the situation does not meet the trainee's needs: the trainee is either incapable of coping with the demands or he is not being challenged enough to motivate him. In both cases an intervention would be necessary to attune the scenario to the trainee's needs. The question is: Do interventions actually lead to an improvement of the learning value?

Research Question and Hypotheses. The research question in the current study is: "*Will the director's interventions during scenario-based training, triggered by explicit behavior cues, improve the learning value of the scenario?*" We hypothesize that interventions of a director will improve the learning value of the training scenario as rated by professional instructors.

Chosen Task Domain: 'Bedrijfshulpverlening'. We chose 'bedrijfshulpverlening' (BHV, a Dutch word) to be the task domain for our research. BHV entails the application of first aid and fire fighting by a team of company employees. We created four scenarios: (A) a diabetic woman suffering from hypoglycemia, (B) a lady trapped within a room because of a fire in a trash can near the door, (C) an unconscious cleaning lady, who fainted because of an intoxicating gas and (D) a woman with a broken hip (as a result of fleeing in panic from a fire) lying near a fire hazard. Scenarios were developed to train one individual BHV member. All scenarios included two NPCs playing the roles of victim and bystander.

A detailed script enabled the director to intervene in the scenario in predefined ways. Supportive as well as challenging interventions were triggered by possible behaviors of the trainee. For example, the director's script for scenario (A) contained the following line: "If the trainee is asking irrelevant questions for over three minutes (behavioral cue), the victim is instructed to tell the trainee that her vision is blurred (intervention)." Other cues for supportive interventions included: the trainee repeatedly calls emergency services or fails to perform certain checks. The director used these cues to initiate supportive interventions, e.g. instructing the NPCs to reassure the trainee or to offer their assistance.

A challenging intervention was triggered if the situation proved to be too simple for the trainee to handle, indicated by perfect or near perfect performance. The following rule comes from the director's script of scenario (B): "If the trainee communicates his plans and checks the door of the burning office according to protocol (cues), the bystander is instructed to remain passive (intervention)." Examples of behavioral cues for challenging interventions included: making eye contact with bystander and victim, remaining calm, and giving clear instructions. The resulting challenging interventions included instructing the NPCs to: ask for trainee's attention simultaneously, withhold important information, or create extra complications (e.g. running into a fire hazard).

Prototype: Wizard of Oz Set-Up. Because of the laborious task of implementing a prototype of the envisioned training system, we developed a Wizard of Oz prototype; all agents (NPCs and director) were human and the simulated environment was not virtual. All scenarios took place within a real office room at trainees' company building. This gave us the opportunity to investigate approaches for directing training and their effects on the quality of training.

Two NPCs (human actors), playing the roles of bystander and victim, both received two versions of the behavior they were to display during the scenarios: a supportive and a challenging version. Supportive behaviors were helpful to the trainee. Challenging behaviors were impeding or distracting. Another script was developed for the director. This script contained explicit trainee behavior cues, triggering the director to intervene in specific ways while the scenario unfolded. The execution of an intervention was implemented by instructing the actors to change their behavior from supportive to challenging or vice versa.

2 Methods

2.1 Raters

Six experienced instructors in BHV were asked to rate the video-fragments.

2.2 Materials

Footage. We selected twenty video fragments as a test set. Each fragment contained a part of a recording of a trainee playing one of the aforementioned BHV scenarios. All selected video fragments contained trainee behavior cueing an intervention. In half of the fragments shown to the instructors, the director executed all the interventions (*directed condition*) by telling the actors through in-ear portophones to switch between their behavior variations. In the other half of the fragments, the director was absent; even though the fragments all contained behavioral cues, the associated interventions were not executed (*non-directed condition*). Additionally, both conditions (directed and non-directed) contained five fragments that started off with the actors playing their supportive parts (*supportive startup*), and five fragments that started off with the actors playing their challenging parts (*challenging startup*).

Questionnaire. The raters were asked to evaluate the learning value of the situation for a particular trainee by answering the following question.

” *The learning situation at this point in time offers the trainee . . . opportunities to achieve the learning goals at his own level.* ”

-3	-2	-1	0	1	2	3
absolutely no	no	not really any	maybe some	some	enough	exactly the right

2.3 Procedure

The raters received an elaborate instruction to this experiment, containing an explanation of scenario-based training, exemplified by a video fragment. The four scenarios were explained and the learning goals of each scenario were explicitly pointed out. Finally the raters received instructions regarding the procedure of the experiment and explanations to the questionnaire. The raters were oblivious of the research question of the experiment.

Raters were then presented with two sets of video fragments (a practice set and a test set) following a standard procedure. The video fragment was introduced by a short description of the original scenario and the intended learning goal. The part of the fragment preceding the point of intervention was shown. At the

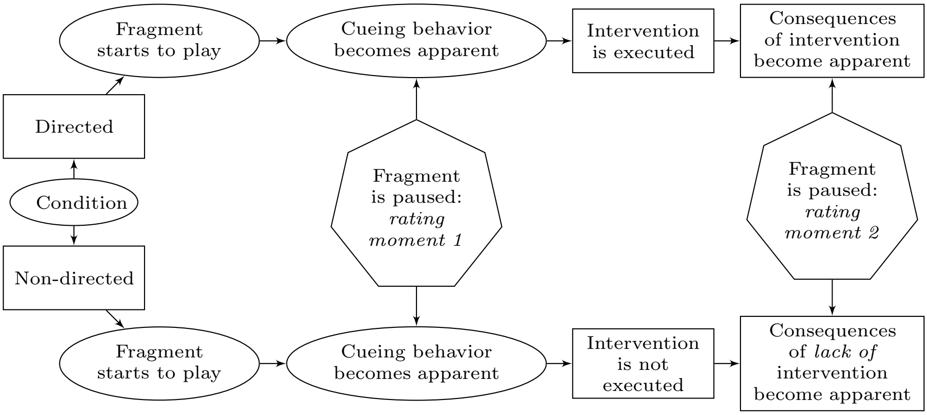

Fragment starts to play
Cueing behavior becomes apparent
Intervention is executed
Consequences of intervention become apparent
Directed
Condition
Non-directed
Fragment is paused: *rating moment 1*
Fragment is paused: *rating moment 2*
Fragment starts to play
Cueing behavior becomes apparent
Intervention is not executed
Consequences of *lack of* intervention become apparent

Fig. 1. A graph of the procedure during the experiment

cue for intervention, the fragment was paused and the raters were asked to rate the learning value (rating moment 1). Subsequently, the fragment was continued and paused again at the time the result of the intervention (or the lack thereof) became apparent. The raters were again asked to rate the learning value (rating moment 2). A diagram of the procedure can be found in Fig. 1.

To test and enhance agreement among raters, they were presented with a practice set of 16 video fragments. The raters were encouraged to discuss their judgments in between series to reach consensus on how to value a learning situation. After the practice set, the experiment proper started, by presenting the test set consisting of twenty video fragments to the raters. The raters were not allowed to discuss their judgments, nor could they see each other's judgments. After the test set, the raters participated in a group discussion about their experiences with scenario-based training and their opinions about the video fragments.

2.4 Analysis

An intra-class correlation analysis was performed to assess inter-rater reliability. A repeated measures ANOVA was used to compute the effects of direction upon the rated learning value of the scenario.

3 Results

Data Exploration and Inter-rater Reliability. Forty ratings per rater (two rating moments for a total of twenty fragments) were entered into the analysis. The consistency intra-class correlation coefficient was 0.694 for average measures ($p<.001$). An inter-rater agreement between 0.60 and 0.79 is considered substantial [6], therefore we consider these data to be appropriate for further analysis.

Table 1. Results of the repeated measures analysis *) p <.05 **) p <.01 ^{1}) one-tailed

effect	F	effect size (partial η^2)	power
director (presence vs absence)^1	13.847**	.735	.841
startup variation (supportive vs challenging)	11.043*	.688	.757
director (presence vs absence) * rating moment^1	27.339**	.845	.984

Repeated Measures Analysis. In order to test whether the interventions of the director had an effect on learning value, rated learning values were entered into a repeated measures analysis with two independent factors: director (presence vs absence) and startup variation (a scenario starting in the supportive vs challenging behavior variation). The results of this analysis are shown in Table 1.

A main effect of direction was found (F(1,5)=13.85; p<.01, one-tailed). Examination of this effect showed that the directed fragments received a significantly higher learning value (M=1.08; SE=.31) than the non-directed fragments (M=.35; SE=.23). A second main effect showed a significant difference between the learning value assigned to the two startup conditions (F(1,5)=11.04; p<.01, two-tailed). Overall, the video fragments in the supportive startup condition received a higher learning value (M=.98; SE=.31) than those in the challenging startup condition (M=.45; SE=.22).

Our main interest is the effect of an intervention on the situation's learning value. Therefore the differences between the director conditions (present vs absent) at rating moment 2 are of importance. It is expected there are no differences between the two conditions at rating moment 1. A significant interaction effect between director (presence vs absence) and rating moment (prior to vs after the cue for intervention) (F(1,5)=27.34; p<.01, one-tailed test), showed that indeed there was no significant difference between the directed and the non-directed condition at rating moment 1 (M=.60 vs M=.43, respectively). However, if an intervention was executed at rating moment 2 (director present), the learning value was significantly higher than when no intervention had taken place (director absent) (M=1.55 vs M=.27, respectively). The means belonging to this interaction effect can be found in the row 'overall' of Table 2.

To find out whether the beneficial effect of the director's interventions is equal for both directions of interventions (from supportive to challenging or vice versa), one-tailed 95% confidence intervals of the means were computed for both startup

Table 2. Mean rated learning value (SE) *)p <.05, one-tailed

	director present		director absent	
	moment 1	moment 2	moment 1	moment 2
challenging startup	.433 (.336)	1.467* (.470)	.233 (.285)	-.333 (.276)
supportive startup	.767 (.391)	1.633* (.363)	.633 (.336)	.867 (.418)
overall	.600 (.306)	1.550* (.394)	.433 (.262)	.267 (.324)

conditions. The interaction effects were significant (p<.05, one-tailed) for both directions of intervention, (see also Table 2), although the effect was stronger for supportive interventions (changing the actor behavior from challenging to supportive).

4 Discussion

The goal of the present study was to investigate the effects of interventions upon the learning quality of a scenario. We created scripts for a director specifying when and how to intervene. Interventions consisted of adaptations in the behavior of the actors (NPCs) and were implemented on-line, i.e. while the scenario unfolded. Video recordings of directed and non-directed training scenarios were shown to experienced instructors, who were asked to rate the learning value of the presented situations. Instructors were naive with respect to the purpose and design of the experiment.

Results confirmed our hypothesis. The rated learning value of scenarios that proceed undirected, without adaptation, were at a fairly low level both halfway and at the end of the scenario. In contrast, the learning quality of directed scenarios improved significantly as a result of the interventions directing the actors to behave appropriately to the performance level of the trainee. Thus, overall, interventions improve the learning value of scenarios. If we examine these results more closely, split for supportive and challenging startup conditions, it becomes clear that scenarios that started in the supportive mode also offer some learning opportunities in the absence of a director. Even though the trainee could use an extra challenge, the mere practice of already acquired skills is still considered useful. However, in the directed condition, it becomes possible to create an extra challenge for the trainee, which results in an even higher learning value. A different pattern is found for the scenarios that started in the challenging mode. For these scenarios, the learning value drops dramatically over time when there is no director present to adjust the scenario. However, in the presence of the director, support is given to the trainee, thereby most likely saving the trainee from losing track and motivation and increasing the learning value of the training.

In a group interview conducted after the experiment, we explained the purpose and design of the study to the instructors and asked them for their experiences in their everyday work. The instructors stated that they find it hard to successfully intervene once they notice that a scenario loses track. They argue that they do realize it when a training situation requires intervention, but that they find it hard to specify beforehand what cues indicate this need. A more practical problem that they put forward is that - in their experience - participating actors tend to be unaware of what is needed, and that it is difficult for instructors to bring across appropriate adjustments to the actors while the scenario is playing. Instructors therefore consider it important to have appropriate and practical instruments to execute the necessary control over their training scenarios. They added to welcome this type of studies to accomplish this need.

In this study we explicitly described cues based upon trainees' responses to specify different types of interventions. These interventions proved to be beneficial to the learning value of the scenario. A next step would be to further refine the different types of interventions a director can execute and to conceptualize the knowledge that is needed to implement such interventions. In the end, the goal is to develop automated systems that formalize relationships between events, learning objectives, trainee behaviors and NPC behaviors to create autonomous, adaptive and effective training scenarios.

Acknowledgements. The authors would like to thank Stichting BHV Nederland for their cooperation in this experiment.

References

1. Baker, R.S.J., D'Mello, S.K., Rodrigo, M., Mercedes, T., Graesser, A.C.: Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive-affective states during interactions with three computer-based learning environments. Int. J. Hum-Comp. St. 68(4), 223–241 (2010)
2. Blumberg, B., Galyean, T.: Multi-level Control for Animated Autonomous Agents: Do the Right Thing.. Oh, Not That.. In: Creating Personalities for Synthetic Actors, Towards Autonomous Personality Agents, pp. 74–82. Springer, Heidelberg (1997)
3. van den Bosch, K., Harbers, M., Heuvelink, A., van Doesburg, W.: Intelligent agents for training on-board fire fighting. In: Duffy, V.G. (ed.) ICDHM 2009. LNCS, vol. 5620, pp. 463–472. Springer, Heidelberg (2009)
4. Cannon-Bowers, J., Burns, J., Salas, E., Pruitt, J.: Advanced Technology in Scenario-Based Training. In: Making Decisions Under Stress: Implications for Individual and Team Training, ch, pp. 365–374. APA, Washington DC (1998)
5. Kirschner, P., Sweller, J., Clark, R.: Why minimal guidance during instruction does not work. Educational Psychologist 41(2), 75–86 (2006)
6. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics 33(1), 159–174 (1977)
7. Magerko, B., Wray, R.E., Holt, L.S., Stensrud, B.: Customizing interactive training through individualized content and increased engagement. In: I/ITSEC (2005)
8. Marsella, S.C., Johnson, W.L., LaBore, C.: Interactive pedagogical drama. In: 4th International Conference on Autonomous Agents, pp. 301–308 (2000)
9. Merrill, M.D.: First principles of instruction. ETR&D 50(3), 43–59 (2002)
10. Miao, Y., Hoppe, U., Pinkwart, P.: Situation creator: A pedagogical agent creating learning opportunities. In: 13th International Conference on AIED, pp. 614–617 (2007)
11. Murray, T., Arroyo, I.: Toward measuring and maintaining the zone of proximal development in adaptive instructional systems. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 749–758. Springer, Heidelberg (2002)
12. Oser, R.L.: A structured approach for scenario-based training. In: 43rd HFES Annual Meeting, pp. 1138–1142 (1999)
13. Rieber, L.P.: Seriously considering play. ETR&D 44(2), 43–58 (1996)
14. Riedl, M.O., Stern, A., Dini, D., Alderman, J.: Dynamic experience management in virtual worlds for entertainment, education, and training. ITSSA, Special Issue on Agent Based Systems for Human Learning 4(2), 23–42 (2008)
15. Vygotsky, L.S.: Mind in Society: the Development of Higher Psychological Processes. Harvard University Press, Cambridge (1978)
16. Westra, J., Hasselt, H.v., Dignum, F., Dignum, V.: Adaptive serious games using agent organizations. Agents for Games and Simulations, 206–220 (2009)

Classroom Video Assessment and Retrieval via Multiple Instance Learning

Qifeng Qiao and Peter A. Beling

Department of Systems and Information Engineering
University of Virginia, USA
{qq2r,pb3a}@virginia.edu

Abstract. We propose a multiple instance learning approach to content-based retrieval of classroom video for the purpose of supporting human assessing the learning environment. The key element of our approach is a mapping between the semantic concepts of the assessment system and features of the video that can be measured using techniques from the fields of computer vision and speech analysis. We report on a formative experiment in content-based video retrieval involving trained experts in the Classroom Assessment Scoring System, a widely used framework for assessment and improvement of learning environments. The results of this experiment suggest that our approach has potential application to productivity enhancement in assessment and to broader retrieval tasks.

1 Introduction

Classroom assessment is a topic of increasing interest among education practitioners, researchers, and policy makers. Recent years have seen a number of observation and assessment protocols developed, fielded, and tested as part of large-scale effectiveness experiments. The Measure of Effective Teaching (MET) project, for example, is designed to help educators and policy makers identify and support good teaching by improving the quality of information about teacher practice. MET has used approximately 500 assessment experts, known as coders, to rate more than 23,000 hours of videotaped lessons using standard classroom observation protocols. Recent years also have seen advances in the fields of computer vision and machine learning, to the point where it is reasonable to consider a role in the classroom assessment process for automatic interpretation of video, audio, and other sensor information. In the near-term, this role is likely to be one of supporting, rather than supplanting, human coders by providing filtering or pre-screening services to distill large volumes of video down to those portions that are likely to be most productive or informative for assessment.

We assert that content-based video retrieval is a core technical problem for the development of filtering schemes. The aim in content-based retrieval is to use training interaction with a human user to gain an understanding of the media content that is of interest to the user. Content-based image retrieval has been widely studied, and recently there has been some extension of this work to video, with focus on entertainment media like television programs and feature films.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 272–279, 2011.

Classroom videos have a number of idiosyncratic properties that present both challenges and opportunities in retrieval. Difficulties in interpretation arise from the complicated and dynamic nature of classroom events, occlusion among students, and pragmatic aspects of human communication. On the other hand, the structured environment of a classroom means that, within the context of a particular assessment methodology, it may be possible to decompose dynamic events into a set of simpler components that are amenable to machine measurement.

In this paper, we propose the Classroom Evaluation and Video Retrieval (CLEVER) system, which is a multiple instance learning (MIL) approach to content-based retrieval of classroom video for the purpose of supporting human assessing the learning environment. The learning aspects of CLEVER are similar to MIL and other approaches that have been used for content-based image and video retrieval (cf. [1,2,3]), but differ in that instances and the feature space are defined in ways that exploit the structure of classroom learning and the nature of the assessment system. The key element in CLEVER is a mapping between the semantic concepts of the assessment system and features of the video that can be measured using techniques from the fields of computer vision and speech analysis. We work with a single assessment methodology, the Classroom Assessment Scoring System (CLASS). CLASS is a theoretically-driven and empirically-supported conceptualization of classroom interactions [4] in which trained coders produce assessment scores on the basis of observation of the classroom, either in person or from a video recording or broadcast. The framework encompasses a consultive process in which teachers used annotated video, produced by the coders using a structured process, as the basis for a self-improvement effort [5]. CLASS has been widely adopted, earning places in both Head Start and MET assessment projects. The CLASS methodology centers on observation of teacher and student actions and interactions, a behavioral orientation that tends to align well with machine interpretation of video, particularly in comparison with assessment approaches that focus on instructional content.

The remainder of the paper is organized as follows: In Section 2, we present a mapping between the structure of CLASS and concepts that have associated measurements created through automated processing of video and audio. We also describe the multiple instance framework that is the basis for our learning method. In Section 3, we report on the use of CLEVER in a formative experiment in content-based video retrieval involving a group of expert CLASS coders. Finally, in Section 4, we offer conclusions and suggestions for future research.

2 Video Understanding in CLASS

The CLASS framework is a theoretically-driven and empirically validated conceptualization of classroom interactions [6,5,4]. CLASS embodies a latent structure for organizing classroom activity in three domains: *emotional support*, *classroom organization*, and *instructional support*. Each domain is composed of several dimensions defined semantically and scored quantitatively [4]. To take one example, the dimension productivity, within *classroom organization* can be classified into three levels: low with a score of 1 or 2, medium with a score of 3, 4 or

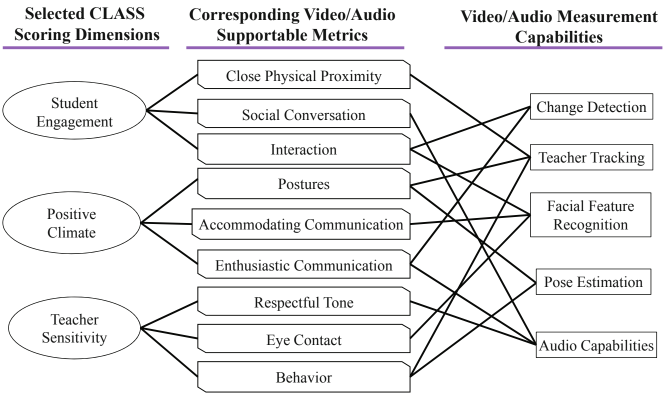


Fig. 1. Example of semantic gap bridging between CLASS and automatic measurement

5 and high with a score of 6 or 7. The high level would be assigned to a classroom in which students are oriented, with respect to expectations and tasks, and transitions from one activity to another happen quickly and efficiently. CLASS coders rely on their judgment and reasoning intelligence to assign scores.

2.1 Feature Extraction

An ideal video retrieval system would allow one to query on high-level concepts, often called *semantic concepts*. As an example, one might like to ask a retrieval system for all classroom videos in which the teacher appears to be frustrated with student progress or those that present a high level of energy on the part of the students. Automated retrieval systems, however, must work with much lower level concepts, such as pixel intensity and pixel change or sound frequency, that can be measured from video and audio using algorithms. The principal challenge in video retrieval is to bridge the gap between semantic concepts and *measurable concepts*, which are the features we can handle using automated video interpretation. In our case, the scoring dimensions of CLASS are the relevant semantic concepts. As they relate to classroom assessment, we call these *semantic assessment concepts*. A good semantic-sensitive video content representation framework emphasizes features that are more capable of representing the semantic assessment concepts and avoids performing uncertain feature extraction. For example, the semantic assessment concepts of instructional aiding materials, lecture presentation, and student engagement are implicitly related to visual analyses, including the detection of moving objects, high luminosity regions, human faces or skin, and blocks of changing pixels, as well as audio analyses, such as detection of individual and dialog speech.

As illustrated in Fig. 1, we propose bridging the gap between semantic assessment concepts and measurable concepts in two steps, first linking semantic assessment concepts with video/audio metrics from CLASS dimensions, and then relating the video/audio metrics with feature variables that can be extracted by

Table 1. Video feature definition used to construct the feature vector for each video

Low-level Attribute	**Description**
Color Histogram	Global color represented in HSV space
Co-occurrence Texture	Global texture containing entropy, energy, and contrast.
Motion Intensity	Average difference of pixel values.
Teacher Position	Teacher's position in the classroom.
Moving Velocity	Mean, maximum, and minimum velocity of detected movement.
High-level Attribute	**Description**
Salient Object	Image regions with homogeneous color or texture.
Pose Orientation	Teacher's orientation: toward students or toward blackboard.
Teacher Gesture	Detection and recognition from a predefined gesture set.
Dynamic Event	Student presentations, group discussion.
Audio Attribute	**Description**
Silence Detection	Silence on the part of the teacher
Pitch	Frequency of speech
Dialog Talking	Question and answer events.

available automatic measurement techniques. Many video processing techniques we need, such as topical detection, synchronization, summarization and editing, have been addressed for content analysis of classroom videos [7]. These tasks depend on static analysis of image features, e.g. detection of the slides using color background detection [8], key-frame detection using similarity measurement and scene-break detection using image differences and color histograms [9]. Making use of the relationship between CLASS and video/audio measurement capability, we characterize classroom videos using the attributes in Table 1. The measurement of high-level attribute requires combination of multiple feature extraction techniques. For example, group discussion events are found using the lower-level features of speech detection and motion intensity estimation.

2.2 Multiple Instance Structure and Learning

Most methods of shot boundary detection focus on segmenting the video clip at frames corresponding to transitions, either abrupt (cuts) or gradual (dissolves and fades). These shot detection techniques have limited application in our context because scene scenarios of classroom videos are relatively stationary and unvaried as measured by global low-level attributes, such as color histogram and textures. Moreover, in classroom video a measurable concept may appear in different temporal locations, implying the concept is represented by a set of small video sequences that are highly correlated. We propose a new framework that depends on an interpretation of the assessment protocol that varies according to individual perceptions. In Fig. 2, we show the traditional structure for

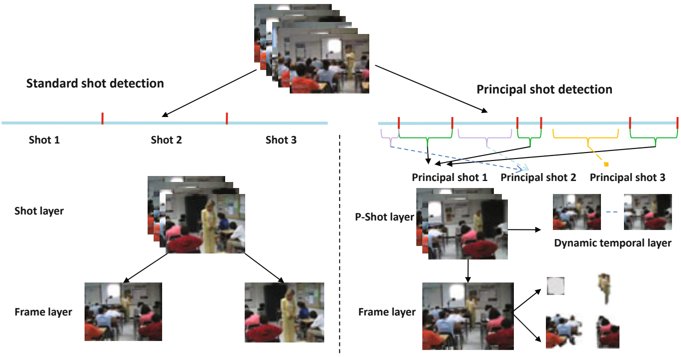


Fig. 2. Comparison of structures for video understanding. The left interprets the traditional structure and the right displays the principles of multiple instance structure.

shots, consisting of contiguous temporal regions, compared with our proposed *principal shot structure* in which shots are composed by aggregating segments from across the video that share a common semantic assessment concept. This structure depends on both static and dynamic video patterns for video content representation and feature extraction. We expect such shot detection structure to enhance the quality of features since it gives rise to a hierarchical analysis of video content and an understanding of semantic objects and temporal events.

We use MIL as the primary method for relating high-level concepts of interest to the user to measurable concepts. MIL is a variation of supervised learning in which there is ambiguity associated with labels [10]. Instead of receiving labels for each instance, the training set is composed of a number of *bags*, each of which is comprised of a set of instances. In binary MIL, a bag is labeled positive if it contains at least one positive instance, and is labeled negative otherwise. Given labels for a set of training bags, the learning algorithm aims to discover the regions of the feature space associated with positive labels, with the particular goal of labeling individual bags and instances correctly. A variety of algorithms have been developed for MIL, including [1,11,10].

MIL has been successfully applied in the field of localized content based image retrieval (LCBIR) [1,12], where the goal is to rank images according to their similarity to training images that a user has labeled as being of interest. In LCBIR, images are the bags and contiguous blocks of pixels are the instances. In our application, video clips are the bags and principal shots are the instances. We construct principal shots by first segmenting each video clip into micro clips (e.g. a segment of 10 seconds length). We then use adaptive k-means clustering to group similar micro clips, with each group forming a principal shot. The general learning process includes: measurement and feature extractions, video segmentations, clustering of micro clips, feature aggregation for principal shots, MIL, and calibration with ground truth data.

Table 2. Performance Accuracy with respect to i-th subject (Si, $i = 1, 2, \cdots, 10$)

Video	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
Course A	0.904	0.645	0.635	0.794	0.734	0.763	0.616	0.768	0.9	0.612
Course B	0.898	0.524	0.652	0.636	0.622	0.652	0.668	0.678	0.994	0.686
Mixed data	0.901	0.585	0.644	0.715	0.678	0.707	0.642	0.723	0.947	0.649

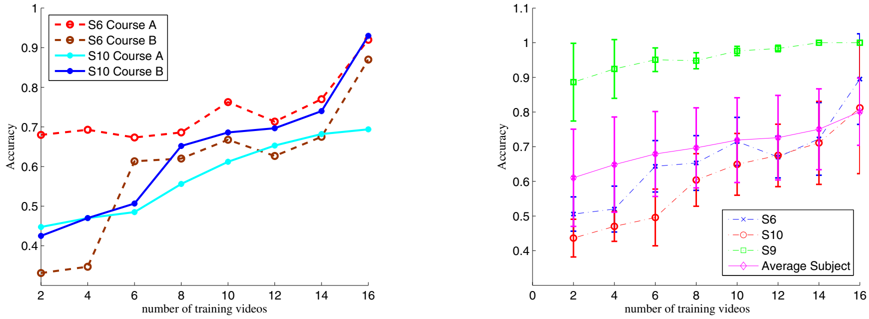
3 Experimentation with Human Evaluators

As a formative experiment, we conducted an experiment with 10 expert CLASS coders. Coders asked to view 40 video clips, each three minutes in length. Clips were taken from video recordings of two junior-level classes in Systems Engineering at the University of Virginia (*Course A* and *Course B*). Coders were instructed to assign either a positive or a negative label to each clip, giving a positive label only if, in their individual judgment, the clip provided significant useful information for the purposes of CLASS assessment. Coders were further instructed to evaluate each clip in isolation from the other clips, so that behavior or activity that had been seen before in the sequence was just as deserving of a positive label as when seen for the first time. Coders were free to formulate their own interpretations of CLASS in relation to the labeling instructions. Perhaps as a result, labels varied greatly across the subjects, with a Fleiss' kappa [13] value of 0.146 for Course A and 0.125 for Course B.

For the basic experiment of learning labels, classification accuracy is defined as the proportion of the correctly predicted labels in the testing dataset. We estimated classification accuracy on the basis of 100 replications, each with equal-sized, randomly chosen training and testing sets. Results are shown in Table 2. The large variation in accuracy across subjects is likely a reflection of the variation in semantic concept reflected in the subjects' choices of labels.

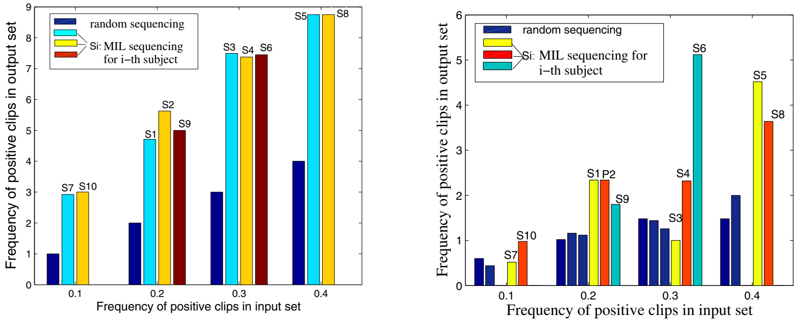
To investigate consistency of predictive performance, we estimated classification accuracy as a function of training set size. Fig. 3 shows this relationship for three coders. In these examples CLEVER performance on individual user is consistent, since accuracy is increasing in the number of training examples used. Average performance across the 10 subjects exhibits the same trend.

To investigate potential filtering roles for CLEVER, we used the label data from the coders in computational experiments on productivity. The setting for these experiments is a hypothetical scenario in which a coder is viewing a sequence of video clips. The machine learning task is to use labels from the first 10 minutes of viewing to reorder the remaining clips with the goal of maximizing the number of positive clips viewed during a 10-minute performance period. Figure 4 (a) compares the expected number of positive clips viewed under a random order with that expected from a reordering done with an accuracy equal to the estimated true positive probability achieved by CLEVER label learning experiments described above. The reordering outputs the predicted positive videos and we assume there is enough predicted positive videos for viewing in 10 minutes.



(a) (b)

Fig. 3. Plots of Performance Accuracy. Fig (a) shows the mean classification accuracy Fig (b) displays the mean accuracy and the standard deviation, where *average subject* represents the accuracy that is averaged across ten subjects.



(a) Theoretic computation (b) Experimental simulations

Fig. 4. Plots of productivity. The productivity bars for ten subjects are shown in groups with regard to the frequency of positive clips in input set.

Fig. 4 (b) shows the results for a similar computation that, instead of estimated accuracies, used 100 replications of the simulation on a boosted testing data set containing 100 positive and 100 negative clips.

4 Discussion and Future Work

CLEVER fuses state-of-the-art machine learning algorithms with advanced assessment concepts from the education community. The results of formative experiments on CLASS coders are encouraging. Accuracy in label prediction is substantially greater than would be expected from random performance and, as our productivity experiments show, would be sufficient to support filters that would reduce human viewing load by a factor of 2 or more. It is also worth noting that other users, such as teachers themselves, might benefit from the content-based retrieval capability of CLEVER as part of a self-improvement or reflective process.

The feature set that we used could be improved through the addition of more audio characteristics. Furthermore, the integration of video and audio techniques might be the key to extracting higher-level features that underscore interactions between teacher and students, which are known to be critically important elements of classroom assessment.

References

1. Qiao, Q., Beling, P.A.: Localized content based image retrieval with self-taught multiple instance learning. In: Proceedings 2009 IEEE International Conference on Data Mining Workshop, pp. 170–175 (2009)
2. Uijlings, J.R.R., Smeulders, A.W.M., Scha, R.J.H.: Real-time visual concept classification. IEEE Trans. On Multimedia 12(7), 665–682 (2010)
3. Fan, J., Luo, H., Elmagarmid, A.K.: Concept-oriented indexing of video databases: Toward semantic sensitive retrieval and browsing. IEEE Transactions On Image Processing 13(7), 974–992 (2004)
4. Pianta, R.C., La Paro, K.M., Hamre, B.K.: Classroom Assessment Scoring System (CLASS) Manual: K-3. Brookes Publishing (2008)
5. Pianta, R.C., Belsky, J., Houts, R., Morrison, F.: Opportunities to learn in america's elementary classrooms. Science 315, 1795–1796 (2007)
6. Pianta, R.C., Howes, C., Burchinal, M., Bryant, D.M., Clifford, R.M., Early, D.M., Barbarin, O.: Features of pre-kindergarten programs, classrooms, and teachers: Do they predict observed classroom quality and child-teacher interactions? Applied Developmental Science 9(3), 144–159 (2005)
7. Wang, F., Ngo, C.-W., Pong, T.-C.: Lecture video enhancement and editing by integrating posture, gesture, and text. IEEE Trans. on Multimedia 9(2), 397–409 (2007)
8. Mahmood, T.S., Srinivasan, S.: Detecting topical events in digital video. In: MULTIMEDIA 2000 Proceedings of the Eighth ACM International Conference on Multimedia, pp. 85–95. ACM, New York (2000)
9. Ju, S.X., Black, M.J., Minneman, S., Kimber, D.: Summarization of videotaped presentations: Automatic analysis of motion and gesture. IEEE Trans. on Circuits and Systems for Video Technology 8, 686–696 (1998)
10. Thomas, G., Dietterich, R.H., Lozano-Përez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence 1446, 1–8 (1998)
11. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. Advances in Neural Information Processing System 15, 561–568 (2003)
12. Rahmani, R., Goldman, S.A., Zhang, H., Cholleti, S.R., Fritts, J.E.: Localized content based image retrieval. IEEE Trans. on Pattern Analysis and Machine Intelligence (2008)
13. Fleiss, J.L.: Measuring Nominal Scale Agreement Among Many Raters. Psychological Bulletin 76(5), 378–382 (1971)

Faster Teaching by POMDP Planning

Anna N. Rafferty[1], Emma Brunskill[1],
Thomas L. Griffiths[1], and Patrick Shafto[2]

[1] University of California, Berkeley, CA 94720, USA
[2] University of Louisville, KY 40292, USA

Abstract. Both human and automated tutors must infer what a student knows and plan future actions to maximize learning. Though substantial research has been done on tracking and modeling student learning, there has been significantly less attention on planning teaching actions and how the assumed student model impacts the resulting plans. We frame the problem of optimally selecting teaching actions using a decision-theoretic approach and show how to formulate teaching as a partially-observable Markov decision process (POMDP) planning problem. We consider three models of student learning and present approximate methods for finding optimal teaching actions given the large state and action spaces that arise in teaching. An experimental evaluation of the resulting policies on a simple concept-learning task shows that framing teacher action planning as a POMDP can accelerate learning relative to baseline performance.

1 Introduction

When assisting a student, a teacher must both diagnose a student's understanding and use a teaching policy for deciding on the best pedagogical action to take next. There has been substantial interest in the cognitive science, education, and intelligent tutoring systems communities in modeling and tracking student learning. In particular, there have been a number of results demonstrating the benefit of taking a Bayesian probabilistic approach (see, e.g., [4,6,7,17]). However, there has been much less work on how to compute an automated teaching policy that leverages a probabilistic learner model in order to achieve a long-term teaching objective, which is the focus of this paper.

We use a probabilistic, sequential, decision-theoretic approach to compute individualized teaching policies. More specifically, we employ a Bayesian probabilistic representation over the learner's (hidden) knowledge, and embed this within a powerful framework known as a partially-observable Markov decision process (POMDP) [14]. Given a learning objective and a set of models describing the learning process, POMDPs provide a framework for computing an optimal teaching policy that maximizes the objective. Though POMDPs are related to other decision-theoretic approaches used in previous education research, they are more powerful in two key respects. First, POMDPs can use sophisticated models of learning, rather than assuming learners' understanding can be directly observed or approximated by a large number of features (as in [1,5]). Second,

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 280–287, 2011.

in contrast to approaches that only maximize the immediate benefit of the next action [6,10], POMDPs reason about both the immediate learning gain and the long-term benefit to the learner after a particular activity.

Though POMDPs offer an appealing theoretical framework, there are often significant obstacles to practical implementation. Specifically, planning teaching requires modeling learning, and richer, more realistic models of learning lead to computational challenges for planning. In this paper we develop an approach for computing approximate POMDP policies, which makes it feasible to use these policies with human learners. In addition, we examine three different models of concept learning, and demonstrate how, given the same learning objective, these lead to qualitatively different teaching policies. We explore the impact of these varying policies in an example concept-learning task. While there exist a few recent papers exploring the use of POMDPs to compute teaching policies [2,3,9,16], to our knowledge ours is the first paper to demonstrate with human learners that POMDP planning results in more efficient learning than baseline performance and the first to explore the impact of different models of learning on the computed policies.

2 Modeling Teaching as a POMDP

POMDP planning is used to compute an optimal conditional policy for selecting actions to achieve a goal, in absence of perfect information about the state of the world. Briefly, a POMDP consists of a tuple $\langle S, A, Z, p(s'|s,a), p(z|s,a), r(s,a), \gamma \rangle$ where S is a set of states s, A is a set of actions a, and Z is a set of observations z [14]. The transition model $p(s'|s,a)$ gives the probability of transitioning from state s to state s' after taking action a. The observation model $p(z|s,a)$ indicates the probability of an observation z given that action a is taken in state s. The planner's probability distribution over the current state is the *belief state* and can be updated using Bayesian filtering. The cost model $r(s,a)$ specifies the cost of taking action a in state s, and the discount factor γ represents the relative harm of immediate costs versus delayed costs. POMDP planning computes a policy that specifies which action to take, given a belief state, in order to minimize the expected sum of (discounted) future costs.

Many teaching tasks can be easily formalized within this framework. We model the learner's knowledge as a state s. The transition model then describes how teaching actions stochastically change the learner's knowledge, and the observation model indicates the probability that a learner will give a particular response to a tutorial action, such as a question, based on her current understanding. We will shortly describe several alternate learner models that employ different state representations, transition models, and observation models.

In the remainder of the paper, we consider how this framework can be applied in a concept learning task. In such a task, we set the cost for each action to be the expected amount of time for the learner to complete the activity, and when the learner knows the correct concept, the action cost drops to zero. As a consequence, the computed policies select actions to minimize the expected time for the learner to understand the concept. The space of tutorial actions may vary widely based on the domain being taught. Within concept-learning, it is natural to consider three types of actions: *examples*, *quizzes*, and *questions with feedback*. *Example* and *quiz* actions are equivalent to the *elicit* and *tell* pedagogical actions that have been used previously in intelligent tutoring systems [5]. The resulting POMDP can be used to find the optimal policy for teaching the learner the concept, taking into account the learner's responses.

3 Learner Models

We consider three learner models, inspired by the cognitive science literature, that correspond to restrictions of Bayesian learning. While the models we describe are only rough approximations of human concept learning, we will show that they are still sufficient to enable us to compute better teaching policies.

Memoryless Model: We first consider a model in which the learner's knowledge state is the single concept she currently believes is correct, similar to a classic model of concept learning proposed by Restle [11]. In this model, the learner does not explicitly store any information previously seen. If an action is a *quiz* action, or if the provided evidence in an *example* or *question with feedback* action is consistent with the learner's current concept, then her state stays the same. If the action contradicts the current concept, the learner transitions to a state consistent with that action, with probability proportional to the prior probability of that concept. The observation model is deterministic: when asked to provide an answer to an equation, the learner provides the answer consistent with her current beliefs. This model underestimates human learning capabilities, and thus provides a useful measure of whether POMDP planning can still accelerate learning when a pessimistic learner model is used.

Discrete Model with Memory: The key limitation of our first model is its lack of memory of past evidence. A more psychologically plausible model is one in which learners maintain a finite memory of the past M actions. Like the memoryless model, this model assumes that the learner stores her current guess at the true concept, and this guess is updated only when information is shown that contradicts the guess. In this case, the learner shifts to a concept that is consistent with the current evidence and all evidence in the M-step history. The transition probability is again proportional to the initial concept probability, and the observation model is deterministic based on the learner's current guess.

Continuous Model: A more complex, but natural, view of learning is that the learner maintains a probability distribution over multiple concepts [15]. In this case the state is a $|C|$-dimensional, continuous-valued vector that sums to 1, where C is the set of possible concepts. The state space S is an infinite set of all such vectors, the simplex $\Delta_{|C|}$. The transition function assumes that for *quiz* actions, each state transitions deterministically to itself. For *example* and *question with feedback* actions, state dimensions for concepts that are inconsistent with the provided information are set to zero. The full joint transition probability is then re-normalized. The observation model assumes the learner gives answer a_n to a question with probability equal to the amount of probability she places on concepts that have a_n as the correct answer for this question.

To improve the robustness of our policies to the coarse learner models we employ, all models include two extra parameters, ϵ_t and ϵ_p. ϵ_t corresponds to the probability that the learner ignores a given teaching action, resulting in the learner not transitioning to a new concept, while ϵ_p corresponds to the probability that the learner produces an answer inconsistent with her current guess.

4 Finding Policies

Our goal is to compute a policy that selects the best action given a distribution over the learner's current knowledge state, the belief state. Offline POMDP planners compute in advance a policy for each belief in the set of potential beliefs.[1] However, since this set grows exponentially with the number of states, offline approaches cannot scale to the large size of common teaching domains. We instead turn to online POMDP forward search techniques, which have proven promising in other large domains (see [13] for a survey). We compute the future expected cost associated with taking different actions from the current belief state by constructing a forward search tree of potential future outcomes. This tree is constructed by interleaving branching on actions and observations. After the tree is used to estimate the value of each action for the current belief, the best pedagogical action is chosen. The learner then responds to the action, and this response, plus the action chosen, is used to update the belief representing the new distribution over the learner's knowledge state. We then construct a new forward search tree to select a new action for the updated belief.

While forward search solves some of the computational issues in finding a policy, the cost of searching the full tree is $O((|A||Z|)^H)$, where H is the task horizon (i.e., the number of sequential actions considered), and requires an $O(|S|^2)$ operation at each node. This is particularly problematic as the size of the state space may scale with complexity of the learner model: the memoryless model has a state space of size $|C|$, while the discrete model with memory has state space of size $|C||A|^M$ and the continuous model has an infinite state space. To reduce the number of nodes we must search through, we take a similar approach to [12] and restrict the tree by sampling only a few actions. Additionally, we limit H to control the depth of the tree and use an evaluation function at the leaves.

Since the belief state in the continuous model is a distribution over an infinite set of states, we approximate the belief state for this model to make inference tractable. We represent the belief state as a weighted set of probabilistic particles and update these particles based on the transition and observation models (see [8] for more about this technique, known as *particle filtering*). If no particles are consistent with the current observation, we reinitialize the belief state with two particles: one with a distribution induced by rationally updating the prior using all previous evidence and one with a uniform distribution.

[1] Most state-of-the-art offline algorithms try to compute a policy over a subset of the reachable subspace, but this is still typically a very large number of beliefs.

5 Empirically Testing Optimized Teaching Policies

POMDP planning provides a way to select actions optimally with respect to a particular learning objective. However, given the simplifications made for computational tractability and that our learner models only approximate true learners, it is necessary to empirically test whether this framework results in more efficient learning. We demonstrate its effectiveness by teaching learners "alphabet arithmetic," a concept-learning task in which letters are mapped to numbers. While this task is artificial, it provides a preliminary evaluation of POMDP planning for problem selection and shares several important characteristics with real teaching domains: it is rich enough that learners may have misconceptions and that we expect some teaching policies to be more effective than others.

In alphabet arithmetic, learners infer a mapping from letters to numbers from a set of equations using letters. For *example* actions, learners are shown an equation where two distinct letters sum to a numerical answer. For instance, A could be mapped to 0 and B to 1, and one might show the learner the equation $A + B = 1$. *Quiz* actions leave out the numerical answer and ask the learner to give the correct sum. *Questions with feedback* combine these two actions. We assume learners have a uniform prior over mappings.

5.1 Methods

Participants. A total of 40 participants were recruited online and received a small amount of monetary compensation for their participation.
Stimuli. All participants were randomly assigned three mappings between the letters A–F and the numbers 0–5. These mappings were learned in succession.
Procedure. Participants were assigned to either the *control condition*, in which teaching actions for all mappings were chosen randomly, or to the *experimental condition*. Each participant in the experimental condition experienced all three of the teaching policies in random order, one for each mapping learned. The experiment consisted of a sequence of teaching and assessment phases. In each teaching phase, a series of three teaching actions was chosen based on condition. After each teaching phase, participants completed an assessment phase in which they were asked to give the number to which each letter corresponded. Teaching of a given mapping terminated when the participant completed two consecutive assessment phases correctly or when 40 teaching phases had been completed. Within all phases, the equations the participant had seen were displayed on-screen, and participants could optionally record their current guesses about which letter corresponded to which number.
Computing policies. We estimated the median time to complete each action type from the control participants: *example* actions took 7.0s, *quiz* actions took 6.6s, and *question with feedback* actions took 12s. These values were the cost for each action in the experimental condition. When computing the action values within the forward search tree, we set the cost for a leaf node to be the probability of not passing the assessment phase multiplied by $10 \cdot \min_a r(a)$, a scaling of the minimum future cost.

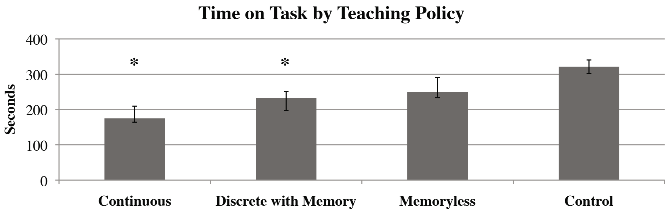


Fig. 1. Median time to learn each mapping, by policy type; error bars correspond to bootstrapped 68% confidence intervals (equivalent to one standard error). Asterisks indicate that the policies based on the continuous model and the discrete model with memory result in significantly faster learning than the control.

We set ϵ_t, the probability of ignoring a teaching action, and ϵ_p, the probability of making a production error when answering a question, by finding the values that maximized the log likelihood under a given model of the data from the control condition.[2] For forward planning, we limited the lookahead horizon to two and stopped planning after three seconds.[3] There was delay of three seconds between actions in all conditions to allow time for planning.

5.2 Results

We compared the number of phases as well as the time participants took to learn each mapping. Initial inspection showed that the distribution of learning times exhibited a long right tail, so we analyzed results using medians, which are more robust than means to outliers and non-symmetric distributions. There was no significant within-subjects difference in the amount of time or number of phases to learn the first, second, or third mapping (Kruskal-Wallis $p > 0.8$).

Overall, participants taught by POMDP planning took significantly fewer phases to learn each mapping than participants in the control condition (3 phases versus 4, Kruskal-Wallis $p < 0.00005$) and also took significantly less time per mapping (232 seconds versus 321 seconds, Kruskal-Wallis $p < 0.001$); see Figure 1. Planned pairwise comparisons show that all of the POMDP policies resulted in fewer phases to completion than the control, and all POMDP policies but the policy from the memoryless model resulted in significantly faster learning.

Differences in policies occurred based on the learner model used; see Figure 2 for part of one policy. The policy from the memoryless learner model repeats specific example actions more often than the other policies since previous actions

[2] The calculation was performed using the EM algorithm for the two discrete models and using a forward filtering approximation for the continuous model. We found the following values: memoryless model: $\epsilon_t = 0.15$ and $\epsilon_p = 0.019$; discrete model with memory: $\epsilon_t = 0.34$ and $\epsilon_p = 0.046$; and continuous model: $\epsilon_t = 0.14$ and $\epsilon_p = 0.12$.

[3] Policies for the first 9 actions were precomputed with 10 actions sampled at each level. Later actions were precomputed by sampling the following number of actions at each level: 7 and 6 actions for the memoryless model; 8 and 8 actions for the discrete model with memory; and 4 and 3 actions for the continuous model. 16 particles were used for the continuous model, and $M = 2$ for the discrete model with memory.

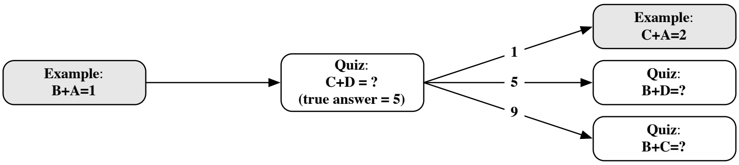


Fig. 2. Part of a policy from the discrete model with memory. Possible student answers to the quiz are indicated on the arrows; some are omitted. Based on the student's response, the action after the quiz may correct a misconception, try to better misdiagnose the cause of an incorrect answer, or continue quizzing to try to detect a misconception.

are not stored in memory. The fact that this model did not significantly decrease time to learn suggests that using too pessimistic of a model may be detrimental for problem selection. Overall, policies for this model also asked more questions (39% of actions) than policies for the other models (about 10% of actions). This is because the state of a memoryless learner after an example is known with less certainty since it is constrained only to be consistent with the last example.

Policies for both the discrete model with memory and the continuous model began with six independent equations that fully specify the mapping. This is the policy one might have hand-crafted to teach this task, demonstrating that despite approximations in planning, the POMDP planner finds reasonable teaching policies. Each of the policies for these two models gives examples until there is a high probability the learner is in the correct state, and then asks quiz questions, which are less costly than examples, to detect misconceptions.

6 Conclusion

In this work, we described how teaching can be modeled within the POMDP framework and demonstrated the effectiveness of POMDP planning experimentally. The experimental results showed that different learner models result in systematically different policies and that the policies for the more complex learner models were more effective. This illustrates that optimal problem selection depends not only on knowledge of the domain but also on one's assumptions about the learner. Computational challenges still exist for using POMDP planning: despite sampling only a fraction of possible actions and using very short horizons, planning took $2-3$ seconds per action. However, we believe further speed ups are possible through more sophisticated ways of constructing the forward search tree (such as in [13]). Despite such challenges, our work demonstrates the potential of POMDP planning to lead to empirical improvements in learning. POMDP planning provides a natural framework for problem selection that can use the many existing learner models developed in the ITS community. One question not addressed by the current work is whether POMDP planning can identify policies that improve upon those chosen by actual teachers. In future work, we would like to investigate this question in more realistic learning situations, and investigate integrating these ideas in existing tutoring systems.

Acknowledgements. This work was supported by a National Defense Science and Engineering Graduate Fellowship to ANR, a National Science Foundation

Mathematical Sciences Postdoctoral Fellowship to EB, and National Science Foundation grant number IIS-0845410 to TLG.

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. Brunskill, E., Garg, S., Tseng, C., Pal, J., Findlater, L.: Evaluating an adaptive multi-user educational tool for low-resource regions. In: Proceedings of the International Conference on Information and Communication Technologies and Development (2010)
3. Brunskill, E., Russell, S.: RAPID: A reachable anytime planner for imprecisely-sensed domains. In: Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (2010)
4. Chang, K.-m., Beck, J.E., Mostow, J., Corbett, A.T.: A bayes net toolkit for student modeling in intelligent tutoring systems. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 104–113. Springer, Heidelberg (2006)
5. Chi, M., Jordan, P., VanLehn, K., Hall, M.: Reinforcement learning-based feature selection for developing pedagogically effective tutorial dialogue tactics. In: Proceedings of the 1st International Conference on Educational Data Mining (2008)
6. Conati, C., Muldner, K.: Evaluating a decision-theoretic approach to tailored example selection. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (2007)
7. Corbett, A., Anderson, J.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction 4(4), 253–278 (1995)
8. Doucet, A., de Freitas, N., Gordon, N.: Sequential Monte Carlo Methods in Practice. Springer, New York (2001)
9. Folsom-Kovarik, J., Sukthankar, G., Schatz, S., Nicholson, D.: Scalable POMDPs for diagnosis and planning in intelligent tutoring systems. In: AAAI Fall Symposium on Proactive Assistant Agents (2010)
10. Murray, R., Vanlehn, K., Mostow, J.: Looking ahead to select tutorial actions: A decision-theoretic approach. International Journal of Artificial Intelligence in Education 14(3), 235–278 (2004)
11. Restle, F.: The selection of strategies in cue learning. Psychological Review 69(4), 329–343 (1962)
12. Ross, S., Chaib-draa, S., Pineau, J.: Bayesian reinforcement learning in continuous POMDPs with application to robot navigation. In: Proceedings of the International Conference on Robotics and Automation (2008)
13. Ross, S., Pineau, J., Paquet, S., Chaib-draa, B.: Online planning algorithms for POMDPs. Journal of Artificial Intelligence Research 32(1), 663–704 (2008)
14. Sondik, E.J.: The Optimal Control of Partially Observable Markov Processes. Ph.D. thesis. Stanford University (1971)
15. Tenenbaum, J.: Rules and similarity in concept learning. Advances in Neural Information Processing Systems 12 (2000)
16. Theocharous, G., Beckwith, R., Butko, N., Philipose, M.: Tractable POMDP planning algorithms for optimal teaching in “SPAIS”. In: IJCAI PAIR Workshop (2009)
17. Villano, M.: Probabilistic student models: Bayesian belief networks and knowledge space theory. In: Proceedings of the Second International Conference on Intelligent Tutoring Systems (1992)

Metacognitive Practice Makes Perfect: Improving Students' Self-Assessment Skills with an Intelligent Tutoring System

Ido Roll[1], Vincent Aleven[2], Bruce M. McLaren[2], and Kenneth R. Koedinger[2]

[1] University of British Columbia, 6224 Agricultural Road, Vancouver, BC V6T 1Z1, Canada
[2] Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA
ido@phas.ubc.ca, {aleven,bmclaren,koedinger}@cs.cmu.edu

Abstract. Helping students' improve their metacognitive and self-regulation skills holds the potential to improve students' ability to learn independently. Yet, to date, there are relatively few success stories of helping students enhance their metacognitive skills using interactive learning environments. In this paper we describe the Self-Assessment Tutor, an intelligent tutoring system for improving the accuracy of the judgments students make regarding their own knowledge. A classroom evaluation of the Self-Assessment Tutor with 84 students found that students improved their ability to identify their strengths while working with the Self-Assessment Tutor. In addition, students transferred the improved self-assessment skills to corresponding sections in the Geometry Cognitive Tutor. However, students often failed to identify their knowledge deficits a-priori and failed to update their assessments following unsuccessful solution attempts. This study contributes to theories of Self-Assessment and provides support for the viability of improving metacognitive skills using intelligent tutoring systems.

Keywords: Metacognition, self-regulated learning, intelligent tutoring systems, Self-assessment, help seeking, feeling of knowing (FOK).

1 Introduction

Students who apply productive metacognitive and self-regulation skills show better learning when working with interactive learning environments [1]. Therefore, many tutoring systems support various self-regulation skills [2,3,4]. Yet, only few systems attempt to improve students' self-regulation skills in a manner that persists even after support is removed and transfers to new learning situations. Two success stories are Betty's Brain, a learning-by-teaching environment for scientific concepts [5], and the Help Tutor, an add-on tutoring agent that gives metacognitive feedback on students' help-seeking behaviors while learning Geometry [6]. In both cases, students who received metacognitive prompts [5] or feedback [6] improved corresponding aspects of their learning trajectories in unsupported transfer tasks within the same environments.

In the current paper we describe the Self-Assessment (SA) Tutor, an intelligent tutoring system for improving students' SA skills. The term SA refers to students'

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 288–295, 2011.

tendency and ability to accurately evaluate their knowledge while learning [7, 8]. Accurate SA was shown to correlate with productive help-seeking behaviors [9]. A small number of systems provide support for SA in order to help students choose appropriate cognitive strategies [10] and monitor their progress [11]. In order for students' SAs to be accurate, students should be aware of the relative strengths and weaknesses of their knowledge, in relation to a target task [12]. However, students often over-estimate their ability [7]. Students who lack sufficient domain knowledge are especially likely to make inaccurate SAs, probably because they cannot distinguish between correct and incorrect answers, even when the solutions are presented to them [12]. In fact, students often base their assessments on familiarity with the problems, not with the answers [13].

The current study further evaluates the relationship between domain knowledge and accuracy of SAs, and focuses on acquisition, calibration, and transfer of SA skills in the context of an interactive learning environment. Specifically, we address the following questions:

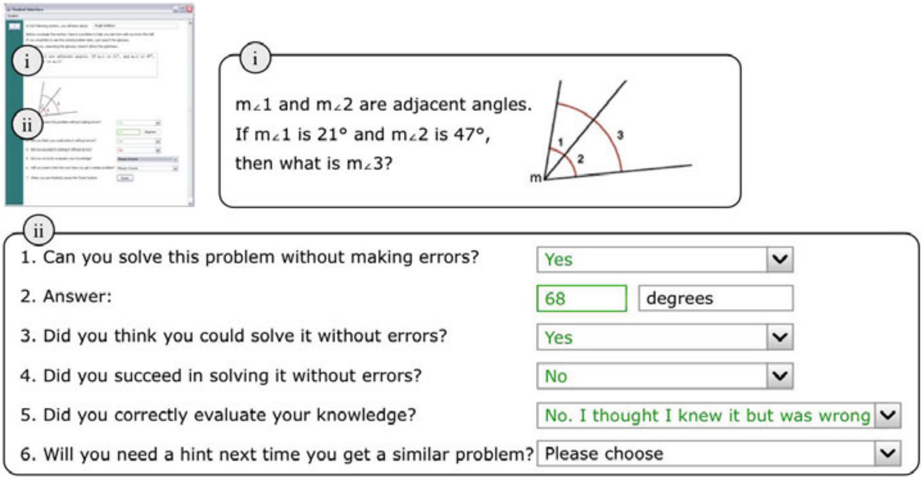
1 Do students who lack domain knowledge also make less accurate SAs?
2 How do students use their actual problem-solving ability to calibrate their SAs?
3 Does the SA Tutor help students improve the accuracy of their SAs?
4 Do improved SA skills transfer to unsupported sections of the problem-solving environment?

In what follows we describe the SA Tutor and its classroom evaluation.

2 The Self-Assessment Tutor

The goals of the SA Tutor are to help students get in the habit of assessing their ability, improve the accuracy of their SAs, and use their SAs to inform strategy choice. The SA Tutor, an intelligent tutoring system [14], adheres to several principles of metacognitive tutoring [15]. The SA Tutor is a *learning by doing* environment in that students learn to self assess by practicing SA in the context of math problem solving. The SA Tutor helps students set the following subgoals: predict one's own ability, attempt to solve the problem, reflect on the experience, and plan future interaction. [15]. Since students who identify their own errors learn better than students who receive feedback on their errors [16], the SA tutor helps students to identify their SA errors. Adaptive feedback is given to students who fail to attend mismatches between their SAs and their actual performance. Last, the SA Tutor supports the entire problem-solving process, starting before students attempt to solve the target problem, and ending after students reflect on the solution.

Students begin the SA process by predicting whether they could solve a given target problem without making errors (Question 1 in Figure 1). Students reply by choosing either ”yes” or “no, I need a hint,” in which case a relevant hint is displayed. Both replies are legitimate, and no feedback is given on students' initial SA. Students are then asked to solve the target problem (Question 2). On this step, typical support is available (correctness feedback, error messages, and on-demands hints). Question 3 asks students to recall their initial SA and Question 4 asks students to reflect on whether they solved the target problem without making errors. Feedback on questions 3 and 4 is given to insure accurate recollection of students' initial SA and actual



m∠1 and m∠2 are adjacent angles.
If m∠1 is 21° and m∠2 is 47°,
then what is m∠3?

1. Can you solve this problem without making errors?	Yes
2. Answer:	68 degrees
3. Did you think you could solve it without errors?	Yes
4. Did you succeed in solving it without errors?	No
5. Did you correctly evaluate your knowledge?	No. I thought I knew it but was wrong
6. Will you need a hint next time you get a similar problem?	Please choose

Fig. 1. The SA Tutor (top left corner) includes two components: (i) domain-level problems, and (ii) self-assessment scaffold

ability. Question 5 is key in getting students to compare their initial SA to their actual ability. In response to the question "did you correctly evaluate your knowledge?", students can choose "yes", "no--I thought I knew it but was wrong", or "no--I knew more than I predicted". Feedback on this question is contingent on students' initial SA and actual ability. For example, a student who estimated she could solve the target problem, yet failed to do so without errors, is expected to choose "no—I thought I knew it but was wrong". Last, students predict the need for help on new, similar, problems, by choosing either "yes, I will need the advice", or "no, I think I got it" (Question 6). No feedback is given on this question. The SA Tutor is an example-tracing tutor and was built using the Cognitive Tutor Authoring Tools [17].

In our study, the SA Tutor was used in conjunction with the Geometry Cognitive Tutor. Each section of the SA Tutor includes 3-5 problems, each of which targets a specific skill that is practiced in the subsequent section of the Geometry Cognitive Tutor. Students first evaluate their ability on the target set of problems in the SA Tutor. Students then complete a sequence of problems that require the same skills, using the Geometry Cognitive Tutor.

3 Methods

The SA Tutor was evaluated in a classroom study together with the Help Tutor [6]. An analysis of students' help-seeking behaviors is presented elsewhere [6,18].

Participants: The study took place in a rural vocational high school with 84 students in five classrooms, taught by two teachers. All students, 10th and 11th graders, were enrolled in the Cognitive Tutor Geometry class, and thus were familiar with the Cognitive Tutor and its interface. Because the experimental conditions differed substantially, whole classes were assigned to conditions, balancing, across conditions, the number and level of students. 46 students in three classes were assigned to the SA

Condition, while 38 students in the remaining two classes were assigned to the Control Condition.

Materials: Students in both conditions worked on two units from the Geometry Cognitive Tutor: Angles (Unit 1) and Quadrilaterals (Unit 2). Each of the units had a single warm-up problem, followed by 3 sections. Each section focused on a different set of skills within the general topic of the unit. Students in the Control condition worked with the unmodified Geometry Cognitive Tutor, which did not include the SA Tutor or the Help Tutor. Students in the SA Condition alternated between the SA Tutor and the Geometry Cognitive Tutor augmented with the Help Tutor.

Procedure: The study spanned 3 months. During Month 1 all students worked on Unit 1 in their respective conditions. During Month 2 the study was put on hold while students prepared for statewide exams using the unmodified Geometry Cognitive Tutor. During Month 3 students worked on Unit 2, again according to the conditions to which they had been assigned. All students were assigned to the beginning of each unit at the same start date. Progress within each unit was at an individual pace. Figure 2 illustrates the structure of the study.

Month:	Month 1, Unit 1 (Angles)							Month 2, Various CT units	Month 3, Unit 2 (Quadrilaterals)						
Section:	Warm -up	Section 1.1		Section 1.2		Section 1.3			Warm -up	Section 2.1		Section 2.2		Section 2.3	
SA:	CT	SA	CT	SA	CT	SA	CT		CT	SA	CT	SA	CT	SA	CT
Control:	CT	CT		CT		CT			CT	CT		CT		CT	

Fig. 2. The procedure of the study. SA and CT denote SA Tutor and Cognitive Tutor respectively.

Analysis: Unless stated otherwise, all analysis involves students in the SA Condition only. Question numbers refer to the questions in the SA Tutor as shown in Figure 1.

4 Results

On average, students worked with the SA Tutor for 18 minutes. As it turns out, many students took longer than expected to complete sections 1.2 and 2.1 in the Geometry Cognitive Tutor, and thus did not reach the more advanced sections. In Unit 1, all 46 students worked on Section 1.1, 37 students worked on Section 1.2, and only 14 students reached Section 1.3. In Unit 2, 44 students worked on Section 2.1, and only 12 and 2 students reached Sections 2.2 and 2.3 respectively.

Research Question 1: Effect of Domain Knowledge. The SA Tutor asks students to predict their ability to solve a target item (Question 1), and following their prediction, to solve it (Question 2). Overall, students assessed their ability correctly on 77% of all problems. The accuracy of students' assessments depends on their knowledge level. There is a high correlation between having the relevant domain knowledge (as assessed by averaging performance on Question 2 on all items within each section) and making accurate SA on the same set of items; $r(160) = .52$, $p < .0005$.

The relationship between having relevant domain-level knowledge and accuracy of SA is most apparent when looking at the single item level (Table 1). Students who had sufficient knowledge to solve the target item predicted their success (prior to attempting) on 84% of the items, while students who lacked sufficient knowledge to solve the target item predicted their failure (prior to attempting) only on 37% of the items. Thus, over-estimation was much more common than under-estimation.

Table 1. Initial SA vs. competence (number of items and row-based percentage)

		Students' initial SA (Question 1)		
		Already know	Need help	Overall
Students' ability to solve the target item (Question 2)	High	455 (84%) ✓ True positive	85 (16%) ✗ False negative (under-estimation)	540 (100%)
	Low	64 (63%) ✗ False positive (over-estimation)	37 (37%) ✓ True negative	101 (100%)

Research Question 2: Calibration of SA. The SA Tutor asks students to report their SA twice for each skill: once before solving the target item (Question 1) and once after solving it (Question 6). Therefore, students can use their performance on the target item (Question 2) to calibrate their SA.

A repeated measures ANOVA (with initial- and updated-SA as a time series, and actual performance as a treatment) found that updated SA (Question 6) depends on the interaction between initial-SA (question 1) and actual performance (Question 2), $F(1{,}638) = 36, p < .0005$. As Table 2 shows, students' updated SA (Question 6) relies heavily on their initial SA (Question 1), but was fine-tuned based on their actual performance (Question 2). The significant interaction shows that students who under-estimated their ability updated their SAs more often than students who over-estimated their ability. In fact, 77% of the students who thought they already knew how to solve the item did not update their SAs following their failure to solve the item. The high persistence of over-estimation is especially noteworthy, given that a single failure is sufficient to suggest that the student does not possess sufficient knowledge.

Table 2. Updated SA: Students' reported confidence in their ability to solve additional problems that require the same skills without additional assistance (Question 6)

Initial SA (Q1)	Actual Performance (Q2)	Updated SA (Q6)
Already know	Got it right	88% will not need additional help
	Got it wrong	77% will not need additional help
Need Help	Got it right	51% will not need additional help
	Got it wrong	27% will not need additional help

Research Question 3: Metacognitive Improvement. Due to the high attrition, and to avoid a selection bias (in that data in the advanced sections pertains to better students), we evaluate the improvement in students' SA only on sections in which attrition was low: Unit 1 Sections 1.1 and 1.2, and Unit 2 Section 2.1. Overall, students became more accurate in their initial self-assessments, as evaluated by comparing their SAs on Question 1 to their actual performance on Question 2: Section 1.1: 71%; Section 1.2: 74%; section 2.1: 79%. However, a likely explanation is that students' SAs improved since their domain-knowledge increased. To control for the effect of domain learning, we analyzed the accuracy of students' SAs separately for items on which students had sufficient knowledge and items for which students lacked sufficient knowledge (as evaluated by performance on Question 2 in each problem). An ANOVA of accuracy vs. section, using data from items that students solved without errors (high competence items), found that students improved their SA significantly from Section 1.1 (77%) to Section 1.2 (88%): $F(1,294) = 4.8$, $p < .03$. There was also a positive trend from Section 1.1 to 2.1 (83%) on high-competence items ($p = .13$). However, there was no improvement in the accuracy of students' SAs on items that they subsequently failed to solve correctly (Section 1.1: 37%; Section 1.2: 40%; Section 2.1: 39%). These results suggest that students got significantly better at identifying their strengths, but not their weaknesses.

Research Question 4: Transfer of SA skills. To evaluate whether students transferred their improved ability to self-assess to an unsupported learning environment, we compare students' SAs in the SA Tutor to their actual help-seeking behavior in the Geometry Cognitive Tutor. Specifically, we examine the rate of asking for help in the Cognitive Tutor prior to attempting new problem-steps. One expects to see that students seek more help in the Geometry Cognitive Tutor on skills for which they report to have low SA in the SA Tutor. It is only natural that students ask for more help on skills they do not know. However, as shown earlier, students are relatively poor at identifying their limitations.

The correlation between skills on which students sought more help in the Cognitive Tutor and skills on which students reported to have low initial-SA in the SA Tutor is high and significant, $r\ (7) = .75$, $p = .02$. Other factors such as inherent difficulty or generic SA skills may affect students' help-seeking behaviors within the Cognitive Tutor. These factors can be accounted for by partialling-out the corresponding help frequencies on the same skills of students in the Control Condition, who were susceptible to the same factors, yet did not work with the SA Tutor. The partial-correlation between help-requests in the Cognitive Tutor and reported need for help in the SA Tutor, controlling for help-requests in the Cognitive Tutor by students in the Control Condition, remains high and significant: $\textit{partial-}r(6) = .73$, $p = .04$. This suggests that training within the SA Tutor, rather than item difficulty or generic self-assessment skills, accounts for the high correlation between students' SA and help-seeking behavior.

5 Discussion and Summary

We have described the SA Tutor, an intelligent tutoring system for SA. The SA Tutor scaffolds the SA process in four steps: predicting ability to solve a target problem; attempting to solve that problem; reflecting on the SA by comparing the initial SA to the actual performance; and updating the SA for future interaction.

A classroom evaluation of the SA Tutor found that the SA Tutor helped students improve several aspects of their SA behavior. The SA Tutor helped students improve the accuracy of their initial SA with practice, and students also calibrated their SAs based on their actual performance on the target set of problems. Last, analysis of students' help-seeking behaviors in the Geometry Cognitive Tutor indicates that students transferred their improved SA knowledge to the subsequent sections in the Geometry Cognitive Tutor.

The *in-vivo* study emphasizes the large dependency of SA on domain knowledge, as was previously found in the lab [12]. Students' SAs were accurate on 84% of the items that they subsequently solved correctly, compared with a mere 37% of the items that they subsequently failed to solve. The rate of over-confidence did not decline with practice, while the rate of under-confidence declined significantly. The phenomenon of over-confidence seems not only common, but also persistent. Students were the least likely to use evidence to calibrate their assessment on items on which they were over-confident, even in the presence of feedback (see Table 2). Apparently, students did not attribute their failure to solve the problem to lack of relevant knowledge. As stated by Kruger and Dunning, "those with limited knowledge in a domain suffer a dual burden: Not only do they reach mistaken conclusions and make regrettable errors, but their incompetence robs them of the ability to realize it" [12]. We have previously reported on students' underuse of help in the Cognitive Tutor environment [6]. The current result suggests that the underuse of help may be a result of students' over-confidence in their ability.

The main limitation of this study is its scope. Not enough data is available on students' SA patterns with more extensive practice, as well as students' spontaneous SA behavior and their domain-level learning gains in the Cognitive Tutor environment.

This work makes several contributions. First, we describe a unique system for tutoring SA skills. The system uses established principles of metacognitive tutoring [15] to help students learn how to evaluate their ability. Second, we demonstrate how classroom research using interactive learning environments can be used to inform our understanding of metacognitive processes. For example, log-file analysis was used to decompose factors that affect students' SA, and to better understand the relationship between domain knowledge and accuracy of SA. Most importantly, this work demonstrates the ability of intelligent tutoring systems to help students improve their metacognitive skills in a manner that transfers to unsupported tasks within the tutoring system.

Acknowledgements. This work was supported by the Pittsburgh Science of Learning Center, which is supported by the National Science Foundation (#SBE-0354420), and by the National Science Foundation (#IIS-0308200). The contents of the paper are solely the responsibility of the authors and do not necessarily represent the official views of the NSF.

References

1. Azevedo, R., Moos, D., Greene, J., Winters, F., Cromley, J.: Why is externally-facilitated regulated learning more effective than self-regulated learning with hypermedia? Ed. Tech. Res. And Dev. 56(1), 45–72 (2008)
2. Aleven, V., Koedinger, K.R.: An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. Cog. Sci. 26(2), 147–179 (2002)

3. Bunt, A., Conati, C., Muldner, K.: Scaffolding self-explanation to improve learning in exploratory learning environments. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 656–667. Springer, Heidelberg (2004)
4. Roll, I., Aleven, V., Koedinger, K.R.: The Invention Lab: Using a Hybrid of Model Tracing and Constraint-Based Modeling to Offer Intelligent Support in Inquiry Environments. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 115–124. Springer, Heidelberg (2010)
5. Biswas, G., Roscoe, R., Jeong, H., Sucler, B.: Promoting self-regulated learning skills in agent-based learning environments. In: Proceedings of the 17th International Conference on Computers in Education (2009) (CDROM)
6. Roll, I., Aleven, V., McLaren, B.M., Koedinger, K.R.: Improving Students' Help-Seeking Skills Using Metacognitive Feedback In An Intelligent Tutoring System. Learning and Instruction 21, 267–280 (2011)
7. Dunning, D., Heath, C., Suls, J.M.: Flawed Self Assessment. Psychological Science In The Public Interest 5(3), 69–106 (2004)
8. El Saadawi, G.M., Azevedo, R., Castine, M., Payne, V., Medvedeva, O., Tseytlin, E., Legowski, E., Jukic, D., Crowley, R.S.: Factors affecting feeling-of-knowing in a medical intelligent tutoring system: the role of immediate feedback as a metacognitive scaffold. Adv. Health. Sci. Educ. Theory. Pract. (2009)
9. Nelson-Le Gall, S., Kratzer, L., Jones, E., DeCooke, P.: Children's self-assessment of performance and task-related help seeking. J. Exp. Child. Psychol. 49, 245–263 (1990)
10. Gama, C.: Metacognition in interactive learning environments: The reflection assistant model. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 668–677. Springer, Heidelberg (2004)
11. Bull, s., Kay, j.: Student Models that Invite the Learner. The SMILI Open Learner Modelling Framework. Int. J. of Art. Int. in Ed. 17(2), 89–120 (2007)
12. Kruger, J., Dunning, D.: Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. J. of Personality and Soc. Psychol. 77(6), 1121–1134 (1999)
13. Reder, L.M., Ritter, F.E.: What Determines Initial Feeling of Knowing? Familiarity With Question Terms. J. Exp. Psychol. Learn. Mem. Cogn. (3), 435–451 (1992)
14. VanLehn, K.: The behavior of tutoring systems. Int. J of AI in Ed. 16(3), 227–265
15. Roll, I., Aleven, V., McLaren, B.M., Koedinger, K.R.: Designing for metacognition - applying cognitive tutor principles to the tutoring of help seeking. Metacognition and Learning 2(2), 125–140 (2007)
16. Mathan, S.A., Koedinger, K.R.: Fostering the Intelligent Novice: Learning from errors with metacognitive tutoring. Ed. Psychol. 40(4), 257–265 (2005)
17. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A new paradigm for intelligent tutoring systems: Example-tracing tutors. Int. J. of AI In ED. 19(2), 105–154 (2008)
18. Roll, I., Aleven, V., McLaren, B.M., Koedinger, K.R.: Can help seeking be tutored? Searching for the secret sauce of metacognitive tutoring. In: Luckin, R., Koedinger, K.R., Greer, J. (eds.) Proceedings of the International Conference on Artificial Intelligence in Education, pp. 203–210. IOS Press, Amsterdam (2007)

Self-assessment of Motivation: Explicit and Implicit Indicators in L2 Vocabulary Learning

Kevin Dela Rosa and Maxine Eskenazi

Language Technologies Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, Pennsylvannia, USA
{kdelaros,max}@cs.cmu.edu

Abstract. Self-assessment motivation questionnaires have been used in classrooms yet many researchers find only a weak correlation between answers to these questions and learning. In this paper we postulate that more direct questions may measure motivation better, and they may also be better correlated with learning. In an eight week study with ESL students learning vocabulary in the REAP reading tutor, we administered two types of self-assessment questions and recorded indirect measures of motivation to see which factors correlated well with learning. Our results showed that some user actions, such as dictionary look up frequency and number of times a word is listened to, correlate well with self-assesment motivation questions as well as with how well a student performs on the task. We also found that using more direct self-assesment questions, as opposed to general ones, was more effective in predicting how well a student is learning.

Keywords: Motivation Modelling, Intelligent Tutoring Systems, Computer Assisted Language Learning, Motivation Diagnosis, English as a Second Language.

1 Introduction

Motivation modelling and its relation to user behavior has receieved attention by the educational computing community in recent years. William and Burden define motivation as "*a state of cognitive and emotional arousal which leads to a conscious decision to act, and which gives rise to a period of sustained intellectual and/or physical effort in order to attain a previously set goal (or goals)*" [1]. The use of self-assessment questionaires is one common approach to measuring motivation. One such construct is Motivated Strategies for Learning Questionnaire (MSLQ), an 81-item survey designed to measure college students' motivational orientations and their use of various learning strategies [2]. While questionnaires are useful to detect enduring motivational traits, some are criticized, particularly those administered prior to interaction. Since a student's motivation is likely to change during an interaction, it is important to use them with other methods to adapt instruction and to gather more transient information about a student's motivation [3]. Other methods of assessing motivation include direct communication with students, emotion detection, and recorded interactions with an intelligent tutor. For modelling and understanding user behavior automatically, Baker [4] showed that machine learning models trained on

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 296–303, 2011.

log data of student activity can be used to automatically detect when a student is off-task. A study by Cetintas et al. [5] reached a similar conclusion, using a regression model personalized to each student. And, Baker et al. [6] showed that a latent response model can be used to determine if a student is "gaming" the system in a way that leads to poor learning.

An important issue has been how to automatically detect a student's current motivational state. As mentioned above, one method of measuring motivation is questionnaires that cover a variety of motivation aspects. One important consideration is how detailed and/or direct these survey questions should be with respect to the task or, in other words, is it better to have questions that are tightly focused on the tasks being performed by the student or is it better to construct questions that are more general and can cover many difference aspects of motivation, such as the MLSQ. Also, a student's usage of a tutoring system, as indicated by the amount of activity and types of actions taken, may furnish good implicit indicators of a student's motivation.

We propose that in a computer-assisted L2 language learning environment certain recorded student interactions during learning activities can act as implicit indicators of that student's motivation. We also propose that these implicit indicators, as well as explicit ones like self-assessment surveys, can be used to predict the amout of learning that is taking place. Lastly we postulate that more *direct* questions may measure motivation better and may also be better correlated with learning.

For this study we used a web-based language tutor called REAP [7]. REAP, which stands for **REA**der-specific **P**ractice, is a reading and vocabulary tutor targeted at ESL students, developed at Carnegie Mellon University, which uses documents harvested from the internet for vocabulary learning. REAP's interface has several features that help to enhance student learning. One key feature in REAP is that it provides users with the ability to listen to the spoken version of any word that appears in a reading, making use of Cepstral Text-to-Speech[1] to synthesize words on demand when they are clicked on. Additionally, students look up the definition of any of the words, during readings, using a built-in electronic dictionary. REAP also automatically highlights focus words, the words targeted for vocabulary acquisition in a particular reading. REAP is a language tutor and a testing platform for cognitive science studies [8, 9], as is the case of this study.

In this paper we describe a classroom study that compares the effectiveness of different motivational indicators in a vocabulary learning environment. We define the different types of survey questions we used as explicit measures of motivation and the various user actions we recorded as indirect indicators of motivation. Next we describe the results of a classroom study that integrated our various motivation indicators and how well they correlated with our learning measures. Finally we discuss the implications of our results and suggest future directions.

2 Classroom Study

In order to determine which of our hypothesized indicators of motivation were most related to learning we conducted a classroom study with a web-based tutor, focused

[1] Cepstral Text-to-Speech. `http://www.cepstral.com`

on L2 English vocabulary learning, and recorded responses to motivation questionnaires and user actions that we log, which may indirectly indicate a student's motivation level. The classroom study consisted of a pre-test and post-test with multiple choice fill-in-the-blank vocabulary questions, and six weekly readings, each followed by practice vocabulary questions similar to, but not the same as those in the pre-test and post-test. During the pre-test and post-test, a set of seventeen self-assessment motivation questions were administered, and after each weekly session there were a set of five motivation questions.

21 intermediate-level ESL college students at the University of Pittsburgh's English Language Institute participated in the study and completed all of the activities. For this study the readings and vocabulary questions had 18 focus words, taken from either the General Service List[2] or the Academic Word List[3], and not part of the class' core vocabulary list.

In the following subsections we describe the types of questionnaires administered, the recorded user actions, and the metrics we used to measure student learning.

2.1 Motivation Questionnaire

We administered motivation survey questions as explicit measures of motivation after each reading. 17 survey questions were administered in the pre-test and post-test, as shown in Table 1, using a five-point *Likert* scale, with a response of 5 indicating the greatest agreement with the statement and 1 indicating the least agreement. The 17 questions were divided into two groups: *General* and *Direct*.

We call *General* survey questions high-level survey questions which have been used in past REAP studies because of their generality; they are used in many studies in the Pittsburgh Science of Learning Center[4]. For example, one of the *General* questions we used was, "*When work was hard I either gave up or studied only the easy parts*", which can be used for many different subject matters. The design of these questions was guided by the MLSQ [2], and aimed to use the fewest number of questions possible that cover the most motivational constructs.

We call *Direct* questions the more explicit items that focused on aspects directly related to the reading activities accomplished over the course of the study. An example of a *Direct* question: "*Learning vocabulary in real documents is a worthwhile activity*". This is focused on the specific REAP tasks.

A total of twelve *General* and five *Direct* self-assessment motivation questions were administered during the pre-test and post-tests. Additionally, the five *Direct* motivation survey questions in Table 2 were asked after each weekly reading activity, at regular intervals in between the pre-test and post-test, to see how the responses correlated with student behavior and learning at each reading. We wanted to determine if there was a difference in how well each of these two question groups correlates to the learning measures we recorded (multiple choice questions). We hypothesize that questions more directly related to the tasks/activities performed will

[2] The General Service List. `http://jbauman.com/gsl.html`

[3] The Academic Word List. `http://www.victoria.ac.nz/lals/resources/academicwordlist`

[4] Pittsburgh Science of Learning Center (PSLC). `http://www.learnlab.org`

be better at predicting motivation and learning. This is guided by unpublished results of past REAP studies which have shown that higher-level questions generally failed to correlate well with learning measures, and previous success with direct questions by Heilman et al. [10].

Table 1. Pre-test/Post-test Motivation Survey Questions

ID	Survey Question Prompt	Group	Type
S1	I am sure I understood the ideas in the computer lab sessions.	General	E
S2	I am sure I did an excellent job on the tasks assigned for the computer lab sessions.	General	E
S3	I prefer work that is challenging so I can learn new things.	General	A
S4	I think I will be able to use what I learned in the computer lab sessions in my other classes.	General	V
S5	I think that what I learned in the computer lab sessions is useful for me to know.	General	V
S6	I asked myself questions to make sure I knew the material I had been studying.	General	O
S7	When work was hard I either gave up or studied only the easy parts.	General	A
S8	I find that when the teacher was talking I thought of other things and didn't really listen to what was being said.	General	A
S9	When I was reading a passage, I stopped once in a while and went over what I had read so far.	General	O
S10	I checked that my answers made sense before I said I was done.	General	O
S11	I did the computer lab activities carefully.	General	E
S12	I found the computer lab activities difficult.	General	A
S13	I continued working on the computer lab activities outside the sessions.	Direct	A
S14	I did put a lot of effort into computer lab activities.	Direct	A
S15	I did well on the computer lab activities.	Direct	E
S16	I preferred readings where I could listen to the words in the document.	Direct	V
S17	Learning vocabulary in real documents is a worthwhile activity.	Direct	V

Table 2. Post-reading Survey Motivation Questions

ID	Survey Question Prompt	Type
Q1	Did you find the spoken versions of the word helpful while reading this document?	V
Q2	Do you find it easy to learn words when you read them in documents?	E
Q3	Did you find this document interesting?	V
Q4	Did you learn something from this document?	V
Q5	Does reading this document make you want to read more documents?	A

Furthermore, for this study we grouped the questions into three types:

- **Affective** (*A*): Deal with *emotional reactions* to a task
- **Expectancy** (*E*): Deal with beliefs about a student's *ability to perform* a task
- **Value** (*V*): Deal with goals and beliefs about the *importance and interest* of a task

The 3 groups are based on the Pintrich and De Groot components of motivation (self-efficacy, intrinsic value, test anxiety) [11]. We used this grouping to simplify the analysis of the results. Note that in the tables and figures, "**Other** (*O*)" signifies a question that failed to group into one of the three types, typically a question on learning strategies.

2.2 Recorded User Interactions

In addition to survey questions, we recorded actions taken by the students which we hypothesize would indirectly correspond to motivation and might also correlate with learning. The following were recorded during each activity:

Word lookup activity, using our built-in electronic dictionary
A1: Total number of dictionary lookups
A2: Number of focus words looked up in the dictionary
A3: Number of dictionary lookups involving focus words
Words listening activity, using our built-in speech synthesis
A4: Mean number of listens per word
A5: Total number of listens
A6: Number of words listened to
Average time spent on activity tasks
A7: Time spent reading the documents
A8: Time spent on practice questions

2.3 Learning Measures

In order to assess how well students learned the target vocabulary words, we recorded the following measures:

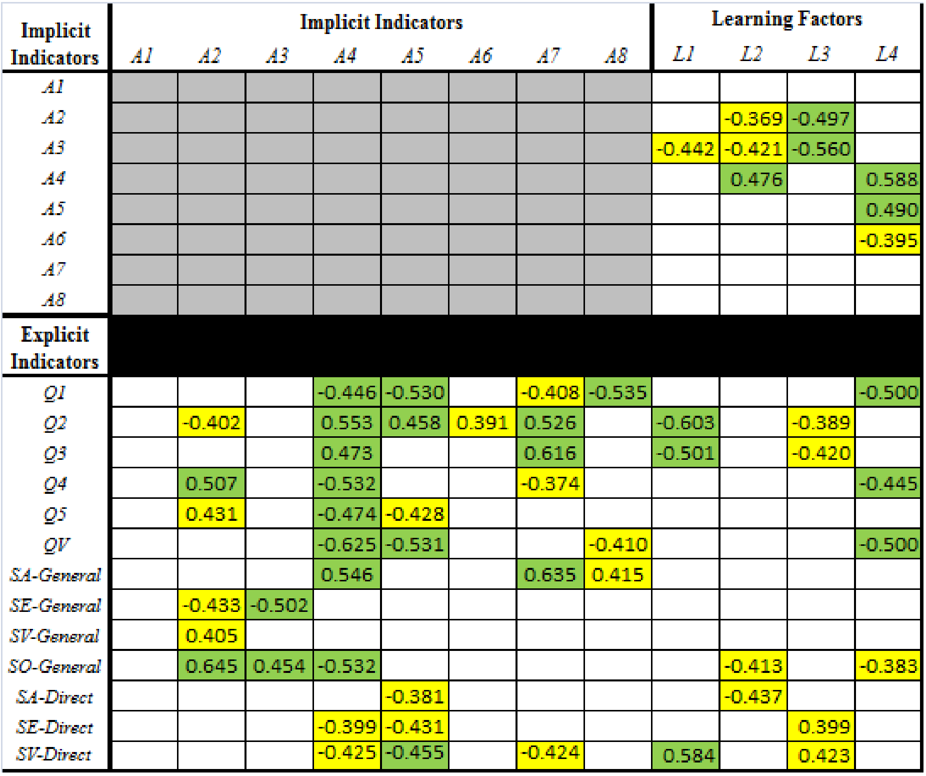
L1: Average post-reading practice question accuracy (for all questions appearing directly after reading the documents)
L2: Pre-test to post-test normalized gain
L3: Post-test accuracy
L4: Average difference between pre-test and post-test scores

Note that that L2 and L4 are two different ways of looking at the improvements made by students over the course of the study, where L2 is tuned to the relative difference between the test scores, and L4 is sensitive to the absolute difference in scores.

3 Results

The results of our study show that the use of the REAP system significantly helped students improve their performance on the vocabulary tests, as evident in the average overall gains between the pre-test and post-test {*L3*} ($p < 0.004$), whose average scores were 0.3439 (± 0.0365) and 0.5000 (± 0.0426) respectively. The average post-reading practice question accuracy was 0.8417 {*L1*} (± 0.0466). The overall average normalized gain {*L2*} between pre-test and post-test was 0.2564 (± 0.0466), and average difference in score {*L4*} between the pre-test and post-test was 0.1561 (± 0.0232).

In order to find the motivational factors that best correlated with learning, we computed a Pearson correlation coefficient matrix for the values for the different factors and grouped question responses, and determined the significance of each pair of correlations using a two-tailed test. Figure 1 summarizes correlation and signifcance values found between the motivational and learning factors, and shows how the indirect indicators correlate with the expicit indicators of motivation.



Implicit Indicators	Implicit Indicators								Learning Factors			
	A1	*A2*	*A3*	*A4*	*A5*	*A6*	*A7*	*A8*	*L1*	*L2*	*L3*	*L4*
A1												
A2										-0.369	-0.497	
A3									-0.442	-0.421	-0.560	
A4										0.476		0.588
A5												0.490
A6												-0.395
A7												
A8												
Explicit Indicators												
Q1				-0.446	-0.530		-0.408	-0.535				-0.500
Q2		-0.402		0.553	0.458	0.391	0.526		-0.603		-0.389	
Q3				0.473			0.616		-0.501		-0.420	
Q4		0.507		-0.532			-0.374					-0.445
Q5		0.431		-0.474	-0.428							
QV				-0.625	-0.531			-0.410				-0.500
SA-General				0.546			0.635	0.415				
SE-General		-0.433	-0.502									
SV-General		0.405										
SO-General		0.645	0.454	-0.532						-0.413		-0.383
SA-Direct					-0.381					-0.437		
SE-Direct				-0.399	-0.431						0.399	
SV-Direct				-0.425	-0.455		-0.424		0.584		0.423	

Fig. 1. Significant correlations values between motivational & learning factors, and between implicit & explicit motivation indicators. Color signifies level of significance, with green representing strong statistical significance ($p < 0.05$), and yellow representing moderate signifance ($p < 0.1$). Note that self-correlations and correlations with low significance values were omitted. Also note that Q1-Q5 correspond to the students' average survey response values for all reading activities, and with respect to the implict indicators, A1-A8 correspond to the students' average values of those indicator values over all reading activties.

Additionally, when we looked at the post-reading accuracies of each individual reading activity (as oppose to the overall averages, which were shown in Figure 1), the following motivational factors tended to significantly correlate with the post-reading practice question accuracies at significance levels varying levels over the six readings between $p < 0.01$ and $p < 0.05$:

- Q2 response: Do you find it easy to learn words when you read them in documents?
- Q3 response: Did you find this document interesting?
- Total number of dictionary lookups
- Number of focus words looked up in the dictionary
- Time spent on practice questions

4 Discussion

We see in Figure 1 that most of the *General* questions did not significantly correlate with the various learning measures, while the *Direct* questions did. In fact, the only sub-group of *General* questions that correlated with the learning measures was *Other*, those questions that did not fit well into our three types (*Affective*, *Expectancy*, and *Value*), which is not surprising since the *Other* questions mainly focused on learning strategies as opposed to motivation. Furthermore, all of the direct questions asked after each reading, except for *Q5* (*Does reading this document make you want to read more documents?*), had significant correlations with the learning measures. Perhaps the reason *Q5* failed to have significant correlations is due to the fact students were not given the option to actually act on the desire to read more documents in our tutor during the semester, due to class constraints. Therefore, our results imply that *General* questions are less effective at predicting the learning outcome of a student than are the *Direct* motivation questions which are more closely focused on the tasks performed. Moreover, our results hew closely to past results by Bandura [12] in the domain of self-efficacy.

Additionally, the implicit motivation indicators, based on recorded student actions, seemed to correlate well with our learning measures, particularly our word listening and dictionary lookup-related interactions, which implies that these kinds of actions can also help in predicting a student's motivational state. Interestingly, the amount of time spent on reading and answering questions did not correlate well with the learning factors, which implies that simply using the absolute amount of time spent on task may not be a good factor to use in predicting a student's learning outcomes, and perhaps taking into account how the time was used by students would be a better factor to consider. Furthermore, most of the implicit indicators we recorded had significant correlations with one or more of the direct motivation survey questions in Figure 1 that we asked after each reading, which implies that implicit indicators may be effective in predicting the motivational state of the student using an intelligent tutor on an activity-by-activity basis.

5 Conclusion

Understanding and modeling motivation and student behavior is an important issue for intelligent tutoring systems. We proposed that some student interactions recorded by tutors can act as implicit indicators of motivation, and that these implicit indicators, as well as explicit ones like self-assessment motivation questions, can be used to predict student learning. We tested our hypothesis with a classroom study using a vocabular tutor that integrates implicit and explicit indicators. The results show that some user actions, such as dictionary look ups and listening to words, correlate well with motivation questions and student performance. We also found that the use of *Direct* questions, specifically tailored to the tasks, rather than *General* and all-encompassing questions, was more effective in predicting student performance.

Acknowledgements. This project is supported by the Cognitive Factors Thrust of the Pittsburgh Science of Learning Center which is funded by the US National Science Foundation under grant number SBE-0836012. 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. The authors would like to thank Betsy Davis, Tamar Bernfeld, and Chris Ortiz at the University Pittsburgh's ELI for their participation and input in the studies.

References

1. Williams, M., Burden, R.L.: Psychology for language teachers: A social constructivist approach. Cambridge University Press, Cambridge (1997)
2. Pintrich, P.R., Smith, D.A.R., Garcia, T., McKeachie, W.: A manual for the use of the motivated strategies for learning questionnaire (MSLQ). Report, Ann Arbor (1991)
3. de Vicente, A., Pain, H.: Motivation diagnosis in intelligent tutoring systems. In: Goettl, B.P., Halff, H.M., Redfield, C.L., Shute, V.J. (eds.) ITS 1998. LNCS, vol. 1452, pp. 86–95. Springer, Heidelberg (1998)
4. Baker, R.S.: Modeling and understanding students' off-task behavior in intelligent tutoring systems. In: 25th ACM SIGCHI Conference on Human Factors in Computing Systems, pp. 1059–1068. ACM, New York (2007)
5. Cetintas, S., Si, L., Xin, Y.P., Hord, C.: Automatic Detection of Off-Task Behaviors in Intelligent Tutoring Systems with Machine Learning Techniques. IEE Trans. Learn. Tech. 3, 228–236 (2010)
6. Baker, R.S., Corbett, A.T., Koedinger, K.R.: Detecting Student Misuse of Intelligent Tutoring Systems. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 54–76. Springer, Heidelberg (2004)
7. Heilman, M., Collins-Thompson, K., Callan, J., Eskenazi, M.: Classroom success of an Intelligent Tutoring System for lexical practice and reading comprehension. In: 9th International Conference on Spoken Language (2006)
8. Dela Rosa, K., Parent, G., Eskenazi, M.: Multimodal learning of words: A study on the use of speech synthesis to reinforce written text in L2 language learning. In: 2010 ISCA Workshop on Speech and Language Technology in Education (2010)
9. Kulkarni, A., Heilman, M., Eskenazi, M., Callan, J.: Word sense disambiguation for vocabulary learning. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 500–509. Springer, Heidelberg (2008)
10. Heilman, M., Juffs, A., Eskenazi, M.: Choosing Reading Passages for Vocabulary Learning by Topic to Increase Intrinsic Motivation. In: 13th International Conference on Artificial Intelligence in Education (2007)
11. Pintrich, P.R., De Groot, E.V.: Motivational and Self-Regulated Learning Components of Classroom Academic Performance. J. of Ed. Psych. 82, 33–40 (1990)
12. Bandura, A.: Self-Efficacy. In: Ramachaudran, V.S. (ed.) Encyclopedia of Human Behavior, vol. 4, pp. 71–81. Academic Press, New York (1994)

Detecting Carelessness through Contextual Estimation of Slip Probabilities among Students Using an Intelligent Tutor for Mathematics

Maria Ofelia Clarissa Z. San Pedro[1], Ryan S.J.d. Baker[2],
and Ma. Mercedes T. Rodrigo[1]

[1] Ateneo de Manila University, Loyola Heights, Quezon City, Philippines
[2] Worcester Polytechnic Institute, Worcester, MA
sweetsp@gmail.com, rsbaker@wpi.edu, mrodrigo@ateneo.edu

Abstract. A student is said to have committed a careless error when a student's answer is wrong despite the fact that he or she knows the answer (Clements, 1982). In this paper, educational data mining techniques are used to analyze log files produced by a cognitive tutor for Scatterplots to derive a model and detector for carelessness. Bayesian Knowledge Tracing and its variant, the Contextual-Slip-and-Guess Estimation, are used to model and predict carelessness behavior in the Scatterplot Tutor. The study examines as well the robustness of this detector to a major difference in the tutor's interface, namely the presence or absence of an embodied conversational agent, as well as robustness to data from a different school setting (USA versus Philippines).

Keywords: Carelessness, Slip, Contextual-Slip-and-Guess, Bayesian Knowledge Tracing, Cognitive Tutors, Scatterplot.

1 Introduction

Recently, there has been increasing attention to studying disengaged behaviors within intelligent tutoring systems [2, 6]. One student behavior that has been less thoroughly explored is carelessness [8, 9, 10] – a label ascribed to the unconscientious performance of actions that were not originally intended by the individual, usually leading to errors [13, 15]. This can happen when an individual is in a hurry or overconfident in carrying out a task, when doing routine activities, or when doing tasks perceived to be of minor importance [12]. Carelessness is not an uncommon behavior in students [8], even among high-performing students [9]. Modeling this student behavior may lead not only to a fuller understanding of a student's true learning capabilities, but also to improved teaching strategies and educational materials.

Recent studies have shown educational software to be useful in measuring student affect, knowledge, and disengaged behavior within a classroom setting. One type of educational software, an Intelligent Tutoring System (ITS), provides students with guided learning support as they engage in problem-solving [16]. Researchers have used ITSs in modeling student learning, approximating the knowledge state of each

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 304–311, 2011.

student at a given time [11]. In recent years, further studies using ITSs have branched out towards modeling and detecting student affective states [1, 17] and behaviors associated with affect and poorer learning, including gaming the system [6] and off-task behavior [2]. Of importance to the analyses in this paper, Baker, et al. [4] have recently developed a slip detector [4] which can be used to detect carelessness as student behavior within ITSs. This operationalization of carelessness accords to the definition of carelessness in Clements, that errors committed by students deemed competent in problem-solving indicate carelessness behavior [9]. However, although the model has been applied within multiple tutors, it is not yet clear how widely the model generalizes. For this model to be broadly useful, it must be able to generalize to new tutor designs and student populations.

Within this paper, we establish the generalizability of models of students' carelessness, using two versions of a Cognitive Tutor for Scatterplot generation and interpretation, differing in the presence or absence of an Embodied Conversational Agent (ECA) [6]. We analyze interaction logs from Philippine high school students under these two conditions, producing two slip detectors based on previous work at modeling this construct [3, 4]. We then test the detectors on the other version of the learning environment's dataset to see how well the detectors generalize to data sets with significant differences in design. We also test the detectors on interaction logs from US middle school students using the same tutors to see how well these models generalize to data with a different school setting. In the long term, the work hopes to contribute to a generalizable model of carelessness.

2 Carelessness Detection in Cognitive Tutors

Cognitive Tutors employ a strategy known as Knowledge Tracing to estimate a student's latent knowledge based on his/her observable performance. This process is based on Corbett and Anderson's Bayesian Knowledge Tracing (BKT) model [11].

The BKT framework, in its original articulation, enables the Cognitive Tutor to infer student knowledge by continually updating the estimated probability a student knows a skill every time the student gives a first response to a problem step regardless whether the response is correct or not. It uses four parameters – two learning parameters L_O (initial probability of knowing each skill) and T (probability of learning the skill at each opportunity to make use of a skill), together with two performance parameters G (probability that the student will give a correct answer despite not knowing a skill) and S (probability that the student will give an incorrect answer despite knowing the skill) – for each skill (estimated from data information in each skill). These parameters are invariant across the entire context of using the tutor. Using Bayesian analysis, BKT re-calculates the probability that the student knew the skill before the response (at time n-1), using the information from the response, then accounts for the possibility that the student learned the skill during the problem step, such that [11]:

$$P(L_n \mid Action_n) = P(L_{n-1} \mid Action_n) + ((1 - P(L_{n-1} \mid Action_n)) * P(T)) \ . \qquad (1)$$

Studies by Baker et al. proposed a variant of the BKT model which contextually estimates the Guess and Slip parameters, with this Contextual Slip being an indicator of carelessness [3, 4]. The Contextual Guess-and-Slip (CGS) model examines the properties of each student response as it occurs, in order to assess the probability that the response to an action is a guess or slip. In this model, the estimates of the slip and guess probabilities are now dynamic and depends on the contextual information of the action, such as speed and history of help-seeking from the tutor. It has been shown that this model can indicate aspects of student learning that are not captured by traditional BKT, which may significantly improve prediction of post-test performance [5]. Based on prior theory on carelessness (as discussed above), we use the slip model as an operationalization of carelessness [cf. 8] (though slips may also occur for other reasons, such as shallow knowledge [e.g. 5]).

3 Methods

Data were gathered from 126 students from a large public high school in Quezon City, Philippines (PH). For 80 minutes, students used a Cognitive Tutor unit on scatterplot generation and interpretation [6]. Students had not explicitly covered these topics in class prior to the study. Prior to using the software, students viewed conceptual instruction. Each student in each class took a nearly isomorphic pre-test and post-test, counterbalanced across conditions.

Within the Scatterplot Tutor, the learner is given a problem scenario. He/she is also provided with data that he/she needs to plot in order to arrive at the solution. He/she is asked to identify the variables that each axis will represent. He/she must then provide an appropriate scale for each axis. He/she has to label the values of each variable along the axis and plot each of the points of the data set. Finally, he/she interprets the resultant graphs. The Scatterplot tutor provides contextual hints to guide the learner, feedback on correctness, and messages for errors. The skills of the learner is monitored and displayed through skill bars that depict his/her mastery of skills.

Sixty four of the participants (Scooter group) were randomly assigned to use a version of the tutor with an embodied conversational agent, “Scooter the Tutor”. Scooter was designed to both reduce the incentive to prevent gaming the system and to help students learn the material that they were avoiding by gaming, while affecting non-gaming students as minimally as possible. Gaming the system is defined in [6] as behavior aimed at obtaining correct answers and advancing within the tutoring curriculum by systematically taking advantage of regularities in the software’s feedback and help. Scooter displays happiness and gives positive message when students do not game (regardless of the correctness of their answers), but shows dissatisfaction when students game, and provides supplementary exercises to help them learn material bypassed by gaming. The remaining 62 participants (NoScooter group) used a version of the Scatterplot Tutor without the conversational agent. As such, skills associated with the tutor version with Scooter have additional Scooter-related skills not present in the tutor without Scooter. The number of students assigned to the conditions in this study was unbalanced because of data gathering schedule disruptions caused by inclement weather.

Log files generated by the Cognitive tutor recorded the students’ actions in real-time. A set of 26 transaction features identical to the set used in [4] was extracted and derived from the logs for each problem step. These features were used since they have been shown to be effective in creating detectors of other constructs [e.g. 6]. Baseline BKT parameters were fit with brute-force search [cf. 5]. From this baseline model, estimates of whether the student knew the skill at each step were derived and used to label actions (whether correct or incorrect response) with the probability that the actions involved guessing or slipping, based on the student performance on successive opportunities to apply the rule [4]. As in [3, 4], Bayesian equations were utilized in computing training labels for the Slip (and Guess) probabilities for each student action (A) at time N, using future information (two actions afterwards – N+1, N+2), in order to infer as accurately as possible the true probability that a student's action at time N was due to knowing the skill, or due to a slip or guess [4]. Using Eq. 2, the probability that the student knew the skill at time N can be calculated, given information about the actions at time N+1 and N+2 ($A_{N+1,N+2}$).

$$P(A_N \text{ is a Slip} \mid A_N \text{ is incorrect}) = P(L_n \mid A_{N+1,N+2}) \, . \qquad (2)$$

Models for Contextual Slip (and Guess) were then produced through Linear Regression using truncated training data [3], to create models that could predict contextual guess and slip without using data from the future. These new models were then substituted for the Guess and Slip parameters per problem step, labeling each action with variant estimates as to how likely the response is a guess or a slip. With dynamic values of Guess/Slip, the learning parameters Lo and T were re-fit per skill.

4 Results and Discussion

Using student-level cross-validation (6-fold) Linear Regression Modeling in RapidMiner, a Carelessness model approximating the Contextual Slip Model was created with the 26 attributes extracted, plus the label of the probability that the action step is a Slip. Table 1 shows a model trained on data that used the tutor without an agent (NoScooter group) and a model trained on data that used a tutor with an agent (Scooter group), with their respective final attributes. The detector from the NoScooter group data achieved a correlation coefficient of r = 0.460 to the labels, while the detector from the Scooter group data achieved r = 0.481, in each case a moderate degree of correlation [19].

The carelessness detectors passed the tests for model degeneracy in [3, 4]. Within the 127 students' activities, there were a total of 1221 scenarios where the student had three consecutive correct actions per skill, while 419 instances where the student had at least 10 consecutive correct actions. In both cases, the model was not empirically degenerate – the estimate of knowing the skill afterwards did not decrease after these correct actions. The generated carelessness model also passed the theoretical degeneracy test – the maximum of the new contextual P(S) values did not exceed 0.5.

This model was successful at predicting whether the student would perform correctly on the next opportunity to practice the skill, in both the NoScooter and Scooter groups. The contextual-guess-and-slip model achieved prediction of A' = 0.821 for the NoScooter group, and A' = 0.814 for the Scooter group (A' refers to the model's ability to distinguish between a right and wrong answer, with a chance probability of 0.5). Both contextual-guess-and-slip models achieved slightly higher A' values than their baseline BKT counterpart (A' = 0.816 for the NoScooter group, and

Table 1. Carelessness (Contextual Slip) Models for NoScooter and Scooter Groups

Carelessness (NoScooter) =	Carelessness (Scooter) =
-0. 07256 * Answer is right	-0. 11895 * Answer is right
-0. 03658 * Action is a bug	-0. 02501 * Action is a bug
+0.08997 * Action is a help request	+0. 05535* Input is a choice
+0.09944 * Input is a choice	-0. 02876 * Input is a number
-0. 03595 * Input is a string	-0. 03772 * Input is a point
-0. 02018 * Input is a number	-0.03632 * Input is checkbox or not choice/string/number/point
-0. 02805 * Input is a point	+0.04486 * Probability that the student knew the skill involved in this action
-0.01662 * Input is checkbox or not choice/string/number/point	+ 0.07296 * Pknow-direct from log files
+0. 00903 * Probability that the student knew the skill involved in this action	+ 0.10466 * Not first attempt at skill in this problem
+ 0.00707 * Pknow-direct from log files	+0.00434 * Time taken, normalized in terms of SD off average across all students at this step
- 0.01495 * Not first attempt at skill in this problem	+0.00249 * Time taken in last three actions, normalized
-0.06562 * First transaction on new problem	+0.11895 * Answer not right
-0.00573 * Time taken, normalized in terms of SD off average across all students at this step	-0.00099 * Errors has this student averaged on this skill across problems
+0.07257 * Answer not right	-0.00033 * Total time spent on this skill across problems
+0.00025 * Number of errors the student made on this skill on all problems	+ 0.02207 * Previous 3 actions were on the same cell
-0.00067 * Errors has this student averaged on this skill across problems	-0.01615 * Previous 5 actions were on the same cell
+0.00021 * Total time spent on this skill across problems	-0.01205 * How many of the previous 5 actions were errors
+0.00532 * Previous 3 actions were on the same cell	-0.02557 * Has the student made at least 3 errors on this problem step, in this problem
-0.00335 * Previous 5 actions were on the same cell	+0.06601
+0.00766 * How many of the previous 8 actions were help requests	
-0.00792 * How many of the previous 5 actions were errors	
-0.03136 * Has the student made at least 3 errors on this problem step, in this problem	
+0.08456	

A' = 0.807 for the Scooter group), although this was not cross-validated. It is worth-noting that with the low number of skills within the Scatterplot Tutor, the potential benefits of the CGS model for reducing over-parameterization are reduced.

In addition to A' values, the goodness of the models were also supported by their Bayesian Information Criterion values for Linear Regression Models [18]. Both models had BIC' values far less than -6 (NoScooter = -414.60, Scooter = -401.21), the cut-off for a model being better than chance [18], making these models better-than-chance indicators of this behavior.

To investigate generalizability, we tested each detector on the opposite data set, i.e. the NoScooter detector was used on the Scooter group dataset and the detector from the Scooter group was used on the NoScooter group dataset. We also tested the detectors with Scatterplot log data from a US school setting [cf. 6]. These interaction logs from the US (described in greater detail in [6]) were gathered from 6^{th}-8^{th} grade students, in the suburbs of a medium-sized city in the Northeastern USA. Fifty-two students used the Scooter version of the tutor, and 65 students used the NoScooter version. Table 2 shows the detectors' correlation between the labeled (from Eq. 2 – our CGS equations) and predicted (from our models) slip values in each data set. Within the NoScooter condition data, the detector trained on the Scooter condition data actually performed slightly better (r=0.471) than the detector trained on the NoScooter data (r=0.460). Within the Scooter data, the detector trained on the NoScooter data performed moderately worse (r=0.392) than the detector trained on the Scooter data (r=0.481), although still respectably. These results appear to indicate between mild degradation and no degradation when a carelessness detector is transferred between versions of the tutor with or without an ECA. The asymmetry in transfer between the two environments can be attributed to the fact that the skills and action steps in the NoScooter environment are also present in the Scooter environment, whereas the opposite is not true. When transferred to data from the USA, both of the detectors trained on data from the Philippines performed quite well, performing better in the USA than in the Philippines for all combinations of training and test conditions. This is striking evidence for detector generalizability, when the detectors perform better in a new country than in the original country, with no re-fitting. As a whole, taking correlation as a metric, the carelessness detectors trained in this study appear to show little to no degradation when transferred to different data sets.

Table 2. Correlation (r value) of Slip Detectors to Slip Labels in Different Data Sets

	NoScooter-Group Detector (PH)	Scooter-Group Detector (PH)
NoScooter Group Data (PH)	0.460	0.471
Scooter Group Data (PH)	0.392	0.481
NoScooter Group Data (US)	0.490	0.591
Scooter Group Data (US)	0.537	0.605

An interesting additional finding was that the Scooter group committed fewer errors compared to the NoScooter group (both PH and US data). Whether or not these errors were careless, it is possible that Scooter’s interventions supported future student performance in the tutor

For both test environments, we also examined the values of P(S) according to the model, when certain conditions hold in the data (average predicted P(S) = 0.12 and maximum P(S) = 0.38 across all conditions). One finding is that errors were more likely to be slips when the probability that the student knew the skill before answering was greater than the initial probability L_O for that skill (the 4009 cases in the data where this condition held had an average predicted P(S) = 0.18, compared to the average P(S) = 0.10 where this condition didn't hold). In addition, if a student’s successive actions (at least two) for a particular problem step and skill are correct, a subsequent mistake was more likely to be a slip (850 cases where predicted P(S) increased to an average of 0.20). Slip was even more strongly associated with cases where the student has made very few prior errors on a skill with a high initial knowledge value (L_O) (355 cases in the data, average predicted P(S) = 0.27).

5 Conclusion

In this paper, we developed detectors of student carelessness within a lesson on scatterplots in a Cognitive Tutor for middle school mathematics, building off prior work in this area [3, 4]. These detectors were tested for robustness when transferred to a different version of the same tutor, and data from schools in a different country. Two carelessness detectors (for the NoScooter condition and the Scooter condition, which incorporated an Embodied Conversational Agent) were created from interaction logs acquired from the tutor usage of Philippine high school students, using a variant of Bayesian Knowledge Tracing, the Contextual Guess and Slip method, which dynamically estimated if an incorrect response was a slip. Our results suggest that these detectors are generalizable and can transfer across tutors with interface differences (i.e. with and without an embodied conversational agent), as well as across different school settings (i.e. Philippine high school and US middle school), increasing potential for automatically intervening in future systems when the students is careless.

Acknowledgments

This research was supported by the Philippines Department of Science and Technology Philippine Council for Advanced Science and Technology Research and Development under the project “Development of Affect-Sensitive Interfaces”, and by the Pittsburgh Science of Learning Center (National Science Foundation) via grant “Toward a Decade of PSLC Research”, award number SBE-0836012. We thank Mrs. Carmela Oracion, Jenilyn Agapito, Ivan Jacob Pesigan, Ma. Concepcion Repalam, Salvador Reyes, Ramon Rodriguez, the Ateneo Center for Educational Development, the Department of Information Systems and Computer Science of the Ateneo de Manila University and the faculty, staff, and students of Ramon Magsaysay Cubao High School for their support in this project.

References

[1] Arroyo, I., Cooper, D., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion Sensors Go To School. In: Proceedings of the International Conference on Artificial Intelligence in Education, pp. 17–24 (2009)

[2] Baker, R.S.J.d.: Modeling and Understanding Students’ Off-Task Behavior in Intelligent Tutoring Systems. In: Proceedings of ACM CHI 2007: Computer-Human Interaction, pp. 1059–1068 (2007)

[3] Baker, R.S.J.d., Corbett, A.T., Aleven, V.: Improving Contextual Models of Guessing and Slipping with a Truncated Training Set. In: Proceedings of the 1st International Conference on Educational Data Mining, pp. 67–76 (2008)

[4] Baker, R.S.J.d., Corbett, A.T., Aleven, V.: More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 406–415. Springer, Heidelberg (2008)

[5] Baker, R.S.J.d., Corbett, A.T., Gowda, S.M., Wagner, A.Z., MacLaren, B.M., Kauffman, L.R., Mitchell, A.P., Giguere, S.: Contextual Slip and Prediction of Student Performance After Use of an Intelligent Tutor. In: Proceedings of the 18th Annual Conference on User Modeling, Adaptation, and Personalization (2008)

[6] Baker, R.S.J.d., Corbett, A.T., Koedinger, K.R., Evenson, S.E., Roll, I., Wagner, A.Z., Naim, M., Raspat, J., Baker, D.J., Beck, J.: 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)

[7] Baker, R.S.J.d., D'Mello, S.K., Rodrigo, M.M.T., Graesser, A.C.: Better to Be Frustrated than Bored: The Incidence, Persistence, and Impact of Learners' Cognitive-Affective States during Interactions with Three Different Computer-Based Learning Environments. International Journal of Human-Computer Studies 68(4), 223–241 (2010)

[8] Baker, R.S.J.d., Gowda, S.M.: An Analysis of the Differences in the Frequency of Students' Disengagement in Urban, Rural, and Suburban High Schools. In: Proceedings of the 3rd International Conference on Educational Data Mining, pp. 11–20 (2010)

[9] Clements, M.A.: Analysing Children's Errors on Written Mathematical Tasks. Educational Studies in Mathematics, 1–21 (1982)

[10] Clements, M.A.: Careless Errors Made by Sixth-grade Children on Written Mathematical Tasks. Journal for Research in Mathematics Education 13(2), 136–144 (1982)

[11] Corbett, A.T., Anderson, J.R.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. User Modeling and User-Adapted Interaction 4, 253–278 (1995)

[12] Craighead, W.E.: The Concise Corsini Encyclopedia of Psychology and Behavioral Science, 3rd edn. (2004)

[13] Dix, A., Finlay, J., Abowd, G., Beale, R.: Human-Computer Interaction. Prentice Hall, Englewood Cliffs (1993)

[14] Fogarty, J., Baker, R.S.J.d., Hudson, S.E.: Case Studies in the Use of ROC Curve Analysis for Sensor-based Estimates in Human Computer Interaction. In: Proceedings of Graphics Interface, pp. 129–136 (2005)

[15] Levitin, D.J. (ed.): Foundations of Cognitive Psychology: Core Readings. MIT Press, Cambridge (2002)

[16] Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.: Intelligent Tutoring Goes to School in the Big City. International Journal of Artificial Intelligence in Education 8, 30–43 (1997)

[17] McQuiggan, S.W., Lee, S., Lester, J.C.: Early Prediction of Student Frustration. In: Paiva, A., Prada, R., Picard, R.W. (eds.) Affective Computing and Intelligent Interaction, pp. 698–709 (2007)

[18] Rafferty, A.E.: Bayesian Model Selection in Social Research. Sociological Methodology 25, 111–163 (2003)

[19] Rosenthal, R., Rosnow, R.L.: Essentials of Behavioural Research: Methods and Data Analysis. McGraw-Hill Humanities, New York (2008)

A "Laboratory of Knowledge-Making" for Personal Inquiry Learning

Mike Sharples[1], Trevor Collins[2], Markus Feißt[1], Mark Gaved[2], Paul Mulholland[2], Mark Paxton[1], and Michael Wright[3]

[1] University of Nottingham, Jubilee Campus, Wollaton Road, Nottingham, NG8 1BB, UK
{mike.sharples,markus.feisst,mark.paxton}@nottingham.ac.uk
[2] The Open University, Walton Hall, Milton Keynes, MK7 6AA, UK
{t.d.collins,m.b.gaved,p.mulholland}@open.ac.uk
[3] University of Bath, Claverton Down, Bath, BA2 7AY
michaelwright1981@googlemail.com

Abstract. We describe nQuire, a constraint-based learning toolkit to support a continuity of inquiry based learning between classroom and non-formal settings. The paper proposes design requirements for personal inquiry learning environments that support learning of personally meaningful science topics with development of metacognitive understanding and self-regulation of the scientific process through situated practice. It introduces a generic implementable model of the inquiry process, and describes an instantiation in the nQuire learning environment. An example of the use of the toolkit for a Healthy Eating inquiry with 28 Year 9 students concludes with results of the trial, design issues and recommendations.

Keywords: inquiry learning, science learning, metacognition, constraint-based.

1 Introduction

In a complex world where scientific knowledge is publically contested, it is essential that children should develop the skills necessary to understand and engage in the science that influences their lives. From early 20th century onwards [1] there have been proposals that children should learn science through collaborative inquiry. Since scientific thinking is essentially social, John Dewey proposed that schools should become "laboratories of knowledge-making" [1] p. 127, where children engage in experimentation, communication, and self-criticism. One hundred years later, the value of learning through shared knowledge making is yet more apparent [2], but most schools are no nearer to being places for children to engage in cooperative inquiry. Rather than hoping to re-fashion schools, we now have the opportunity to design computer-based laboratories that can enable shared knowledge-making within and beyond the classroom [3] [4] [5] [6].

The nQuire learning environment has been developed as part of a three-year project between the University of Nottingham and the Open University UK to support children aged 11-14 in coming to understand themselves and their world through a

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 312–319, 2011.

process of personal inquiry learning. With the aid of software running on both mobile and desktop computers children have been able to investigate issues that affect their lives, across different settings – including the classroom, their homes, and discovery centres – through a scientific process of gathering and assessing evidence, conducting experiments and engaging in informed debate. The software guides the inquiry process, providing an interactive visual representation of scientific practice. Other papers have covered the conduct and outcomes of the trials [7] [8]. Here, we describe the design principles and implementation of the nQuire toolkit for personal inquiry learning, exemplified by an investigation into ‘healthy eating’.

2 Requirements

The requirements for nQuire were developed through process of co-design of technology and pedagogy. We refined and revised an initial set of design requirements, derived from a survey of published literature on inquiry-based science learning, through a series of seven school-based trials in partnership with teachers and their students, and science learning advisors, on topics of: urban heat islands (twice in successive years), heart rate and fitness, microclimates, healthy eating, sustainability, and effect of noise pollution on birds. The later topics were developed from ideas proposed by the school students as being relevant to their personal lives and interests.

The general requirements are:

- *Relevance*: The inquiry topics should have personal meaning and relevance to the learners, without being too personal or embarrassing.
- *Accessibility*: The technology should be available and accessible to each child throughout the investigation.
- *Continuity*: The inquiry should start in the classroom, with a teacher supporting the children to gain a shared understanding of the science inquiry process and agree the aims of specific investigation, then continue in a setting that allows experiment, discussion and collection of rich authentic data, and conclude back in the classroom for sharing, discussion and presentation of results.
- *Coordination*: The toolkit should enable students to carry out group or whole class inquiries, with data and interim findings being shared amongst the group.
- *Visualisation*: The learners should be able to examine the emerging findings in relation to the goals or hypotheses of the inquiry, in a form that makes the data visual and appealing.
- *Metacognition*: Learners should be able to reflect on their progress by reference to a representation of the science inquiry process, and to explore the consequences of future actions through previews and ‘what if’ explorations.
- *Flexibility*: There should be support for a range of inquiry formats, including fair test experiments, quasi-experimental designs, surveys, exploratory investigations, and debate with peers and experts.
- *Bricolage*: The technology should offer a choice of tools and a place to engage in playful exploration of data.

These requirements have informed the design of the toolkit. What is novel about nQuire is that: a) it guides the learner by an explicit visual implementation of the entire inquiry cycle; b) it supports a continuity of learning on handheld and desktop devices between classrooms and non-formal settings; c) it employs a constraint-based representation of the data flow between learning activities, so that rather than prescribing a fixed sequence of tasks it gives the learner flexibility to explore the consequences of actions; d) the data flow as well as the choice and sequence of activities can be authored by non-technical users; e) the system can run either in a client browser from a web server, or as a stand-alone system on Windows, Apple, or Linux devices with data being synchronised across members of a workgroup.

3 An Implementable Representation of the Inquiry Process

Previous work has represented the inquiry process as a list, cycle or spiral, e.g. [9] [10]. Drawing on this work, we have developed a generic representation of the inquiry learning process (Fig. 1) that can serve as an aid to metacognitive learning and also be implemented as a computer-mediated activity and data structure to orchestrate the computer-based activities. Thus, the 'octagon' representation is shown on the home page of the nQuire toolkit and was also copied as a wall chart by the teacher to provide a classroom overview of the inquiry process.

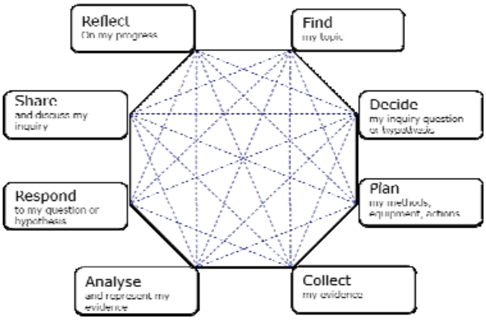

Reflect
On my progress
Find
my topic
Share
and discuss my inquiry
Decide
my inquiry question or hypothesis
Respond
to my question or hypothesis
Plan
my methods, equipment, actions
Analyse
and represent my evidence
Collect
my evidence

Fig. 1. An implementable representation of the inquiry learning process

The inquiry process is depicted as a cycle of phased activities, where an investigation can begin at any phase (for example, it could start with analysing data collected by another group and use that to frame a new inquiry question) and each phase builds on knowledge gained from previous activities. The hatched lines indicate possible dependencies so, for example, the Respond phase revisits the inquiry question in the knowledge of collected data, or the Reflect phase could result in a change of Plan for a new cycle of inquiry. Figure 2 shows possible data dependencies between the Decide and Respond phases of the inquiry, with the evidence to support answers to the learners' key questions depending on their selection of appropriate measures and accurate collection and presentation of data.

Implemented in the nQuire toolkit, these dependencies determine relations between activities and their associated data, providing the user with a constraint-based 'activity guide'. Fig. 3 shows a typical screen from nQuire, with the phases of the inquiry process shown as a navigation panel (1). The 'octagon' representation of the inquiry process is also available as a link from each screen, as a home page and visual reminder of the inquiry process (2). To the user, the navigation panel functions as a dynamic To Do list. It provides an ordering of phases, with each phase associated with one or more activities specific to the inquiry (such as 'view, add or edit my data'). The current activity (in this example, to view current data) is displayed in the main area of the screen (3).

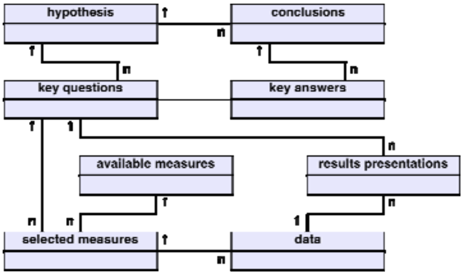
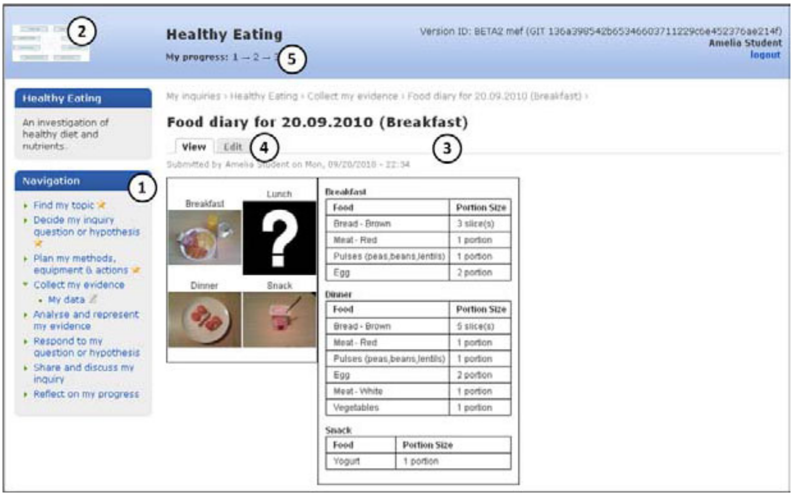


Fig. 2. Data dependencies for the inquiry elements


2
Healthy Eating
My progress: 1 → 2 → 3
5
Version ID: BETA2 mef (GIT 136a398542b65346603711229c6e452376ae214f)
Amelia Student
logout
Healthy Eating
An investigation of healthy diet and nutrients.
Navigation
1
Find my topic
Decide my inquiry question or hypothesis
Plan my methods, equipment & actions
Collect my evidence
My data
Analyse and represent my evidence
Respond to my question or hypothesis
Share and discuss my inquiry
Reflect on my progress
My inquiries › Healthy Eating › Collect my evidence › Food diary for 20.09.2010 (breakfast) ›
View
Edit
4
3
Breakfast
Lunch
?
Dinner
Snack

Food diary for 20.09.2010 (Breakfast)

Submitted by Amelia Student on Mon, 09/20/2010 - 22:34

Breakfast

Food	Portion Size
Bread - Brown	3 slice(s)
Meat - Red	1 portion
Pulses (peas,beans,lentils)	1 portion
Egg	2 portion

Dinner

Food	Portion Size
Bread - Brown	5 slice(s)
Meat - Red	1 portion
Pulses (peas,beans,lentils)	1 portion
Egg	2 portion
Meat - White	1 portion
Vegetables	1 portion

Snack

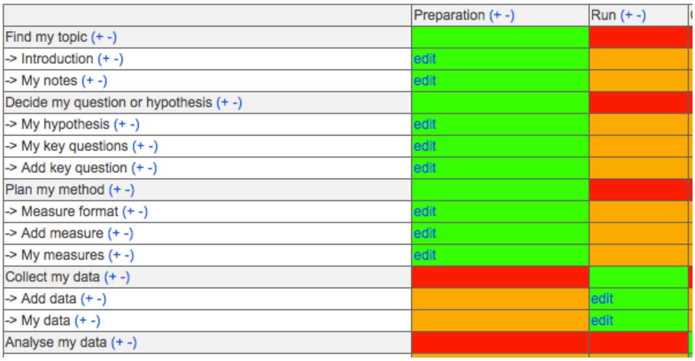
Food	Portion Size
Yogurt	1 portion

Fig. 3. nQuire screenshot showing an activity from the Healthy Eating investigation

The user can move between viewing and editing an activity (4), and so can preview an activity, such as viewing data collected from a previous investigation, before adding and editing new material. The inquiry process can be organised into temporal stages (for example, to correspond to a sequence of school lessons or project assignments) (5). In the authoring module of nQuire, the teacher or instructional designer can assign which phases of the inquiry process are active for each stage (Fig. 4). Each activity can be given a status of 'unavailable', 'view', 'start', or 'edit'.

Fig. 4 shows the first Preparation stage, with the first three inquiry phases being editable and the remaining phases being available to view only. The editable phases associated with the current stage are indicated by stars beside the phases on the user's navigation panel (Fig. 3, (1)). So, in the screen on Fig. 3, the user is currently at stage 1 of the inquiry, but is looking ahead to preview some example data, as an ‘advance organiser’ [11] to understand and explore the entire sequence of activities. Further authoring tools enable the teacher or designer to associate specific activities with phases and to allocate students to groups that can share and edit each other’s data.

Dependencies between the activities create productive constraints on the inquiry process. For example, for some investigations, the students must choose measures (such as ‘location’, ‘temperature’, ‘humidity’) and determine which of these are key measures and which are dependent ones. These choices will constrain how the results are collected and presented. However, at any time the user can change a selection and these changes will be propagated through the system, so the user can explore ‘what if’ possibilities without being committed to the outcomes.



	Preparation (+ -)	Run (+ -)
Find my topic (+ -)	🟩	🟥
-> Introduction (+ -)	🟩 edit	🟧
-> My notes (+ -)	🟩 edit	🟧
Decide my question or hypothesis (+ -)	🟩	🟥
-> My hypothesis (+ -)	🟩 edit	🟧
-> My key questions (+ -)	🟩 edit	🟧
-> Add key question (+ -)	🟩 edit	🟧
Plan my method (+ -)	🟩	🟥
-> Measure format (+ -)	🟩 edit	🟧
-> Add measure (+ -)	🟩 edit	🟧
-> My measures (+ -)	🟩 edit	🟧
Collect my data (+ -)	🟥	🟩
-> Add data (+ -)	🟧	🟩 edit
-> My data (+ -)	🟧	🟩 edit
Analyse my data (+ -)	🟥	🟥

Fig. 4. Authoring component to associate inquiry phases and activities with stages

3.1 Implementation

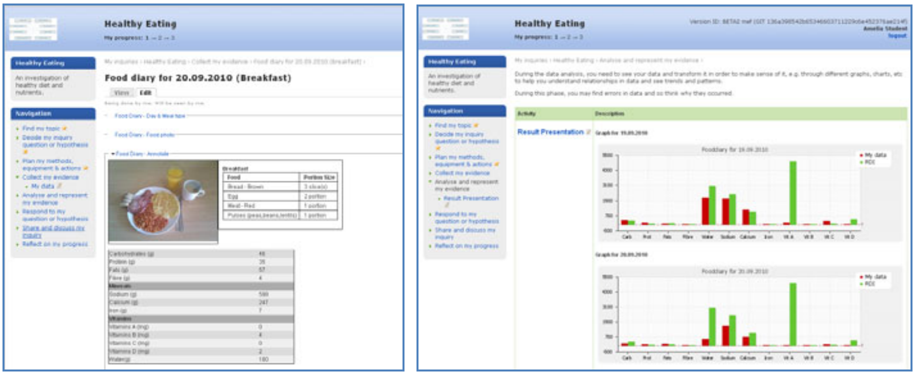
The nQuire system is implemented in the PHP-based Drupal open source content management system. Drupal modules provide support for handling web forms, content presentation, managing users and groups and storing and presenting media. A series of additional nQuire modules support the authoring and navigation of inquiry phases, stages and activities. The activity modules support specific inquiry activities such as forming and revisiting the inquiry questions, data collection, analysis, and uploading presentations. Existing Drupal modules, such as a voting module, can be added as nQuire activities. A further set of utility modules offer additional functionality such as import, export and synchronisation of inquiries. The nQuire system can run on a remote server, accessed through a standard web browser, or can be downloaded to run locally under Windows, Mac, or Linux operating systems, including a complete installation that runs from a USB Flash Drive so no software need be installed on the computer. The system has been tested on netbook computers and on the Apple iPhone. It is available for access or download at www.nquire.org.uk.

4 Healthy Eating Example

As a worked example, we describe the activities associated with the Healthy Eating investigation. A full analysis of the study and its results is presented elsewhere [12]; here we show how the investigation was supported by the nQuire toolkit. The design of the study was as follows.

The participants consisted of 28 students from Year 9 (aged 14) of an inner-city school in Nottingham, UK. A full class (14 girls and 14 boys) was recruited to the study. They followed an inquiry investigation on the topic of Healthy Eating involving nine science lessons plus out of school activities over a three week period. The children engaged in two types of inquiry: each child complied a photo diary of their personal eating habits and then worked with other children in a group to explore the relation between their daily diet and the Recommended Daily Intake (RDI) for children of their age; as a class, the children proposed and sent questions by email to an expert in nutrition. A second class of children (16 girls and 13 boys) followed normal school lessons, but took the same tests at equivalent points in the school year.

Each child in the intervention group was loaned an Asus Eee netbook computer running the nQuire toolkit and a camera for the duration of the trial. The nQuire software was pre-loaded onto the computer and the system booted to a login page for the inquiry. An Apache server running locally on each computer allowed it to be used in stand-alone mode, without connection to the web. Within the school, the data for each group could be synchronised using the school’s wireless network.



Healthy Eating

My progress: 1 → 2 → 3

Healthy Eating

An investigation of healthy diet and nutrients.

Navigation

- Find my topic
- Decide my inquiry question or hypothesis
- Plan my methods, equipment & actions
- Collect my evidence
  - My data
- Analyse and represent my evidence
- Respond to my question or hypothesis
- Share and discuss my inquiry
- Reflect on my progress

My inquiries › Healthy Eating › Collect my evidence › Food diary for 20.09.2010 (Breakfast) ›

Food diary for 20.09.2010 (Breakfast)

View Edit

Food Diary - Day & Meal type

Food Diary - Food photo

Food Diary - Annotate

Breakfast

Food	Portion Size
Bread - Brown	3 slice(s)
Egg	2 portion
Meat - Red	1 portion
Pulses (peas,beans,lentils)	1 portion

Carbohydrates (g)	48
Protein (g)	28
Fats (g)	57
Fibre (g)	4
Minerals	
Sodium (g)	580
Calcium (g)	247
Iron (g)	7
Vitamins	
Vitamins A (mg)	0
Vitamins B (mg)	4
Vitamins C (mg)	0
Vitamins D (mg)	2
Water(g)	180

Healthy Eating

My progress: 1 → 2 → 3

Amelia Student

logout

Healthy Eating

An investigation of healthy diet and nutrients.

Navigation

- Find my topic
- Decide my inquiry question or hypothesis
- Plan my methods, equipment & actions
- Collect my evidence
- Analyse and represent my evidence
  - Result Presentation
- Respond to my question or hypothesis
- Share and discuss my inquiry
- Reflect on my progress

My inquiries › Healthy Eating › Analyse and represent my evidence ›

During the data analysis, you need to see your data and transform it in order to make sense of it, e.g. through different graphs, charts, etc to help you understand relationships in data and see trends and patterns.

During this phase, you may find errors in data and so think why they occurred.

Activity	Description
Result Presentation	Graph for 19.09.2010 Graph for 20.09.2010

Fig. 5. a) Describing the nutritional content of a meal, b) viewing a comparison with RDI

The sequence of lessons followed the inquiry process shown in Figure 1, starting at Find My Topic. The investigation started with the teacher introducing the children to the representation of the inquiry process and the Healthy Eating Topic, and explaining how to use the technology. Between each of the lessons, the children could use the computer freely in school, at home or outside (for example in cafés). When they were familiar with uploading the photos from the camera to computer they were asked to record all food they ate over one or more days. For each dish, as part of the Collect My Evidence activity, the nQuire software provided pull-down menus to describe the content of the meal (Fig. 5a) and calculated the quantity its nutritional elements (protein, fat etc). Then, in Analyse and Represent my Evidence (Fig. 5b), the child's nutritional content for each day is plotted against the RDI. Other activities support the children in forming hypotheses, planning the study including posing questions to the nutrition expert, sharing and presenting findings in relation to the hypotheses, and reflecting on the inquiry process. Each child can view but not edit the data from other children in the group, and can explore possibilities by, for example, inputting data for an 'ideal diet' for a day.

A related-sample Wilcoxon test of learning gains showed a significant increase in the pre to post test scores for the intervention group ($T = 1$, $p<0.005$), whereas the pre and post test scores for the control group did not differ ($T=6$, p = ns). A test of the children's attitudes to science showed no difference between groups apart from the Enjoyment of Science Lessons subscale, where the scores of the control group decreased but those of the intervention group stayed the same.

The learning process was investigated though recordings from three cameras in the classroom for each of the lessons, focus group interviews with children, and individual interviews with the teacher, at the start, mid-point and end of the sequence of lessons. Logfiles from the computers were also analysed. These showed that all the children completed the inquiry process, including collecting photos of one or more meals outside the classroom and describe their content. The children engaged in scientific reasoning and communication, with opportunities for them to share and discuss each other's data. Some children commented that they had gained a better understanding of their diet and where it was lacking in nutrition. An unexpected finding was that at the early stages of the investigation some children were reluctant to share photos of their unhealthy food to peers within their work group and had to be assured by the teacher that their eating habits would not be revealed to the class. This issue of 'too personal inquiry', and the balance in inquiry-based learning between motivation and embarrassment should be explored in future research. The nQuire toolkit connected learning between classroom and home, though some children complained about carrying the equipment into school. The teacher was able to sustain interest over nine lessons and to manage the inquiry process, despite some difficulties in coordinating the flow of data from activities conducted outside the classroom.

5 Conclusion

The Personal Inquiry project has demonstrated that a constraint-based toolkit can maintain and guide inquiry-based learning between the classroom and non-formal settings. The nQuire system has successfully implemented a model of the inquiry learning process, providing a means for teachers or designers to author inquiry topics and for young learners to engage in a variety of exploratory activities including ranging from a study of the decay of food in the kitchen to the effect of noise pollution on the feeding habits of birds in the school grounds. A set of tools for inquiry, within a framework that enables preview, 'what if' exploration, and guided collection and analysis of authentic data, has the opportunity to extend science learning outside the classroom. Issues for future research include getting the right balance between engagement and embarrassment of collecting personal data, and the opportunity to implement the toolkit on a smaller, lighter, and more powerful device such as a smartphone or tablet computer.

Acknowledgments. We should like to thank the teachers and pupils of Hadden Park High School that participated in our study, team members at the Open University and the University of Nottingham, and our corporate partner ScienceScope. This project is supported by an ESRC/EPSRC Technology Enhanced Learning Grant for Personal Inquiry (PI): Designing for Evidence-based Inquiry Learning across Formal and Informal Settings, reference: RES-139-25-0382.

References

1. Dewey, J.: Science as Subject-matter and as Method. Science 31(787), 121–127 (1910)
2. Bereiter, C., Scardamalia, M.: Learning to Work Creatively with Knowledge. In: De Corte, E., Verschaffel, L., Entwistle, N., van Merriënboer, J. (eds.) Unravelling Basic Components and Dimensions of Powerful Learning Environments. EARLI Advances in Learning and Instruction Series, pp. 54–68 (2003)
3. Pea, R., Maldonado, H.: WILD for Learning: Interacting through New Computing Devices Anytime, Anywhere. In: Sawyer, K. (ed.) Cambridge Handbook of the Learning Sciences, pp. 427–442. Cambridge University Press, New York (2006)
4. De Jong, T., Van Joolingen, W.R., Giemza, A.: The SCY team: Learning by Creating and Exchanging Objects: The SCY Experience. Brit. Jnl. Educ. Tech. 41(6), 909–921 (2010)
5. Mulholland, P., Collins, T., Gaved, M., Paxton, M., Feisst, M., Scanlon, E.: nQuire: A Customizable Toolkit for Inquiry Learning across School, Home and Field Trip Locations. In: Montebello, M., Camilleri, V., Dingli, A. (eds.) Proc. mLearn 2010, World Conf. on Mobile and Contextual Learning, pp. 159–166. University of Malta, Valetta (2010)
6. Vogel, B., Spikol, D., Kurti, A., Milrad, M.: Integrating Mobile, Web and Sensory Technologies to Support Inquiry-based Science Learning. In: Proc. of the IEEE WMUTE Intl. Conf. on Wireless, Mobile and Ubiquitous Technologies in Education WMUTE 2010, Kaohsiung, April 12-16 (2010) (in press)
7. Anastopoulou, S., Sharples, M., Wright, M., Martin, H., Ainsworth, S., Benford, S., Crook, C., Greenhalgh, C., O'Malley, C.: Learning 21st Century Science in Context with Mobile Technologies. In: Traxler, J., Riordan, B., Dennett, C. (eds.) Proc. mLearn 2008 Conf., pp. 12–19. University of Wolverhampton, Wolverhampton (2008)
8. Scanlon, E., Littleton, K., Gaved, M., Kerawalla, L., Mulholland, P., Collins, T., Conole, G., Jones, A., Clough, G., Blake, C., Twiner, A.: Support for Evidence-based Inquiry Learning: Teachers, Tools and Phases of Inquiry. In: Proc. 13th Biennial Conference of the European Association for Research on Learning and Instruction (EARLI), Amsterdam August 25–29 (2009)
9. Llewellyn, D.: Inquire Within: Implementing Inquiry-Based Science Standards. Corwin Press, Thousand Oaks (2002)
10. Shimoda, T.A., White, B.Y., Frederiksen, J.R.: Student Goal Orientation in Learning Inquiry Skills with Modifiable Software Advisors. Science Education 86(2), 244–263 (2002)
11. Ausubel, D.P.: The Acquisition and Retention of Knowledge: A Cognitive View. Kluwer, Dordrect (2000)
12. Anastopoulou, S., Sharples, Ainsworth, S., Crook, C., O'Malley, C., Wright, M.: Creating Personal Meaning through Technology-supported Science Inquiry Learning across Formal and Informal Settings. Paper Accepted for Publication in International Journal of Science Education (forthcoming)

Early Prediction of Cognitive Tool Use in Narrative-Centered Learning Environments

Lucy R. Shores[1], Jonathan P. Rowe[2], and James C. Lester[2]

[1] Department of Curriculum & Instruction, North Carolina State University,
Raleigh, NC 27695
[2] Department of Computer Science, North Carolina State University, Raleigh, NC 27695
{lrshores,jprowe,lester}@ncsu.edu

Abstract. Narrative-centered learning environments introduce novel opportunities for supporting student problem solving and learning. By incorporating cognitive tools into plots and character roles, narrative-centered learning environments can promote self-regulated learning in a manner that is transparent to students. In order to adapt narrative plots to explicitly support effective cognitive tool-use, narrative-centered learning environments need to be able to make early predictions about how effectively students will utilize learning resources. This paper presents results from an investigation into machine-learned models for making early predictions about students' use of a specific cognitive tool in the Crystal Island learning environment. Multiple classification models are compared and discussed. Findings suggest that support vector machine and naïve Bayes models offer considerable promise for generating useful predictive models of cognitive tool use in narrative-centered learning environments.

Keywords: Narrative-Centered Learning Environments, Cognitive Tools, Self-Regulated Learning.

1 Introduction

Narrative-centered learning environments have become the subject of increasing attention in the AI in Education community [1,2,3,4,5,6]. By contextualizing learning within narrative settings, narrative-centered learning environments tap into students' innate facilities for crafting and understanding stories [7]. An additional benefit of narrative-centered learning environments is their capacity to discreetly scaffold students' learning processes by tightly integrating pedagogy and narrative elements. For example, narrative-centered learning environments have been developed that teach negotiation skills [3] and foreign languages [2] through conversational interactions with virtual characters. Also, scientific inquiry has been realized in interactive mysteries where students play the roles of detectives [8,9].

A particularly promising opportunity presented by narrative-centered learning environments is supporting self-regulated learning, i.e., students' ability to generate, monitor and control their cognitive, metacognitive, and motivational processes [10]. Students often possess varying degrees of competency in self-regulated learning [11].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 320–327, 2011.

Narrative plots and character roles can introduce contextualized *cognitive tools* that discreetly support self-regulation through elements of the story world. However, not all students use cognitive tools equally effectively; tools' effective use may need to be encouraged or guided. In narrative-centered learning, this support is ideally delivered by adapting narrative sequences to encourage effective cognitive tool use. Narrative-centered learning environments should be capable of making early predictions about how a student will use cognitive tools during a narrative-centered learning interaction, and subsequently use these predictions to inform decisions about tailoring the narrative and problem solving support.

This paper focuses on early prediction of students' cognitive tool use in a narrative-centered learning environment. The work extends previous research that identified a particular cognitive tool, a *diagnosis worksheet*, to be associated with significant content learning gains in the CRYSTAL ISLAND environment [18]. Several supervised machine-learning models are compared for early prediction of students' diagnosis worksheet usage, and potential directions for incorporating the predictive models into narrative-centered learning environments are discussed.

2 Related Work

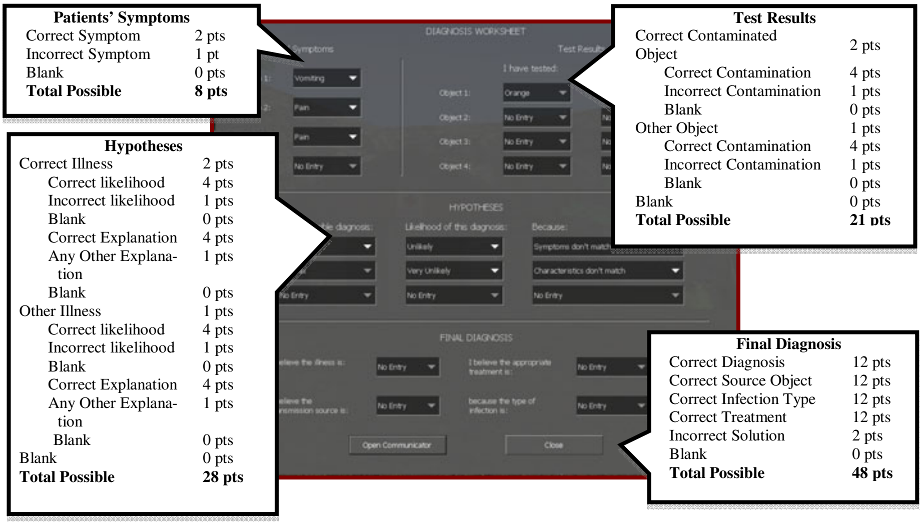
Narrative-centered learning environments are a class of serious games that tightly couple educational content and problem solving with interactive story scenarios. Recent work on narrative-centered learning environments has leveraged a range of techniques for providing effective, engaging learning experiences. FearNot! uses affectively-driven autonomous agents to generate dramatic, educational vignettes about bullying [1]. The Tactical Language and Culture Training System uses a range of AI techniques for speech recognition and virtual human behavior in interactive narrative scenarios for language and culture learning [2,6]. BiLAT is a story-centric serious game that enables students to practice cross-cultural negotiation skills during interactions with virtual characters [3]. BiLAT features a *leader preparation worksheet* that students complete to prepare for virtual negotiations, and it has similarities to the cognitive tool (diagnosis worksheet) that is the focus of this work. However, none of these systems explicitly model students' cognitive tool use during narrative-centered learning interactions to our knowledge.

Cognitive tools [12] are external, compensatory resources for problem solving. They are used to moderate student ability deficits and to maximize the effects of learning experiences. Cognitive tools for supporting self-regulated learning have been incorporated into several intelligent tutoring systems. For example, prompts for self-explanation have been shown to enhance learning during interactions with the Cognitive Tutor and SE Coach systems [13,14]. Similarly, self-regulatory prompts in the Betty's Brain environment have been shown to positively influence student learning and problem-solving behaviors [15]. During interactions with MetaTutor, students receive several forms of self-regulated learning instruction and, as a result, used self-regulation strategies more successfully [16].

Given the benefits of cognitive tools for supporting self-regulatory behaviors, it is critical to provide effective scaffolds for cognitive tool use. Schunk [11] explains that the development of self-regulatory skills occurs socially over time, making a one-size-fits-all approach to self-regulated learning support problematic. Developing predictive models of students' cognitive tool-use can enable intelligent tutoring systems to tailor support for students' self-regulated learning. Narrative-centered learning environments stand to benefit from these predictive models by adapting stories to support cognitive tool-use in a manner that is embedded in plots [17].

3 Crystal Island

Crystal Island 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. Students play the role of the protagonist who is attempting to discover the details of an infectious disease plaguing a research station. Several of the team's members have fallen ill, and it is the student's task to discover the cause of the outbreak (for more information, see [8]).



Patients' Symptoms	
Correct Symptom	2 pts
Incorrect Symptom	1 pt
Blank	0 pts
Total Possible	**8 pts**

Hypotheses	
Correct Illness	2 pts
Correct likelihood	4 pts
Incorrect likelihood	1 pts
Blank	0 pts
Correct Explanation	4 pts
Any Other Explanation	1 pts
Blank	0 pts
Other Illness	1 pts
Correct likelihood	4 pts
Incorrect likelihood	1 pts
Blank	0 pts
Correct Explanation	4 pts
Any Other Explanation	1 pts
Blank	0 pts
Blank	0 pts
Total Possible	**28 pts**

Test Results	
Correct Contaminated Object	2 pts
Correct Contamination	4 pts
Incorrect Contamination	1 pts
Blank	0 pts
Other Object	1 pts
Correct Contamination	4 pts
Incorrect Contamination	1 pts
Blank	0 pts
Blank	0 pts
Total Possible	**21 pts**

Final Diagnosis	
Correct Diagnosis	12 pts
Correct Source Object	12 pts
Correct Infection Type	12 pts
Correct Treatment	12 pts
Incorrect Solution	2 pts
Blank	0 pts
Total Possible	**48 pts**

Fig. 1. Crystal Island's diagnosis worksheet and associated scoring rubrics

An important element throughout Crystal Island's narrative and gameplay is the diagnosis worksheet (Figure 1). The worksheet consists of four sections: the *Patients' Symptoms* area where students record traits of the spreading disease; the *Test Results* area where students record findings from laboratory tests; the *Hypotheses* area where students record their beliefs about the likelihoods of candidate diagnoses; and the *Final Diagnosis* area where students report the identity, source, and treatment of the illness. A scoring scheme was devised to assess students' worksheet completion (see Figure 1). The total worksheet score was calculated by summing the sub-scores for each region (max = 105 points). Regions that involved complex inferences were weighted more heavily than regions that involved rote recording of information.

4 Predicting Cognitive Tool Use

The data used for the current investigation was collected during an experiment involving human participants from the eighth grade 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 investigation is a secondary analysis of the data and considers data from all conditions as a whole.

4.1 Data Collection

A total of 153 eighth grade students ranging in age from 13 to 15 (M = 13.3, SD = 0.47) interacted with the CRYSTAL ISLAND environment. Eight of the participants were eliminated due to incomplete data and eight participants were removed because they had prior experience with CRYSTAL ISLAND. Among the remaining 137 students (male: 77, female: 60), 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. The study was conducted prior to students' exposure to the microbiology curriculum unit of the North Carolina state standard course of study.

Students completed a series of pre-experiment questionnaires one week prior to playing CRYSTAL ISLAND. Post-experiment materials were completed immediately following the learning interaction. In addition to pre- and post-experiment measures, the CRYSTAL ISLAND software logged student actions, locations, and narrative state during gameplay, including the complete state of students' diagnosis worksheets.

4.2 Inductive Framework

A previous investigation indicated that students achieved significant learning gains as a result of their interactions with CRYSTAL ISLAND [8], and maintaining a thorough and accurate diagnosis worksheet was associated with improved learning outcomes, especially for students with low levels of prior microbiology knowledge [18]. Each student was classified as being either a low or high diagnosis worksheet student using a median split on their final diagnosis worksheet score. Students with low prior knowledge who earned high scores on their diagnosis worksheet experienced greater content learning gains than their low-scoring counterparts, and they performed comparably to high prior knowledge students on the microbiology post-test. Significant worksheet differences between the high and low groups began to appear after twenty-five minutes, which was almost halfway through the learning interaction. The current machine learning analysis focuses on early prediction, and it therefore classifies whether students will be high or low diagnosis worksheet users during the first twenty-five minutes of interaction, which is prior to the score divergence.

In order to identify useful predictor features for machine learning, a series of ANOVAs compared the gameplay characteristics of high and low diagnosis worksheet students. *In-game score*, a numerical sum that is based on a student's problem-solving engagement and effectiveness (for full details, see [8]), revealed significant differences between high and low diagnosis worksheet students after one minute of play. *Microbiology manual use,* measured by counting the number of times a student opened the feature on his/her in-game PDA device, was used significantly more by low diagnosis worksheet students around five minutes of elapsed gameplay. *Dialogue moves with non-player characters,* calculated as the total number of conversational turns with virtual characters, was greater among high diagnosis worksheet students after five minutes of play. *Virtual book reading,* calculated as the number of times a student opened in-game virtual books, was found to occur more frequently among high diagnosis worksheet students around ten minutes of gameplay.

A supervised learning approach leveraging the above predictors was taken in order to predict students' diagnosis worksheet group (high/low). All models were induced using the WEKA machine learning toolkit [19]. Naïve Bayes, decision tree, and support vector machine (SVM) classification techniques were compared to a *most frequent category* baseline (in this case, high diagnosis worksheet) for predicting whether a student would end the game as either a high or low diagnosis worksheet student. To enable early classification of student worksheet outcomes, instances of each model were learned for the 10, 12, 15, 18, 20, 22, and 25 minute marks. Predictor feature values were calculated using data up to the relevant time in the logs. In total, 28 models were trained and tested (including baseline). A student-level tenfold cross validation scheme was used to evaluate the performance of each model.

5 Findings

After ten minutes of gameplay, the best performing model (SVM) correctly classified 60.5% of instances, which was found to be significantly better than baseline (p < .05). The SVM model maintained significance over baseline for the 12-, 15-, 18-, 20-, 22-, and 25-minute models (see Table 1). The twelve-minute naïve Bayes model was found to significantly out predict baseline, correctly classifying 60.9% of instances (p < .05). Again, the naïve Bayes model was found to consistently and significantly outperform baseline for the remaining timestamps. However, decision tree models were not found to be reliable predictors of diagnosis worksheet performance.

Table 1. Accuracy percentages for classification models predicting diagnosis worksheet group with regard to time

Time (Minutes)	Baseline	Naïve Bayes	Decision Tree	SVM
10	56.9	57.7	52.7	60.6*
12	56.9	60.9*	53.3	62.0**
15	56.9	63.0**	55.6	62.1**
18	56.9	61.1**	55.4	63.6**
20	56.9	61.8**	54.6	61.2*
22	56.9	62.7**	56.0	61.5*
25	56.9	60.9*	55.4	63.8**

Note: * (p < .05) and **(p < .01) indicate significant performance above baseline.

The results indicate that SVM models are significantly more accurate than baseline for classifying students' diagnosis worksheet performance after ten minutes of interaction with the Crystal Island environment. Naïve Bayes modeling techniques are effective after twelve minutes have elapsed and tend to sustain higher levels of significance than SVM models. However, a series of ANOVAs found both the twelve-minute SVM and fifteen-minute naïve Bayes models to have higher increased significance over baseline than the ten-minute and twelve-minute models, respectively. The decision tree models' lack of significant improvement over baseline suggests that they may not be well-suited to the current task. The areas under the Receiver Operating Characteristic (ROC) curve for the ten-minute and twelve-minute SVM and twelve-minute and fifteen-minute naïve Bayes models are displayed in Table 2.

Table 2. Areas under the ROC curve for the best performing time-based models

Class	Ten-Minute SVM	Twelve-Minute SVM	Twelve-Minute Naïve Bayes	Fifteen-Minute Naïve Bayes
High Diagnosis Worksheet	0.55	0.61	0.68	0.74
Low Diagnosis Worksheet	0.54	0.65	0.66	0.73

As previous analyses of the diagnosis worksheet suggest, efficient use of the diagnosis worksheet is particularly advantageous for low prior knowledge students in terms of content learning gains [18]. Although accurately classifying students into four groups, high/low prior knowledge and high/low worksheet, is a more challenging problem than the previous two-group task, this finer-grained classification can better inform real-time, personalized scaffolding, particularly to assist low prior knowledge students in utilizing the diagnosis worksheet. A low diagnosis worksheet/low prior knowledge student might benefit from both tool use and content-related scaffolding; whereas, a low diagnosis worksheet/high prior knowledge student might find content scaffolding to be redundant, running the risk of expertise reversal effects [20]. Additionally, this finer-grained classification opens opportunities for tailoring scaffolding without the need for prior information about students.

In a follow-up analysis, models were created to classify students as high diagnosis worksheet/high prior knowledge, high diagnosis worksheet/low prior knowledge, low diagnosis worksheet/high prior knowledge, or low diagnosis worksheet/low prior. Again, a median split was used to distinguish performance on the microbiology pre-test. The highest performing model (SVM) accurately classified 40.00% of instances after ten minutes of gameplay, which significantly outperformed the baseline model (33.57%; $p < .01$). The SVM model maintained significance over baseline models for all time periods ($p < .01$). Naïve Bayes 10-, 12-, and 15-minute models were found to significantly out predict baseline models ($p < .01$); however, this dominance was not found for later time points after fifteen minutes. Again, the decision tree model was observed to be insufficient for accurately classifying the students.

6 Conclusions

Narrative-centered learning environments offer important opportunities for supporting effective self-regulated learning behaviors. By incorporating cognitive tools into narrative plots and character roles, narrative-centered learning environments can discreetly scaffold complex cognition and metacognition during problem solving. Previous research has indicated that not all students use cognitive tools equally effectively. In order to dynamically adapt narrative plots to support effective cognitive tool use, it is necessary for narrative-centered learning environments to make early predictions about how students will use provided cognitive supports.

Several machine-learning models were trained and evaluated for predicting students' diagnosis worksheet performance in the CRYSTAL ISLAND learning environment. Support vector machine (SVM) and naïve Bayes models were found to achieve promising predictive accuracy as early as ten minutes into a learning interaction. SVM and naïve Bayes models were also found to be a promising method for jointly predicting diagnosis worksheet performance and microbiology prior knowledge, although more work is needed to enhance the accuracy of these fine-grained classifications. Continued investigations of machine-learned models for predicting cognitive tool-use may introduce opportunities for dynamically scaffolding students' self-regulatory behaviors in narrative-centered learning environments.

It should be noted that the machine-learned models were trained using only 137 instances, a relatively small dataset for machine learning purposes. As a consequence, very few predictor features were used for training the models. This may explain the relatively low accuracies, particularly for the decision tree models. Future work will utilize a larger corpus of students with additional predictor features in hopes of improving predictive accuracy. Furthermore, incorporating the models into runtime narrative-centered learning environments will enable further investigations to determine whether model-informed narrative adaptations can lead to more effective use of the diagnosis worksheet, and consequently improved learning outcomes.

Acknowledgments. The authors thank members of the IntelliMedia group for their assistance, Omer Sturlovich and Pavel Turzo for use of their 3D model libraries, and Valve Software for access to the Source^{TM} SDK. This research was supported by the National Science Foundation under Grants REC-0632450, DRL-0822200, CNS-0540523, and IIS-0812291. This work was also supported by the National Science Foundation through a Graduate Research Fellowship. 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. Aylett, R.S., Louchart, S., Dias, J., Paiva, A.C.R., Vala, M.: FearNot! - an experiment in emergent narrative. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 305–316. Springer, Heidelberg (2005)
2. 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)

3. Kim, H., Durlach, L., Forbell, C., Marsella, P., Hart: BiLAT: A Game-Based Environment for Practicing Negotiation in a Cultural Context. Int. J. AIED 19, 289–308 (2009)
4. Niehaus, J., Riedl, M.: Toward Scenario Adaptation for Learning. In: 14th International Conference on AI in Education, pp. 686–688 (2009)
5. Thomas, J., Young, R.M.: Using Task-Based Modeling to Generate Scaffolding in Narrative-Guided Exploratory Learning Environments. In: 14th International Conference on Artificial Intelligence in Education, pp. 107–114. IOS Press, Amsterdam (2009)
6. 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)
7. Bruner, J.: Acts of Meaning. Harvard University Press, Cambridge (1990)
8. Rowe, J.P., Shores, L., Mott, B., Lester, J.: Integrating learning and engagement in narrative-centered learning environments. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 166–177. Springer, Heidelberg (2010)
9. Ketelhut, D.: The Impact of Student Self-Efficacy on Scientific Inquiry Skills: An Exploratory Investigation in River City, a Multi-User Virtual Environment. J. Science Ed. and Tech. 16, 99–111 (2007)
10. Zimmerman, B.J.: Self-Regulated Learning and Academic Achievement: An Overview. Ed. Psychologist 25, 3–18 (1990)
11. Schunk, D.: Social Cognitive Theory and Self-Regulated Learning. In: Zimmerman, Schunk (eds.) Self-Regulated Learning and Academic Achievement: Theoretical Perspectives, 2nd edn., pp. 125–152. Lawrence Erlbaum Associates, Mahwah (2001)
12. Lajoie, S., Derry, S.J. (eds.): Computers as Cognitive Tools. Lawrence Erlbaum Associates, Hilldale (1993)
13. Aleven, V., Koedinger, K.: An Effective Metacognitive Strategy: Learning by Doing and Explaining with a Computer-Based Cognitive Tutor. Cog. Sci.: A Multidisciplinary Journal 26, 147–179 (2002)
14. Conati, C., VanLehn, K.: Towards Computer-Based Support of Meta-Cognitive Skills: a Computational Framework to Coach Self-Explanation. Int. J. AIED 11, 398–415 (2000)
15. Biswas, G., Jeong, H., Roscoe, R., Sulcer, B.: Promoting Motivation and Self-Regulated Learning Skills through Social Interactions in an Agent-Based Learning Environment. In: 2009 AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems (2009)
16. Azevedo, R., Witherspoon, A., Chauncey, A., Burkett, C., Fike, A.: MetaTutor: A Metacognitive Tool for Enhancing Self-Regulated Learning. In: 2009 AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems, pp. 34–39 (2009)
17. Rowe, J., Shores, L., Mott, B., Lester, J.: A Framework for Narrative Adaptation in Interactive Story-Based Learning Environments. In: Workshop on Intelligent Narrative Technologies III at the 5th International Conference on Foundations of Digital Games (2010)
18. Shores, L., Nietfeld, J.: The Role of Compensatory Scaffolds for Inquiry Learning in Narrative-Centered Learning Environments. To appear in: American Educational Research Association (2011) (in press)
19. Witten, I., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufman, San Francisco (2005)
20. Kalyuga, S., Ayers, P., Chandler, P., Sweller, J.: The Expertise Reversal Effect. Ed. Psychologist 38, 23–31 (2003)

Feedback during Web-Based Homework: The Role of Hints

Ravi Singh[1], Muhammad Saleem[1], Prabodha Pradhan[1],
Cristina Heffernan[1], Neil T. Heffernan[1], Leena Razzaq[2], Matthew D. Dailey[1],
Cristine O'Connor[3], and Courtney Mulcahy[3]

[1] Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609 USA
{ravisingh,msaleem,prpradhan,ch,nth,mdailey}@wpi.edu
[2] University of Massachusetts Amherst, 140 Governors Drive, Amherst, MA 01003 USA
leena@cs.umass.edu
[3] Oak Middle School, 45 Oak Street, Shrewsbury, MA 01545 USA
{COConnor,CMulcahy}@shrewsbury.k12.ma.us

Abstract. Prior work has shown that computer-supported homework can lead to better results over traditional paper-and-pencil homework. This study about learning from homework involved the comparison of immediate-feedback with tutoring versus a control condition where students got feedback the next day in math class. After analyzing eighth grade students who participated in both conditions, it was found that they gained significantly more (effect size 0.40) with computer-supported homework. This result has practical significance as it suggests an effective improvement over the widely used paper-and-pencil homework. The main result is followed with a second set of studies to better understand this result: is it due to the timeliness of feedback or quality tutoring?

Keywords: evaluation of CAL systems; intelligent tutoring systems; interactive learning environments; secondary education; teaching/learning strategies.

1 Introduction

The increasing popularity of computer assisted learning (CAL) applications in schools and colleges has led to the development of various web-based CAL tools that aim towards improving the quality of student learning. The scope of CAL systems has expanded from tools used in classrooms to web-based applications that are capable of supporting and guiding students through homework as well. Many preparatory tools for mathematics tutoring have been developed and tested. WebWork (www.webwork.rochester.edu), WebAssign (www.webassign.com) and Blackboard (www.blackboard.com) are web applications that are already popular across colleges in the US. The introduction of educational technology has also been increasing recently in K-12 grades. For instance, since 2002, the state of Maine has introduced a 1:1 laptop program for 7th and 8th grade students and their teachers. A study [1] on the impact of 1:1 computing programs has shown increased motivation and engagement in classrooms and better retention of content material based on reports from teachers.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 328–336, 2011.

The use of web-based homework by teachers is more feasible through programs such as Maine's program and through the increasing development in educational technology. Teachers may use web-based homework to supplement or replace conventional teaching methods such as paper-and-pencil homework. However, the increasing popularity of such technology brings into question the advantages in terms of effectiveness of web-based homework for students. Concerns are often raised about the cost-effectiveness of technology towards improving the standard of education.

Feedback on homework has been shown to have a large positive effect on student learning [2]. Most CAL systems used for homework attempt to improve the quality and timeliness of the feedback for homework. But, the effectiveness of computer-supported homework is still largely debated. Thus, in order to determine how a computer-supported tutoring system may affect student performance on their homework, a study by Mendicino, Razzaq & Heffernan [3] analyzed the effectiveness of computer-supported homework over traditional paper-and-pencil homework for K-12 students. Mendicino et al. reported an effect size of 0.61 in favor of computer-supported homework over paper-and-pencil homework. However, a study conducted using WebAssign [4] to deliver computer-supported homework for college level students showed no significant difference in the student performances.

In this study, we aim to improve the experimental design of Mendicino et al. [3] to further understand the effectiveness of web-based tutoring systems for delivering homework and improving student learning for K-12 students. We further analyze the characteristics of this tutoring mechanism to determine what factors contribute towards its effectiveness. The ASSISTment System was used for this study.

The ASSISTment System (www.assistments.org) is a web-based tutoring system, capable of offering instructions to students while providing detailed evaluations of their performance to teachers [5]. The system integrates assistance and assessment to efficiently tutor students in mathematics and is being used by middle and high school teachers throughout Massachusetts. Teachers may use the system as part of their coursework to assist students in learning while also obtaining detailed reports on individual students. Teachers may then identify difficulties students may be facing to tailor their instruction to be more effective. The system is free to use and supported by grants from the U.S. Department of Education and the National Science Foundation.

2 Experiments

Three experiments for investigating the effectiveness of web-based homework were conducted. In the first experiment (Experiment-1), we attempt to strengthen the claims made by Mendicino et al. [3] comparing computer-based homework with paper-and-pencil homework. This study compares learning gains for students in two conditions; *Immediate Feedback with Tutoring* (IFT: where students received homework with tutoring and immediate feedback on each problem) and *Business as Usual* (BAU: where student received no feedback). However, this study confounded the effects of Immediate Feedback and Tutoring. Experiment-2A & 2B were designed to understand the independent effect of Tutoring controlling for immediate feedback.

2.1 Experiment-1

Mendicino, et al. [3] compared web-based homework with paper-and-pencil homework. The students in the web-based homework condition used ASSISTments to complete their homework and so received immediate feedback through hints and scaffolding. The paper-and-pencil homework group completed their homework on paper and received feedback the next day in class. Hence, this study analyzed whether homework could be improved with immediate feedback with tutoring and found positive results in its favor.

However, as stated earlier Mendicino et al. [3] chose student classes as units of assignment for their conditions, but analyzed data at the student level. According to the What Works Clearinghouse [6] the unit of assignment should be the unit of analysis. Having classes as the unit of assignment and analyzing at the student level may lead to overestimation of the observed effects. Also a large Randomized Controlled Trial (RCT) number is recommended to evaluate educational software in reading and math. Mendicino, et al. had an RCT number of 4, as there were 4 different classes. Also, the use of the same test for pre- and post-tests may have created a test-retest effect and contributed to overestimation of learning rates.

In Experiment-1 we attempt to replicate this experiment by expanding the sample size and making some critical changes to the design and procedure. One major change in the experiment is the use of ASSISTments by both treatment and control groups to complete their homework. Instead of providing paper-and-pencil homework to the control group the students received Test Mode problems, and the treatment group received Tutoring Mode problems. Problems in Test Mode provide no feedback to the students and so were used as a replacement for paper-and-pencil homework. The unit of assignment was at the student level and not the class level and counter-balanced pre- and post-tests were used rather than using the same tests for both cases.

Experimental Design. The students in eight classes were randomly assigned based on their last names to the Immediate Feedback with Tutoring (IFT) condition or the Business as Usual (BAU) condition. After treatment, the conditions for the student groups were then switched. This provided a repeated measure for each participant. All students received two computer-based homework assignments as per their conditions. Before the homework, every student was given a pre-test and after completion of the homework a post-test was administered. In order to account for any test-retest effect two different test forms (Form-A1 & Form-B1) were randomly distributed to students for the first pretest. The students who received Form-A1 were provided with Form-B1 for the first post-test and vice versa. The same was done with test forms (Form-A2 & Form-B2) for the second round of pre- and post-tests when the second homework was assigned. These tests were paper-and-pencil based.

The pre- and post-test and homework assignments consisted of 10 problems each that were intended to be a Geometry and Number-Sense Review for the students. The assignment tested understanding of supplementary angles, properties of triangles, properties of quadrilaterals and parallel lines, transversals and the Pythagorean Theorem. The pre- and post-test consisted of problems that were very similar to problems from the homework assignment.

Procedure. Eight 8th grade classrooms with computers and the students' home computers were the settings used for this study. The students were familiar with the ASSISTment system and had used it for math homework before. Two teachers instructed four classes each and the total number of students was 172.

On the first day of the experiment all students completed a pre-test in class. The students then were assigned homework to be done with ASSISTments. The IFT group received homework with immediate feedback in terms of correctness with tutoring, which consisted of 3-4 hints for solving the problem, and the BAU group received no feedback at all. The following day, the teachers reviewed selected problems from the homework in class. After the review, students completed a post-test. The next day, students completed another pre-test and were assigned a second homework where the treatment and control groups were switched to provide a repeated measure for each student. The teachers reviewed the homework on the following day. This review session was videotaped to analyze the quality of their feedback. The students completed the second post-test after the review. The data from the first and second round of pre- and post-tests were then analyzed as paired samples**.**

Results. The eight classes included in this study had a total of 172 students. For the first homework assignment, 22 students in the *BAU* condition and 15 students in the *IFT* condition did not complete the homework. For the second homework, 14 students in the *BAU* conditions and 23 students in the *IFT* condition did not complete the homework. After the first homework assignment was assigned, some students in the *BAU* condition might have received feedback after completing the assignment by visiting a report page on the ASSISTments website. Due to this fact, we excluded 30 students who received this form of immediate feedback when in the *BAU* group. For the second homework assignment, the report page was disabled so that students in the *BAU* condition could not receive immediate feedback or tutoring from the system. Excluding these students and those who were not present for all or part of the experiment, 68 students participated in the study.

Based on the gain scores for the students, overall learning was observed in both conditions. The mean gain for the students in the *IFT* condition was 2.4 (SD=1.81), whereas it was 1.63 (SD=1.93) for the *BAU* condition. Both gain scores were reliably different than zero, $t(67)=10.9$, $p < 0.001$ for *IFT* and $t(67)=6.97$, $p < 0.001$ for *BAU*. Furthermore, comparing the gain scores of the two different conditions showed a reliable difference, $t(67) = 2.322$, $p = .023$, with higher gain scores in the *IFT* condition than in the *BAU* condition. The effect size observed in the direction of *IFT* was 0.40 with the 95% confidence interval of (-0.03 – 0.86).

The results suggest that students learned more from homework in the *IFT* condition as opposed to the *BAU* condition. Figures 1a and 1b show the distribution of gain scores for the students in the two conditions. From the graphs, it can be seen that the students in the *IFT* condition earned higher gain scores than those in the *BAU* condition and that our analysis is not sensitive to a few students.

However, certain factors such as the difference in the pre- and post-test forms and the quality of delayed feedback provided by the two different teachers may have had a significant impact on the results of student learning. We decided to dig deeper into the effect of these factors by doing additional analysis. We chose to focus on the second day of the experiment since the teacher reviews were videotaped then.

To test the effect of the different pre- and post-tests, a one-way ANOVA was performed with respect to Form as the independent variable. The test Forms that were assigned did not reliably predict post-test gains, $F(1, 119) = 0.78$, $p = 0.38$. We conclude that the test forms were balanced and excluded Forms from the analysis.

The next factor considered was the difference in homework reviews provided by the two teachers. Based on the video recordings of the homework reviews, there was a significant difference in the way they reviewed homework. Teacher B spent significantly more time reviewing more problems (mean = 19.8 minutes, 4.5 problems) than Teacher A (mean = 8 minutes, 3.5 problems). Teacher B also spent time reading the problems and answers to the students while Teacher A did not read the problems. We had no reason to believe that the quality of feedback provided by the two teachers was significantly different, but given the differences in the review methods we decided to test for the effect of teacher.

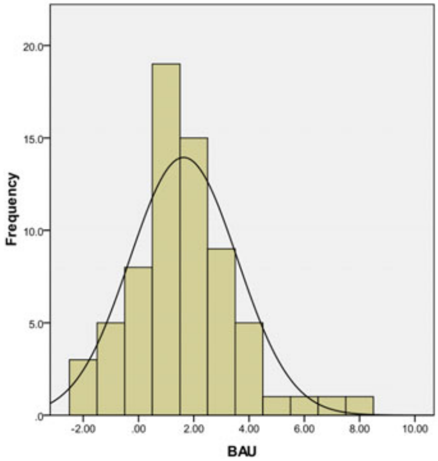


Fig. 1a. Distribution of gain scores for BAU

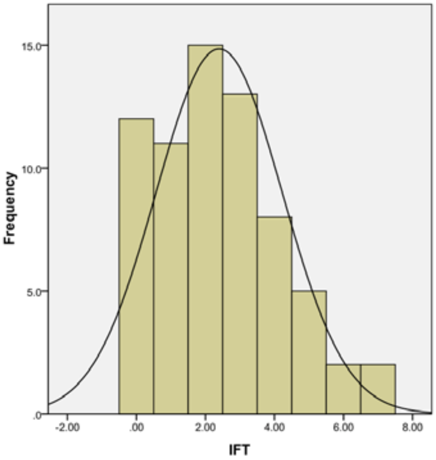


Fig. 1b. Distribution of gain scores for IFT

Comparing gain scores of students of Teacher A and Teacher B showed that the average gain for students who had Teacher B (M = 2.44, SD = 1.76) was higher than the average gain for students of Teacher A (M = 1.92, SD = 2.03). This could be due to the amount of time spent by Teacher B on reviewing homework problems the next day in class. However, a one-way ANOVA with Teacher as a factor showed that the difference was not significantly reliable, $F(1, 118) = 2.150$, $p = 0.145$, based on Teacher. But it did seem reasonable to keep in the model as we know from the review sessions that the two teachers spent different amounts of time going over problems. We continue our analysis considering Teacher as a potential factor in the model.

The results of a two-way ANOVA with Teacher and Condition as factors showed Condition to be a reliable factor, $F(1, 118)$=8.27, p=0.005, and the Teacher by Condition was also significantly reliable, $F(1, 118)$=8.38, p=0.005, in predicting post-test gains. Also, while Teacher A's students in the *IFT* condition had higher mean scores they are not reliably higher than Teacher B's students in the *IFT* condition.

The difference in mean scores between the *BAU* and *IFT* conditions for Teacher B suggest that delayed but quality feedback from teachers can make *BAU* homework as effective as the *IFT* homework if they spend a lot of time going over the questions.

Discussion. The positive results obtained from this experiment certainly reinforced the observation that computer-supported homework can produce superior results to more traditional approaches. The observed effect was smaller than the effect size reported by Mendicino et al., which was 0.61. The analysis based on Teacher by Condition showed that teachers may be able to make delayed feedback as effective as immediate feedback with tutoring. However, to do this Teacher B spent significantly more time than Teacher A reviewing the homework in class. The marginal gains for the two teachers were not significantly different. This seems to suggest that an effective strategy would be to give computer-based homework with tutoring and utilize the homework review time more effectively, perhaps going over new material.

Given the positive gains for students in the IFT condition, it seemed reasonable to examine the effects of immediate feedback and tutoring separately. Thus Experiments 2A and 2B look at student gains with immediate feedback with and without tutoring.

2.2 Experiment-2A

This experiment was conducted to analyze the effects of tutoring over immediate feedback. The two conditions for this experiment were *Tutoring* and *No Tutoring*. In the *No Tutoring* condition no tutoring is provided and the students are only given feedback on the correctness of their answers and provided with the right answer if they answered incorrectly using Correctness Mode problems. In the *Tutoring* condition students could ask for up to 3-4 hints on solving the problem before being presented with the final answer. With this experiment we hoped to understand the size of the effect with respect to Tutoring while controlling for the timeliness of feedback.

Experimental Design. The students were assigned four homework assignments. The homework assignments were completed by the students using the ASSISTment system at home. The students were randomly placed into either the *Tutoring* condition or the *No Tutoring* condition by the system. The homework assignments were designed such that the first half of each of the four assignments could be treated as the pre-test and the second half would act as the post-test for the experiment.

The content for this experiment consisted of problems that required the use of the Pythagorean Theorem to find the lengths of sides of triangles or deduce the area of geometric figures. The problems were similar to problems from the Connected Mathematics Project – “Looking for Pythagoras” unit.

Procedure. The setting and participants were the same as Experiment-1. The students were familiar with Correctness and Tutoring Mode in ASSISTments. The homework was assigned as a review after students were done with their regular bookwork. The students were not told that the assignment included pre- and post-tests.

Results. There were 72 students who finished at least one out of the four homework assignments. Out of the 72 students, 32 students were placed in each condition at least once over the period of the four homework assignments. The gain score in a condition for each student was calculated to be the average gain score for homework assignments completed by the student in that condition. The average gain score for the Tutoring group was 1.0 (N = 32, SD = 1.16) and the average gain score for the No Tutoring group was 0.4 (N = 32, SD = 1.16). The analysis for comparing the gain scores in the two test conditions showed a reliably significant difference between the two conditions, $t(31) = -2.178$, $p = 0.037$, in favor of *Tutoring* with an effect size of 0.54. The 95% confidence interval of the observed effect size was (0.14 – 0.95).

Discussion. The effect observed in the direction of *Tutoring* indicates that providing tutoring for students does significantly improve their learning. These results suggest that tutoring homework is more effective than immediate feedback alone. However, the effect size (0.54) is higher than that observed in Experiment-1 in favor of *IFT* over *BAU*. We expected the effect to be smaller when controlling for immediate feedback.

2.3 Experiment-2B

After observing a surprisingly large effect size in Experiment-2A, the purpose of Experiment 2B was to see if the result could be replicated.

Experimental Design. The students were randomly placed in the two experiment groups based on their last names. The structure of the homework assignment was similar to the ones used in Experiment-2A but contained more problems.

The problems assigned for this experiment dealt with exponential and linear growth rates. The problems were similar to problems from the Connected Mathematics - "Growing Growing Growing" unit and were used as a review.

Procedure. The procedure was the same as Experiment-2A.

Results. Out of the 172 students, 20 students did not start the homework assignment. Three students, two placed in *Tutoring* and one placed in *No Tutoring* started but did not complete the assignment. The remaining 149 students completed the assignment. We excluded students who received perfect scores on the pretest, which left us with 107 students for our analysis. Overall, the average gain score was 0.80 (SD = 1.48) and the overall scores were reliably different than zero, $t(106) = 5.61$, $p < 0.001$.

The average gain score for the Tutoring group was 1.16 (N=64, SD=1.26) and the average gain score for the No Tutoring group was 0.28 (N=43, SD=1.6). When comparing the gain scores in the two conditions a reliable difference in favor of *Tutoring* was observed, $t(74.34)=2.97$, $p=0.004$. The observed effect size was 0.54 with a 95% confidence interval of (0.22–1.01).

Discussion. Upon replication of Experiment-2A, we found that the size of the effect was comparable to that observed in Experiment-2A. This indicates that tutoring had a large impact on student gains and that it is more beneficial to have tutoring in addition to immediate feedback. Most math textbooks provide answers to selected problems that can serve as immediate feedback for homework, but students often do not get immediate tutoring. This suggests that learning can be significantly improved from computer-based homework by providing immediate feedback with tutoring.

3 Conclusions and Contributions

The experiments presented in this paper help strengthen the claims made by Mendicino et al. [3], while improving the experiment design. The results suggest that spending proportionately more time in class going over homework can make learning equivalent to giving computer-based homework. Furthermore, the results show an advantage of tutoring when controlling for immediate feedback.

Strictly speaking, we did not look at the amount of learning in the *IFT* condition if the Teacher *did not* give a review the next day. It would be interesting to see if the *IFT* condition results stay as high as they do if the teacher did not go over the homework at all the next day. But, it seems that there is some value in going over the homework in class as seen by the strong gain scores for students in the *BAU* condition for Teacher B. We propose a future study that tests the value of homework review after receiving *IFT*. If the value of reviewing homework is small then a cost-benefit analysis should be considered to see if time is better used to introduce new content.

We did a survey of six curriculum supervisors from different towns and asked them "What is the appropriate amount of time for teachers to spend going over homework?" We got the following answers: 8-10, 10, 10, 10, 5-10 and 10-15 minutes. Based on these responses, we assume that 10 minutes is the right amount of time teachers should be spending on average going over homework. If we assume that students are spending 20 minutes doing their homework and their class period is effectively 40 minutes long, they have a total "math" time of 60 minutes combining homework and class. If students spend 20 minutes doing homework and 10 minutes reviewing homework in class, they are spending half of their "math" time on homework. This amount of time for review is probably significant for helping students learn, but also means that a better method can have an impact of practical significance. If we can improve homework by even half a standard deviation, it would be reasonable to see if we can improve student performance on state tests.

Our results reinforced the observation [3] that computer-supported homework is better for students compared to traditional homework approaches. It can be claimed that detailed scaffolding and hints can improve homework performance significantly. Systems such as ASSISTments can provide the necessary tools for improving homework by providing quality tutoring and immediate feedback and allowing teachers to identify areas in which students are struggling at an individual and class level, advantages that are much harder to achieve with traditional homework.

Acknowledgments. This material is based upon work supported by the National Science Foundation (NSF), under Grant DGE-0742503 and under Grant #0937060 to the Computing Research Association (CRA) for the CIFellows Project. For the full list of over a dozen funders of ASSISTments please see http://www.webcitation.org/5xp605MwY.

References

1. Bebell, D.: Technology promoting student excellence: An investigation of the 1st year of 1:1 computing in New Hampshire middle schools. In: Technology and Assessment Study Collaborative, Boston College (2005)

2. Walberg, H.J., Paschal, R.A., Weinstein, T.: Homework's powerful effects on learning. Educational Leadership 1995, 76–79 (1985)
3. Mendicino, M., Razzaq, L., Heffernan, N.T.: Comparison of Traditional Homework with Computer Supported Homework. Journal of Research on Technology in Education 41(3), 331–359 (2009)
4. Bonham, S.W., Deardorff, D.L., Beichner, R.J.: Comparison of student performance using Web- and paper-based homework in college-level physics. Journal of Research in Science Teaching 40(10), 1050–1071 (2003)
5. Razzaq, L., Heffernan, N., Koedinger, K., Feng, M., Nuzzo-Jones, G., Junker, B., Macasek, M., Rasmussen, K., Turner, T., Walonoski, J.: Blending Assessment and Instructional Assistance. In: Nedjah, N., de Macedo Mourelle, L., Neto Borges, M., Nunesde Almeida, N. (eds.). Intelligent Educational Machines within the Intelligent Systems Engineering Book Series, pp. 23–49. Springer, Heidelberg (2007)
6. What Works Clearinghouse Standard of Evidence for Reviewing Studies, Institute of Education Sciences, http://ies.ed.gov/ncee/wwc/pdf/wwc_version1_standards.pdf (retrieved May 2010)

Transferring Teaching to Testing – An Unexplored Aspect of Teachable Agents

Björn Sjödén[1], Betty Tärning[1], Lena Pareto[2], and Agneta Gulz[1]

[1] Lund University Cognitive Science, Sweden
{Bjorn.Sjoden,Betty.Tarning,Agneta.Gulz}@lucs.lu.se
[2] Media Production and Informatics Departments, University West, Sweden
Lena.Pareto@hv.se

Abstract. The present study examined whether socio-motivational effects from working with a Teachable Agent (TA) might transfer from the formative learning phase to a summative test situation. Forty-nine students (9-10 years old) performed a digital pretest of math skills, then played a TA-based educational math game in school over a period of eight weeks. Thereafter, the students were divided into two groups, matched according to their pretest scores, and randomly assigned one of two posttest conditions: either with the TA present, or without the TA. Results showed that low-performers on the pretest improved significantly more on the posttest than did high-performers, but only when tested with the TA. We reason that low-performers might be more susceptible to a supportive social context – as provided by their TA – for performing well in a test situation.

Keywords: Learning-by-teaching, teachable agent, assessment, transfer.

1 Introduction

Teachable Agents, *TAs*, is a form of educational technology based on the idea that a good way to learn is to teach someone else. In brief, a TA is a computer agent that is taught by a student, where AI techniques guide the agent's behavior based on what it is taught. Students can revise their TA's knowledge (and their own) based on the agent's behavior [1, 2]. Numerous studies have shown that TA-based software can be powerful in terms of learning outcomes. For example, students working with a TA exhibited deeper causal understanding than students using the same software without a TA [1], and they produced more accurate concept maps [3]. In a comparison to "pen and paper"-methods, Chin and colleagues [4] demonstrated that an equivalent system using a TA provided "added value" in terms of students learning more complex ways of reasoning and being more successful in taking on new learning material.

Lately, there has been an increased focus on the social and motivational aspects of TA software. In particular, students' feelings of responsibility and engagement from developing a social relation to their TAs has been proposed as an explanatory mechanism as to why students seem to make greater efforts and spend more time on learning material when using a TA, than when alone. Chase and colleagues [5] re-

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 337–344, 2011.

ported two studies to this effect, noting that students acted as though their TAs were sentient, semi-independent beings, which engage in mental activity and were given partial credit for the outcomes. Students instructed to learn for their TA, rather than for themselves, were more inclined to approach, discuss and attempt to revise errors and misunderstandings. The authors suggest that the TA may provide an “ego-protective buffer” by offering a means for students to distribute the responsibility for errors and mistakes, thereby decreasing their fear of failure.

In sum, TA studies suggest that the sense of social relationship between students and their TAs can have positive effects on learning through an impact on motivation and engagement. But what happens when turning from a learning situation to a test situation? Can the sense of meaningfulness, engagement and responsibility developed in relation to the TA be reestablished when performing a formal test and lead to improved performance? We explore these questions by having a TA, which students have interacted with in an educational game, reappear in a summative assessment form detached from this software[1]. The scope of the present article is limited to the possible effects of TAs specifically, not of other pedagogical agents, for assessment.

1.1 Relation to Previous Studies and Present Research Aims

Test situations are generally included as central features within TA software. Students get feedback on how well they have “taught” their TA by testing the TA under various forms, for example in a game show-like quiz [5, 6]. Although a test of the TA becomes an implicit test of the student’s knowledge, it is not presented explicitly so. To our knowledge, no previous study has targeted how socio-motivational factors associated to a TA may replicate in a test situation, when the TA is removed from the original learning software and put in a completely new environment. That is where the novelty of the present study resides.

Our aim was to examine students’ performance on a formal summative test, taken by the student (not the TA) in a situation clearly separated from the learning phase and the primary TA environment. Would performance be affected by the mere visual presence of the student’s TA in this situation? We made use of a TA-based learning game in mathematics [7] and focused on the motivation and engagement aspects related to the TA’s role as a “protégé” [cf. 5] and learning companion. In two other recent studies of this TA system, we found empirical support for the following: (a) that students playing the TA game improved performance on subsequent math tests compared to students not playing the game [8], and (b), that students became emotionally involved with their TAs and related to it socially while playing [9]. The two main questions posed for the present study were:

1. Following an extended period of learning with the TA, would test performance differ between students who performed a standard summative math test in the presence of their TA, and students who performed the same test without their TA?

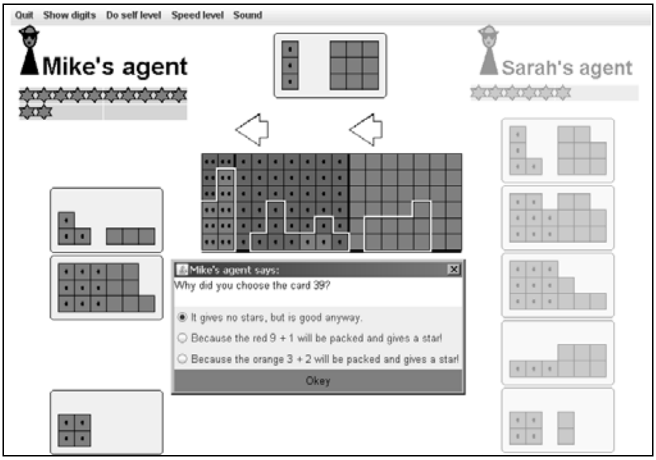
2. Would the presence of the TA affect students’ experiences, in terms of their ratings of engagement, effort, difficulty and confidence for taking the standard summative math test, and how would this relate to their test performance?

[1] This does not imply that we take sides with traditional, summative assessment before formative assessment more closely integrated with learning (in which TAs can be productively involved). However, summative tests are commonly used in education.

Next, we describe the TA environment and the method we used for measuring how the socio-motivational effects of the TA might transfer from the learning phase to a test situation. We report our primary analysis of the results and how these relate to relevant subgroups of students and discuss some possible explanations.

2 The TA Environment: An Educational Math Game

The TA learning environment used in the present study is an educational game in elementary mathematics [7], specifically aimed at training the base-10 system (such as carry-overs and borrowings). The game employs a board-game design, including playing cards and a common game board, with several game modes and levels of difficulty. All arithmetic operations are visualized, using the graphical metaphor of squares and boxes that can be "packed" or "unpacked" in numbers of 10. Students typically play the game in pairs, either in their own name or with a TA. A game move consists of picking a card that depicts a certain constellation of squares and boxes, which then adds or subtracts to the present (previously played) squares and boxes on the game board. The goal is to consistently pick the cards that, in combination with what is represented on the game board, maximize the number of carry-overs (in the addition games) or borrowings (in the subtraction games). See screenshot in Fig. 1.



Why did you choose the card 39?

● It gives no stars, but is good anyway.
○ Because the red 9 + 1 will be packed and gives a star!
○ Because the orange 3 + 2 will be packed and gives a star!

Fig. 1. Screenshot of the math game, which depicts two competing TAs in an addition game. Here, "Mike's agent" poses a question as to why Mike picked a particular card.

The TA can be set in one of three different modes. In "Watch and learn"-mode, the TA successively learns the game rules, by "watching" the student's game moves and how the student responds to occasional questions, all in multiple-choice format. A typical question from the TA would be "Why did you pick this card?". The student is given a list of alternatives with only one correct option. In "Try and play"-mode, the TA suggests game cards, which the student can confirm by clicking "Ok" or refute by selecting another card. In "Play Self"-mode, students can watch their TA perform as it plays a session of the game against the computer, another TA or a human player.

3 Method

3.1 Participants

Forty-nine 4th-graders (9–10 years old), 19 girls and 30 boys, from two school classes in the same school, participated in the study. The two classes followed the same curriculum. The students were experienced in using laptops and were familiar with the question and answering formats (e.g. Likert scales) used in this study. Due to student absence and some computer mishaps when saving test data, only the results of 43 students could be used from the pretest, and of 47 students from the posttest.

3.2 Design and Instruments

Because there were no established instruments for the kind of manipulations we wanted to make, we needed to develop new tentative test materials. These included a digital pretest and a digital posttest, each of which appeared in a "TA version" and a "standard version" (that is, one test including the TA and the same test excluding the TA). The test questions, partly based on the Swedish national tests in mathematics, all targeted base-10 transformations (except one control question, which addressed multiplication). Examples include circling which sums end in "00" from six alternatives, or which sum is bigger of "236+342" and "432+127". Thus, the test questions corresponded conceptually to the content of the math game, but did not in form or detail resemble the TAs' questions and the students' multiple choice answers in the math game.

In total, the test comprised 41 items, each scored zero if incorrect and one if correct (theoretical score range 0–41). In order for the pretest and posttests not to be completely identical, two forms of the tests were created (form A and form B). These forms had only superficial differences (e.g. the item "27+13" in form A was replaced by "13+37" in form B). Students were randomly assigned form A or B as their pretest and the other form as their posttest. Exclusively for the TA versions of the posttest, the graphical TA was copied from the math game and placed in the margin of the screen. The TA's role was restricted to its visual, non-animated presence, including some introductory phrases (displaying e.g. "Hi, it's me – your agent – can you help me answering this questionnaire? I learn from you."). At two occasions during the test there was an opportunity for the student to click on the TA. When doing so, the TA was presented together with a similar – but not identical – task to the one the student had just answered. The TA then went through the sub-items (e.g. a–d) and answered them in an automated sequence. Importantly, the TA did not respond to the same items as the student. Furthermore, the TA was programmed so as to display as many correct answers as the student had done on the immediate previous task. In other words, the TA did not provide any help, support or clues as to whether answers were right or wrong, but always responded on the same level of accuracy as the student.

Attitudes and Experiences Questionnaires. Two pen-and-paper questionnaires were administered. One was in connection with the pretest and related to self-efficacy and motivation for math, and one was after the posttest, relating to one's experience of answering the posttest. This study focused on the posttest ratings. Four questions applied to all students (e.g., "How fun was the posttest?", "How much effort did you put into it?"). Students doing the TA version of the posttest were given two additional questions, relating to their sense of being helped by the TA and wanting to teach their TA, respectively. All answers were rated on a 0–10 Likert scale, where 0 represented the negative end ("not at all fun", "no effort", "very disturbing", etc) and 10 the positive end ("very much fun", "very much effort", "very helpful", etc).

3.3 Procedures

In the *pretest phase*, all 49 students did the pretest on one day, on individual laptops. Two experimenters led each testing session. The students filled out the pretest attitude questionnaire and were then presented with a practice test, in order to familiarize themselves with the test format (e.g., how to click and scroll through questions). After five minutes' practice, students did the proper pretest (either form A or B). The students were not timed, but had about 25 minutes for the test; almost all finished within this limit. The students were not given any feedback on their performance and were not informed that they were going to perform a similar test (the posttest) later.

In the *learning phase*, students participated in one session of 30 minutes per week over a period of eight weeks (including one week's intermission due to holidays). The sessions were semi-structured, such that new elements (the addition game, the TA and the subtraction game) were introduced by the experimenters in the beginning of each session, but the students were largely free to "practice what needed", with respect to training their TA in its weakest areas. The conditions during the learning phase, such as the location, group size, the number and duration of sessions, instructors (the researchers), and provided instructions, were equivalent for all students.

In the *posttest phase*, students followed a similar procedure to the pretest, but now doing the posttests (and without initial practice). By random assignment, half the students were given the TA version and half the standard version of the posttest. The two groups were matched on basis of their pretest scores, so there were as many students scoring above the median as below the median in each group. Finally, each student filled out an attitudes and experiences questionnaire.

4 Results

4.1 Summative Test Performance in Relation to the Presence of the TA

Our first research question was how the performance of students completing a regular summative test in the presence of their TA would relate to the performance of students doing the same test without the TA. Our primary analysis was concerned with the posttest results. On average, students (n = 23) who performed the TA version of the posttest scored higher (M = 30.0, SD = 5.5) than students (n = 24) performing the standard version of the same test (M = 26.5, SD = 8.9). However, an independent samples t-test comparing the two means was not significant; $t(45) = 1.572$, $p = .12$.

Upon closer analysis, we were interested in how students' posttest scores related to their baseline, in terms of their pretest scores. A linear regression analysis showed that adding the factor of posttest version (TA or standard) significantly improved the fit of the model; $F(1,40) = 4.82$, $p = .039$, compared to the baseline model of just pretest scores as the predictor of posttest scores. That is, how students were affected by the TA in the posttest apparently depended on their baseline level.

We therefore decided to compare the subgroups of "high-performing students" (*n* = 13), represented by the top quartile of pretest scorers (*M* = 33.0, *SD* = 1.9), to "low-performing students" (*n* = 12), represented by the bottom quartile of pretest scorers (*M* = 16.5, *SD* = 4.5). As seen in Fig. 2, high-performers hardly differed between test versions (*M* = 34.0, *SD* = 3.9 in the TA version; *M* = 33.7, *SD* = 6.7 in the standard version), nor did they improve much from their pretest. Low-performers, on the other hand, showed considerably improved scores on the TA version (*M* = 25.5, *SD* = 6.0), but slightly lower scores on the standard version of the posttest (*M* = 15.0, *SD* = 6.4).

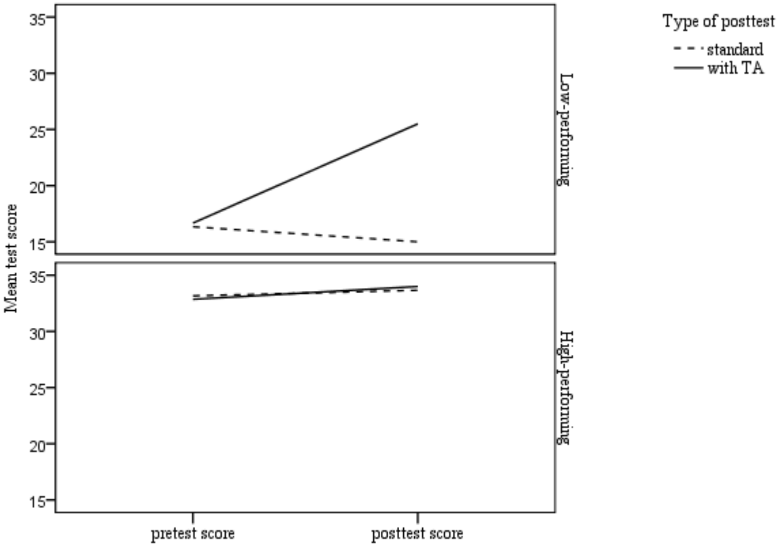


Fig. 2. Pre- and posttest mean scores for low-performing students (bottom 25% on pretest) and high-performing students (top 25% on pretest), with TA versus the standard posttest version.

Notably, all students were accompanied by a TA during the period of learning with the math game. It was only in the posttest that conditions differed between students, such that half the students again were accompanied by their TA (the TA version) whereas the other half were not accompanied by their TA (the standard version). Nevertheless, coincidence could have it that low-performers assigned to the TA version had learned more than low-performers assigned to the standard version, while using the math game in the learning phase. To control for this, we examined log data of the performance level of the students' TAs, and saw that students in the two conditions were comparable in this respect. Hence, the different results for the two groups of low-performing students did not seem due to varying success with training their TA in the learning phase, but to whether the TA was present or absent in the posttest.

4.2 Students' Subjective Experiences

Our second research question referred to how the TA would affect students' self-rated experiences of taking the posttest. In view of the posttest results, we chose to focus on the experience ratings by low- and high-performing students. See Table 1. The results showed some striking differences: Low-performing students' ratings of enjoyment were nearly twice as high for the TA version (*M* = 8.0, *SD* = 2.5) than for the standard version (*M* = 4.3, *SD* = 4.6). Another intriguing pattern is that low-performers with the standard version rated their confidence higher than their effort into doing the test, whereas low-performers with the TA version showed the reversed relationship.

Table 1. Mean ratings of experience items by low-performers and high-performers, on the TA and standard version posttests (on a 0–10 Likert scale, where 0 = very little, 10 = very much)

Experience item	Low-performers' rating (*SD*)		High-performers' rating (*SD*)	
	Standard I	TA version	Standard I	TA version
Enjoyment	4.3 (4.6)	8.0 (2.5)	7.7 (1.5)	6.8 (2.6)
Ease	6.2 (3.3)	6.5 (1.8)	8.0 (1.8)	8.2 (1.8)
Effort	5.3 (3.9)	7.6 (2.9)	7.1 (1.9)	7.0 (1.6)
Confidence	6.7 (3.0)	6.1 (1.3)	8.2 (1.8)	7.4 (1.8)

5 Discussion

This study set out to examine how the very presence of a TA would affect students' performance, when the TA from a math learning game recurred in a regular summative math test. The results showed that the effect of the TA's presence differed in relation to how students had performed on a math pretest. Low-performing students scored 70% higher on a posttest with the TA, compared to a standard posttest without the TA. For high-performing students, the presence of the TA seemed to make little difference. Questionnaire data also showed divergent patterns: Low-performers accompanied by a TA found the test considerably more enjoyable, and rated their own efforts into doing the test considerably higher, than did low-performers who completed the standard version. For high-performers there were no major differences on experience ratings between test versions.

How can these results be explained? One proposal is that the TA's presence in the TA version changes the student's mindset from that of "taking a test" to that of "teaching a TA" – even when, in fact, the test items in the TA version and the standard version are identical. The cognitive resources and support provided to the students for solving the tasks in the two conditions were also equal, since the TA did not add any feedback or clues to improve the students' problem-solving. Having an alternative mindset to that of "taking a test" is likely to benefit low-performers more than high-performers, since for low-performers, "taking a test" per definition is associated with non-success. Low-performers are therefore less likely than high-performers to enjoy tests and to make a large effort – just like the questionnaire data suggest.

Furthermore, with the TA present, students were clearly positioned as "teachers". This is likely to have affected self-efficacy beliefs (as reflected in confidence ratings) more for low-performing students, who typically lack previous experience of teaching someone. Self-efficacy beliefs, in turn, are well known to affect performance.

In sum, we believe that socio-motivational factors lie behind the results of the study. High-performers are already sufficiently motivated to make an effort when completing a test. Low-performers need a more supportive context in order to be motivated to accomplish and demonstrate what they have learned under the constraints of a conventional test. They may therefore particularly benefit from the TA as a form of social support. We hold as future research questions how assessment forms can be more effectively designed to benefit from social interaction in TA systems and related educational technologies.

Acknowledgments. This research was financed by the Wallenberg Foundation.

References

1. Biswas, G., Katzlberger, T., Brandford, J., Schwartz, D., TAG-V.: Extending intelligent learning environments with TAs to enhance learning. In: Moore, J.D., Redfield, C.L., Johnson, W.L. (eds.) Artificial Intelligence in Education, pp. 389–397. IOS Press, Amsterdam (2001)
2. Blair, K., Schwartz, D., Biswas, G., Leelawong, K.: Pedagogical agents for learning by teaching: Teachable Agents. Educational Technology Special Issue 47, 56–61 (2007)
3. Wagster, J., Tan, J., Wu, Y., Biswas, G., Schwartz, D.L.: Do learning by teaching environments with metacognitive support help students develop better learning behaviors? In: Proc. of the 29th Meeting of the Cognitive Science Society, Nashville, USA, pp. 695–700 (2007)
4. Chin, D., Dohmen, I., Cheng, B., Oppezzo, M., Chase, C., Schwartz, D.: Preparing students for future learning with TAs. Education Tech. Research Dev. 58, 649–669 (2010)
5. Chase, C., Chin, D., Oppezzo, M., Schwartz, D.: Teachable Agents and the Protégé Effect: Increasing the Effort Towards Learning. J. Sci. Educ. Technol. 18, 334–352 (2009)
6. Schwartz, D.L., Blair, K.P., Biswas, G., Leelawong, K., Davis, J.: Animations of thought: interactivity in the teachable agents paradigm. In: Lowe, R., Schnotz, W. (eds.) Learning with Animation, pp. 114–140. Cambridge University Press, Cambridge (2007)
7. Pareto, L., Schwartz, D., Svensson, L.: Learning by guiding a teachable agent to play an educational game. In: Proceeding of the International Conference on Artificial Intelligence in Education, pp. 662–664. IOS Press, Amsterdam (2009)
8. Pareto, L., Haake, M., Lindström, P., Sjödén, B., Gulz, A.: A Teachable Agent Based Game Affording Collaboration and Competition (under revision)
9. Lindström, P., Gulz, A., Haake, M., Sjödén, B.: Matching and mismatching between the pedagogical design principles of a math game and the actual practices of play. Journal of Computer Assisted Learning 27, 90–102 (2011)

Experimental Evaluation of Automatic Hint Generation for a Logic Tutor

John C. Stamper[1], Michael Eagle[2], Tiffany Barnes[2], and Marvin Croy[3]

[1] Human-Computer Institute, Carnegie Mellon University
john@stamper.org
[2] Department of Computer Science, University of North Carolina at Charlotte
{maikuusa,tiffany.barnes}@gmail.com
[3] Department of Philosophy, University of North Carolina at Charlotte
mjcroy@uncc.edu

Abstract. In our prior work we showed it was feasible to augment a logic tutor with a data-driven Hint Factory that uses data to automatically generate context-specific hints for an existing computer aided instructional tool. Here we investigate the impact of automatically generated hints on educational outcomes in a robust experiment that shows that hints help students persist in deductive logic courses. Three instructors taught two semester-long courses, each teaching one semester using a logic tutor with hints, and one semester using the tutor without hints, controlling for the impact of different instructors on course outcomes. Our results show that students in the courses using a logic tutor augmented with automatically generated hints attempted and completed significantly more logic proof problems, were less likely to abandon the tutor, and performed significantly better on a post-test implemented within the tutor.

Keywords: data mining, machine learning, logic tutor.

1 Introduction

In our previous work, we added the Hint Factory, an automatic hint generator, to the Deep Thought logic proofs tutor to automatically deliver context specific hints to students solving logic proofs, demonstrated its feasibility on historical data [2], and ran a pilot study to ensure hints were delivered correctly and appropriately [3]. We now evaluate its impact on educational outcomes, by adding hints to eight of eleven logic proof problems and examining their use in six deductive logic classes across two semesters. To control for the impact of different instructors, each of the three college philosophy instructors taught one semester of Deductive Logic with the Deep Thought tutor augmented with our data-derived hints, and each taught a semester using the tutor without hints. We hypothesized that providing our context specific hints automatically generated from data would improve students' ability to solve the given proof problems, and that having these hints available while students are solving practice problems would improve overall learning of the material.

We tested our first hypothesis on the impact of hints by examining the attempt and completion rates between students with hints and those without on three levels of

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 345–352, 2011.

problems. We tested the second hypothesis on overall learning by testing the learning on two post-test problems where no hints were available for any students. The results show that students in the hint group attempt and complete significantly more problems than students in the no-hint group. Further, students who were given hints on early problems outperformed students without hints on the post-test.

2 Background and Related Work

Marking student work as right or wrong is a simple form of feedback that can often be automated, but automatically generating effective formative feedback is a much more complex problem. Shute's review of the literature suggests that effective formative feedback be multidimensional and credible, specific but not evaluative, and infrequent but timely [14]. Determining the timing and frequency of hints is a particular challenge, but studies suggest that offering hints on demand, instead of proactively, can have positive effects on learning [11]. While some studies have suggested as much as 72% of help-seeking behaviors can be unproductive [1], Shih's work suggests that some of these behaviors are in fact helpful [13]. Shih argues that using help to achieve a bottom-out hint can be seen as looking for a worked example, an effective learning strategy [13].

Based on the hint and help literature, we devised a strategy of automatically generating hints to be as specific as possible, derived on-demand, and directed to the student's problem-solving goal, to provide the right type of help at the right time. Based on our experience in teaching logic for many years, we have observed that students often know how to execute the steps needed to solve logic proof problems but may have trouble choosing what to do next. These observations confirm that our on-demand, context-specific system could address the needs of students solving logic proof problems, but the research in our current study was needed to evaluate whether our implemented system achieved that goal.

Historically, the research and development of intelligent tutoring systems (ITS) have relied on subject area experts to provide the background knowledge to give hints and feedback. Two classes of effective tutors, cognitive tutors and constraint based tutors, rely on "rules" that experts create in a time-intensive process [8]. While this expertise and time are limited, the amount of data being collected from computer aided instruction continues to grow at an exponential rate. Data-driven methods applied to large data repositories like the PSLC DataShop [7] can enable the rapid creation of new intelligent tutoring systems, making them accessible for many more students.

As with RomanTutor, an ITS that uses sequential pattern mining over collected data to recommend actions to astronauts learning to operate a robot arm [10], we provide direct, data-driven feedback in an environment where students can choose from a large space of actions to perform and many are correct. We construct Markov Decision Processes (MDPs) that represent all student approaches to a particular problem, and use these MDPs directly to generate hints with the Hint Factory [3]. Barnes and Stamper demonstrated the feasibility of this approach on historical data, showing that extracted MDPs with our proposed hint-generating functions could provide hints over 80% of the time [2]. Fossati and colleagues have used our MDP method in the iList tutor used to teach linked lists and deliver "proactive feedback" based on previous student attempts [6]. In a pilot study, we augmented Deep Thought with the Hint Factory and showed that students were able to solve more logic proof problems when hints were included [3]. Almost the opposite of Bootstrap Novice Device (BND), which bootstraps example-based tutors with student data [9], we create a data-driven tutor that can be bootstrapped with expert solutions [12], providing at least some automatically generated hints initially and improving as additional student problem attempts are added to the model.

3 Hint Factory and Deep Thought Tutor

The Hint Factory consists of the MDP generator and the hint provider. The MDP generator is created through an offline process that assigns values to states in student problem attempt data. The hint provider uses these values to select the next "best" state at any point in the problem space.

The MDP Generator uses historical student data to generate a Markov Decision Process (MDP) that represents a student model, containing all previously seen problem states and student actions. A Markov decision process (MDP) is defined by its state set S, action set A, transition probabilities T, and a reward function R [15]. 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, representing 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 our experiments, 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 [15]. 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*.

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

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 [3]. The Hint Factory uses these values when a student is in a particular state to choose the next “best” state from which to generate a hint. When the hint button is pressed, the hint provider searches for the current state in the MDP and checks that a successor state exists. If it does, the successor state with the highest value is used to generate a hint sequence.

We have augmented Deep Thought, a custom online tool implemented as a Java applet, whose graphical interface allows students to visually connect premises and apply logic rules to solve logic proof problems [5], with the Hint Factory. For students with hints, a hint button appears, as shown at the lower right in Figure 1, when a student loads a problem. The button is bright yellow to make it more visible. When a new problem with hints is selected, the hint provider loads the entire hint file into memory. The Hint Factory for Deep Thought generates four types of hints: 1) indicate a goal expression to derive, 2) indicate the rule to apply next, 3) indicate the premises where the rule can be used, and 4) a bottom-out hint combining 1-3 [see 3 for more details].

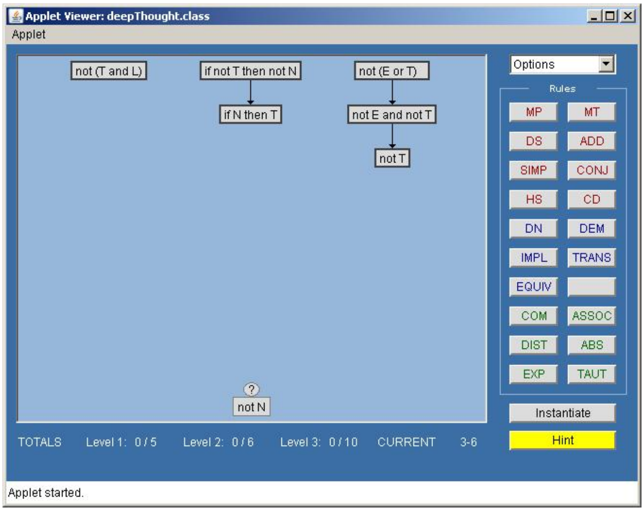


Fig. 1. The Deep Thought tutor showing a partially completed solution to problem 3-6. The student can select statements on the left side and apply rules from the buttons on the right. The added hint button in the lower right was only visible to students in the Hint group.

4 Experiment

Students from six different sections of a deductive logic course used the Deep Thought tutor. The sections included three sections in the Spring 2009 semester and three sections in the Fall 2009 semester. Each of three college philosophy professors taught one section each semester, one semester using Deep Thought with the Hint Factory and one semester using Deep Thought with no hints. This controlled for effects from different instructors by switching between the experimental and control conditions between the two semesters. Students in the Hint group classes could receive unlimited hints on the eight problems that had them and the Control group received no hints throughout. Students generally completed the problems, in order, over the course of the semester, but could access the problems at any time. Students accessed the Deep Thought tutor via the Moodle learning management system used to administer the course. All three classes used the same learning management system and were assigned the same problems. In the first semester there were 82 students in the Hint group and 37 students in the Control group; in the second semester there were 39 students in the Hint group and 83 in the Control group. Students with no log-data were dropped from the study; resulting in 68 and 37 students in the Hint group, and 28 and 70 students in the Control group for the spring and fall semesters respectively. This results in a total of 105 students in the Hint group and 98 students in the Control group.

Students from the six classes were assigned 13 logic proofs in the Deep Thought tutor. We have organized these problems into three constructs: level one (L1) consisting of the first 6 problems assigned, which use only inference rules; level two (L2) consisting of 5 problems using replacement and inference rules; and the post test (L3) consisting of the last two problems assigned. We use L3 as a post test measure since there were no hints for these two problems for either group.

5 Results and Discussion

We tested Hypothesis 1 by measuring the average percentage of completed problems for each of the three levels for each group, as shown in Table 1. Students were given credit if and only if they found a solution to the problem. To investigate the differences in performance between the groups, we submitted the results of L1, L2, and L3 to between-subjects two-tailed tests with an alpha of .05. The Hint group performed significantly better on all three levels, as shown in Table 2.

Table 1. Percent problem completion rates for each level. Students in the Hint group completed significantly more problems in each of the 3 levels. The L3 level had no hints for either condition and also acted as a post test measure.

Group	N	L1 Mean*	L1 SD	L2 Mean*	L2 SD	L3 Mean*	L3 SD
Hint	105	78.17	31.17	67.4	37.8	59.0	45.0
Control	98	61.17	36.17	42.4	40.8	41.5	46.0

Next we examined the effects on student motivation by comparing the number of problems attempted for each group. Table 3 shows the average percent of problems attempted and standard deviation for each of the three levels. Each of these percentages is from a total of six, five, and two problems in L1, L2, and L3 respectively. Students were given credit if they attempted to solve the problem, even if they did not find a solution (logs were flagged as solved when students solved a problem but data was collected even if a solution was not found). To investigate the differences in student motivation we submitted the attempt rates for L1, L2, and L3 to between-subjects two-tailed tests with an alpha of .05. There was no significant difference between the attempt rates for L1 with *d* = .27. The Hint group attempted significantly more L2 and L3 problems, with *t* (186.94) = -3.07, *p* =.002, *d* = .44, for L2, and *t* (195.33) = -2.32, *p* =.021, *d* = .33 for L3. This suggests that students begin the first level with the same motivation, but as the class continues, having hints available keeps students engaged so they attempt more problems in the later levels.

Table 2. Two-tailed t-test results comparing performance between the Hint and Control groups. All results are significant.

Level	t-test results	P level	Effect size
L1	t (191.82) = -3.58	p <.001	d = .51
L2	t (196.71) = -4.52	p <.001	d = .64
L3	t (201) = -2.78	p =.006	d = .39

Table 3. Percent problem attempt rates for each level. Students in both group attempt roughly the same number in L1, but the Hint group attempts significantly more problems in L2 and L3.

Group	N	L1 Mean	L1 SD	L2 Mean*	L2 SD	L3 Mean*	L3 SD
Hint	105	83.17	27.17	73.2	35.0	71.5	41.5
Control	98	75.33	31.00	56.0	43.2	57.0	46.0

To further test the effects of hints on student performance and persistence we looked at the overall rate in which students abandoned the Deep Thought tutor after the first level (L1) as seen in Table 4. Twenty-eight percent of the Control group abandoned the tutor after L1 (classified as Dropped below), while only 10% of the Hint group stopped attempting Deep Thought problems. A chi-square test of the relationship between group (Hint, Control) and Dropout (Continued, Dropped) produced $\chi^2(1) = 11.05$, which is statistically significant at *p* = .001. This is associated with an odds ratio of 3.62, indicating that the odds of dropping after the first level are more than 3.6 times higher when the students are not provided hints. This is a meaningful difference that suggests that online computer-aided instruction tools could benefit greatly from being augmented with automatically generated hints using the Hint Factory.

These results suggest that automatically generated hints keep students engaged and motivated to continue through the tutor. The mechanism for this effect may lie in the ability of students who are frustrated to ask for more help in Deep Thought plus the Hint Factory, while students in the Control group had no such alternative. Interestingly, we have observed students using the Hint Button for more than just asking for hints. The Hint Button is visible whenever a problem has hints, and is enabled when a

Table 4. Number of students that continued or dropped out of the tutor after L1

Group	Continued	Dropped	Total
Hint	95	10	105
Control	71	27	98

specific hint is available (which is about 80% of the time). When the button is disabled, its label is grey and clicking on it has no effect. When students perform a step that does not already exist in our MDP, the Hint Button is disabled, and some students have observed this. After several steps, if the student gets stuck the student may want a hint but know that none are available. We have observed that, rather than give up, students will delete their steps until the Hint Button becomes available again. In cases where steps do not exist in our MDPs, it is often the case that the student has tried something correct, but unusual, that may not lead to a solution. We believe that some students may have learned this through experience with the Hint Button and are thus receiving some tacit hints that particular steps in their work are unusual or unorthodox. Since students often generate several steps when they don't know a good strategy for moving ahead in a logic proof problem, this tacit help, while not evaluative, may be adding up to gentle nudges in the right direction. This can be thought of like seeing a subtle worn path while walking in the forest – you can go another way, but some who've gone before have taken the path.

6 Conclusions and Future Work

The main contribution of this work was to show that adding automatically generated hints to Deep Thought increased the attempt and completion rate for students solving logic proofs, independent of instructor or semester. Our results showed that the Hint group had significantly higher completion rates for all three levels of problems. We also showed that, while the two groups were equally motivated to attempt solving problems in level L1, the Hint group attempted significantly more problems in levels L2 and L3. Furthermore, students without hints were 3.6 times more likely to quit using the tutor ***altogether*** after the first level of problems.

When we consider L3 as a post-test measure that both groups completed without hints after levels L1 and L2, we see an overall learning effect for students in the Hint group, who were able to complete significantly more problems. This means that having hints early on helps students later when hints are not available, and suggests that having hints improves overall learning of logic proof solving.

In our future work, we plan to further analyze data for these six courses to more fully understand how students used hints. For example, we have collected but were unable to analyze pre and post test data for this research because it includes data for other topics in deductive logic. We plan to partition out the questions related to logic proofs and use these to group students by ability to investigate the hint usage patterns for each ability level. We are also inspired by Beck's excellent work on when to provide help based on his modification of Bayesian knowledge tracing [4]. The MDP method and Hint Factory serve as a model tracer that provide hints for step-based problem solving but does not use any information about student knowledge because that is not explicitly modeled. In the future we hope to compare methods for modeling knowledge in the logic domain using Bayes Nets and comparing these with methods of creating MDPs tailored to students at different knowledge levels. We also plan to extend our methods to create effective data-driven intelligent tutors for other step-based problem solving domains.

References

1. Aleven, V., McLaren, B.M., Roll, I., Koedinger, K.R.: Toward tutoring help seeking: Applying cognitive modeling to meta-cognitive skills. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 227–239. Springer, Heidelberg (2004)
2. 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)
3. 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.) Educational Data Mining (EDM 2008), Montreal, Canada, pp. 197–201 (2008)
4. Beck, J.E., Chang, K.-m., Mostow, J., Corbett, A.T.: Does help help? Introducing the bayesian evaluation and assessment methodology. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 383–394. Springer, Heidelberg (2008)
5. Croy, M., Barnes, T., Stamper, J.: Towards an Intelligent Tutoring System for propositional proof construction. In: Brey, P., Briggle, 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, pp. 186–195. IOS Press, Brighton (2009)
7. Koedinger, K.R., Baker, R.S.J.d., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the EDM commuity: 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 (2010)
8. Mitrovic, A., Koedinger, K., Martin, B.: A comparative analysis of cognitive tutoring and constraint-based modeling. In: User Modeling, pp. 313–322 (2003)
9. 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 (IT 2004), Maceió, Brazil (2004)
10. 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)
11. Razzaq, L., Heffernan, N.T.: Hints: Is It Better to Give or Wait to Be Asked? In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 349–358. Springer, Heidelberg (2010)
12. Stamper, J., Barnes, T., Croy, M.: Enhancing the Automatic Generation of Hints with Expert Seeding. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 31–40. Springer, Heidelberg (2010)
13. Shih, B., Koedinger, K.R., Scheines, R.: A Response Time Model For Bottom-Out Hints as Worked Examples. In: Educational Data Mining 2008, pp. 117–126 (2008)
14. Shute, V.J.: Focus on formative feedback. Review of Educational Research 78(1), 153–189 (2008)
15. Sutton, R., Barto, A.: Reinforcement Learning: An Introduction. The MIT Press, Cambridge (1998)

Human-Machine Student Model Discovery and Improvement Using DataShop

John C. Stamper and Kenneth R. Koedinger

Human-Computer Interaction Institute, Carnegie Mellon University

Abstract. We show how data visualization and modeling tools can be used with human input to improve student models. We present strategies for discovering potential flaws in existing student models and use them to identify improvements in a Geometry model. A key discovery was that the student model should distinguish problem steps requiring problem decomposition planning and execution from problem steps requiring just execution of problem decomposition plans. This change to the student model better fits student data not only in the original data set, but also in two other data sets from different sets of students. We also show how such student model changes can be used to modify a tutoring system, not only in terms of the usual student model effects on the tutor's problem selection, but also in driving the creation of new problems and hint messages.

Keywords: data mining, machine learning, cognitive modeling.

1 Introduction

Student models drive many of the instructional decisions that automated tutoring systems make, whether it is what instructional messages to provide and when, how to sequence topics and problems in a curriculum, how to adapt pacing to student needs, and even what problems and instructional materials are needed. Better student models yield better instruction. And student models can be improved by mining student interaction data. A better student model is one that better matches student behavior patterns. A better student model, in this empirical sense, is one that better predicts task difficulty and transfer of learning between related problems (during and after tutoring). We present a method for guiding the application of data mining algorithms for discovery of better student models. We show how data analysis tools such as those in the PSLC DataShop [6] can be used to identify areas for improvement, and then discuss how to quantitatively evaluate the new models.

Student models have traditionally been developed by domain experts engaging in manual analysis of course content. Cognitive Task Analysis (CTA) is an approach to understanding domain learning that has resulted in the design of significantly better instruction [2][9]. But CTA methods, like structured interviews, think aloud protocols, and rational analysis, have limitations. They are highly subjective and different analysts may produce different results. They also demand substantial human effort at each stage in data collection, analysis, and modeling. Automated techniques applied to large sets of student data can provide both more objectivity and reduce human effort.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 353–360, 2011.

Learning Factors Analysis (LFA) is an automated search technique for discovering student models [1]. Compared to some prior student model discovery approaches [10][12], LFA can apply to learning data not just performance data, and it is arguably more interpretable because of the use of human-provided labels (the "P matrix" in LFA). A key limitation, however, is that models can only be discovered within the space of the human-provided factors. If a better model exists but requires a factor that is not in the given set of input factors into LFA, it will not discover that model. How can such factors be discovered? We address this key question below. Applying automated methods is becoming more practical as the amount of student data continues to grow. Cognitive Tutors for mathematics are now in use for more than 500,000 students per year in the USA. While these systems have been quite successful, our analysis of log data suggests that the models behind them can be improved. Repositories to store large datasets from diverse educational domains can provide a central point to make these improvements. DataShop is such a repository that also includes a set of associated visualization and analysis tools (http://learlab.org/datashop). Student actions are coded as correct or incorrect and categorized in terms of the hypothesized competencies or "knowledge components" (KCs) needed to perform that action. A KC is a generalization of any element of a knowledge representation including a production rule, schema, or constraint. Each step the student performs that is related to a KC is recorded as an "opportunity" for the student to show mastery of that KC. (A step is a set of correct, incorrect, or help request actions related to the same problem subgoal [11].) Visualizations and analysis tools in DataShop are designed to help model builders find potential flaws in an existing student model and discover new KCs that yield a better fit to student data.

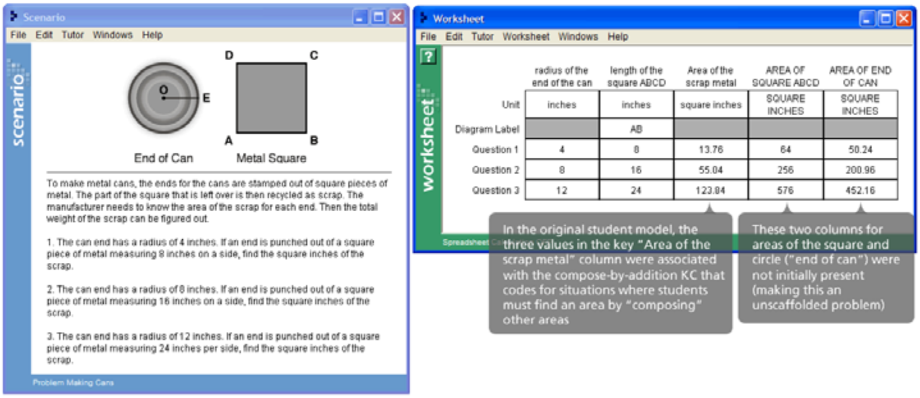
2 Human-Machine Student Model Discovery

Our Human-Machine Student Model Discovery method uses human input to identify model improvements from visualizations created from student log data and then evaluated by a statistical fit with the data. We use the Additive Factor Model (AFM), a statistical algorithm for modeling learning and performance that uses logistical regression performed over the "error rate" learning curve data [1]. AFM is a specific instance of logistic regression, with student-success (0 or 1) as the dependent variable and with independent variable terms for each student, each KC, and the KC by opportunity interaction. It is a generalization of the log-linear test model [13] produced by adding the KC by opportunity terms. Model discovery with AFM finds a set of KCs that best fits student data (without over-fitting). We chose AFM over other more complex alternatives (e.g., models with more terms in the logistic regression) because its simplicity enhances interpretation of the model parameters.

We describe three strategies for discovering opportunities for student model improvement that are supported by the visualization and analysis tools in DataShop:

1) *Smooth learning curves* - We expect that the learning curve for each KC will be reasonably smooth. When the learning curve of a purported KC is noisy, with upward or downward "blips", the student model is suspect.
2) *No apparent learning* - If the student model is accurate, we expect the error rate to decline over the number of opportunities a student has to learn and apply a KC. A flat learning curve is another indication of a potentially flawed student model.

3) *Problem steps with unexpected error rates* – A KC is suspect if the problem steps it labels have an error rate that is much higher or lower than the expected. In the ideal student model, the expected error rate for all steps labeled by the same KC should be about the same (albeit, the error rate should decline as students have more opportunities to practice). More precisely, the expected error rate is computed from the AFM statistical model that is built into the DataShop and the Performance Profiler tool provides a way to visualize whether any steps have error rates that are discrepant with this expectation.



To make metal cans, the ends for the cans are stamped out of square pieces of metal. The part of the square that is left over is then recycled as scrap. The manufacturer needs to know the area of the scrap for each end. Then the total weight of the scrap can be figured out.

1. The can end has a radius of 4 inches. If an end is punched out of a square piece of metal measuring 8 inches on a side, find the square inches of the scrap.

2. The can end has a radius of 8 inches. If an end is punched out of a square piece of metal measuring 16 inches on a side, find the square inches of the scrap.

3. The can end has a radius of 12 inches. If an end is punched out of a square piece of metal measuring 24 inches per side, find the square inches of the scrap.

	radius of the end of the can	length of the square ABCD	Area of the scrap metal	AREA OF SQUARE ABCD	AREA OF END OF CAN
Unit	inches	inches	square inches	SQUARE INCHES	SQUARE INCHES
Diagram Label		AB			
Question 1	4	8	13.76	64	50.24
Question 2	8	16	55.04	256	200.96
Question 3	12	24	123.84	576	452.16

In the original student model, the three values in the key "Area of the scrap metal" column were associated with the compose-by-addition KC that codes for situations where students must find an area by "composing" other areas

These two columns for areas of the square and circle ("end of can") were not initially present (making this an unscaffolded problem)

Fig. 1. A problem from the Geometry Cognitive Tutor. All cells values are filled by the student.

We focused on a publicly available data set from DataShop called "Geometry Area (1996-97)." This data was generated from student interactions with a cognitive tutor for learning Geometry, and a screen shot of from a newer version of the tutor can be seen in Figure 1. The data included 5,104 student steps completed by 59 students.

Smooth learning curves. The first discovery strategy was applied to this dataset by inspecting the learning curves of knowledge components (KCs) from the existing best student model, which was called Textbook-New. This model has 10 KCs. A subset of the learning curves for these KCs is shown in Figure 2. The lines represent the error rate (y-axis) averaged over all students for the first 20 practice opportunities for each KC (e.g., on the fifth opportunity on the trapezoid-area KC about 55% of students made an error). Defining exactly what constitutes a smooth learning curve is still an open research question, but most of the KCs have reasonably smooth learning curves, like compose-by-multiplication, parallelogram-area, and trapezoid-area. (Roughness in the learning curve can result from noise rather than a bad KC and particularly so when there a fewer observations being averaged is common at higher opportunity numbers.) The compose-by-addition curve is particularly jagged with upward blips in error rate. At opportunities 12 and 15-18, the curve jumps above from about 25% to about 50%. Assuming there are particular problem steps that are more likely to occur at these opportunities (which is the case in this data set), those steps appear to have some knowledge demand that the other steps do not. The compose-by-addition KC is involved in "composition problems", that is, problems where the area of an irregular shape (e.g., what's left when a circle is cut from a square) must be found by combining (adding or subtracting) the areas of the regular shapes that make it up (e.g., a square and circle). Identifying what makes these steps harder, may improve the student model.

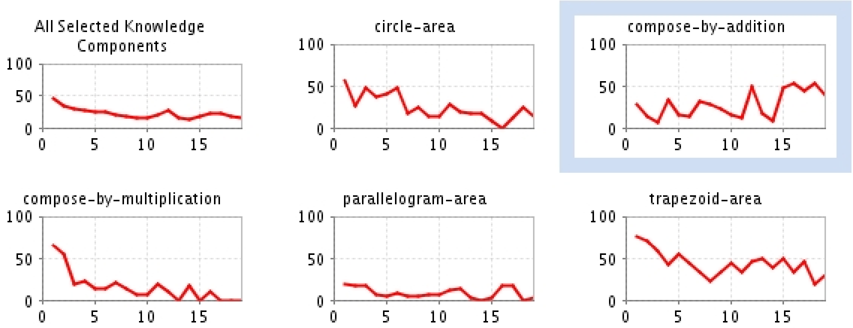
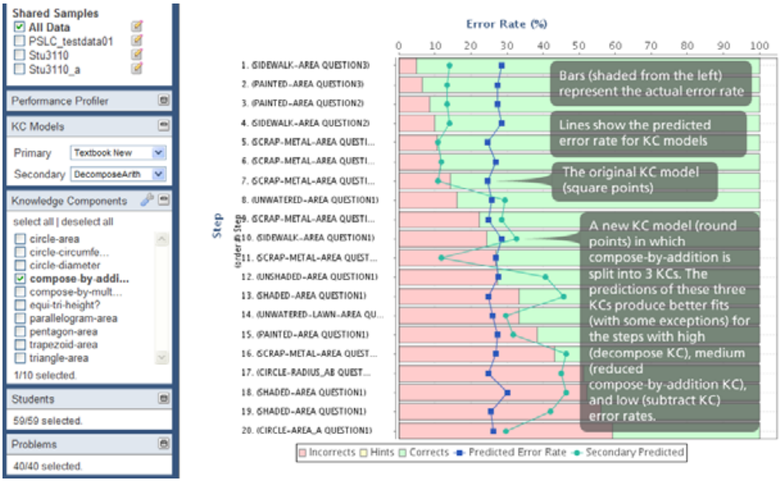


Fig. 2. Example learning curves from Textbook-New Student Model showing first 20 attempts for all students. Y-axis is the error rate and the X-axis is learning opportunities. Most of these curves are reasonably smooth and decreasing. "Compose-by-addition" is not smooth, with large jumps in the error rate at opportunities 12 and 15.

No apparent learning. The second discovery strategy is to identify KCs that do not indicate any student learning and are initially non-trivial. The parameter estimates for the KC terms in the AFM regression equation (introduced above) indicate the "intercept" or starting point of the learning curve and those for the KC by Opportunity interaction terms indicate the "slope" or rate of learning. Because AFM predicts success as the dependent variable (rather than error rate as shown in Fig 1), high values indicate a well-learned KC. To find KCs indicating little learning, DataShop's Learning Curve > AFM Values menu item provides parameter estimates. Looking at the slope column, we see little or no learning for compose-by-addition (0) and parallelogram-area (0.019). Parallelogram-area is of less concern because of the high intercept value, 89% correct (2.13 in log odds) at the first opportunity, indicating students mostly have this skill. That the compose-by-addition intercept is not high (74%) and the slope is absent indicates either that students are not learning or that the compose-by-addition is a poorly defined KC. We pursue the latter.

Problem steps with unexpected error rates. The third discovery strategy utilizes the Performance Profiler tool within DataShop (see Figure 3). Using this tool we confirmed that the error rates for problem steps coded by compose-by-addition are not well fit. As shown in Figure 3, there is a large discrepancy between easy steps at the top of the figure, with an error rate (indicated by the shaded bar) of about 5-10%, and hard steps at the bottom of the figure, with an error rate of about 40-60%. We also see that the predicted error rate from the Textbook-New model (the straighter line of connected points at about 30%) slices through the middle of these extremes and only rarely does an accurate job of predicting the error rates of any of these steps.

Not only does this strategy further implicate the compose-by-addition KC as a candidate for improvement, it also provides guidance for inspecting variations in the content of the problems to potentially identify new knowledge components. In particular, it suggests that we try to identify why some problem steps are harder than others and hypothesize what factor or knowledge-demand may make them harder? What we noticed is that some of the composition problems were "scaffolded" such that they included columns that cued students to find the component areas (square and circle) first [4]. Other problems were "unscaffolded" and did not start with such columns, thus students had to pose these subgoals themselves. Indeed the blips for compose-by-addition (seen in the learning curve in Figure 2) do correspond with a high frequency of these unscaffolded problems.



Shared Samples
☑ All Data
☐ PSLC_testdata01
☐ Stu3110
☐ Stu3110_a

Performance Profiler

KC Models

Primary: Textbook New
Secondary: DecomposeArth

Knowledge Components

select all | deselect all

☐ circle-area
☐ circle-circumfe...
☐ circle-diameter
☑ compose-by-addi...
☐ compose-by-mult...
☐ equi-tri-height?
☐ parallelogram-area
☐ pentagon-area
☐ trapezoid-area
☐ triangle-area

1/10 selected.

Students

59/59 selected.

Problems

40/40 selected.

Error Rate (%)
0
10
20
30
40
50
60
70
80
90
100
Step
Order in Step
1. (SIDEWALK-AREA QUESTION3)
2. (PAINTED-AREA QUESTION3)
3. (PAINTED-AREA QUESTION2)
4. (SIDEWALK-AREA QUESTION2)
5. (SCRAP-METAL-AREA QUESTI...
6. (SCRAP-METAL-AREA QUESTI...
7. (SCRAP-METAL-AREA QUESTI...
8. (UNWATERED-AREA QUESTION1)
9. (SCRAP-METAL-AREA QUESTI...
10. (SIDEWALK-AREA QUESTION1)
11. (SCRAP-METAL-AREA QUEST...
12. (UNSHADED-AREA QUESTION1)
13. (SHADED-AREA QUESTION1)
14. (UNWATERED-LAWN-AREA QU...
15. (PAINTED-AREA QUESTION1)
16. (SCRAP-METAL-AREA QUEST...
17. (CIRCLE-RADIUS_AB QUEST...
18. (SHADED-AREA QUESTION1)
19. (SHADED-AREA QUESTION1)
20. (CIRCLE-AREA_A QUESTION1)
🟥 Incorrects 🟨 Hints 🟩 Corrects ■ Predicted Error Rate ● Secondary Predicted

Bars (shaded from the left) represent the actual error rate

Lines show the predicted error rate for KC models

The original KC model (square points)

A new KC model (round points) in which compose-by-addition is split into 3 KCs. The predictions of these three KCs produce better fits (with some exceptions) for the steps with high (decompose KC), medium (reduced compose-by-addition KC), and low (subtract KC) error rates.

Fig. 3. DataShop's Performance Profiler shows the error rate on the steps for compose-by-addition KC

Based on the analysis, compose-by-addition was not at a fine enough level to accurately explain the student data, suggesting additional KCs may be present. To improve the model, compose-by-addition was split into 3 KCs, one representing the current compose-by-addition with scaffolding present, another where the student had to *decompose* the an irregular area without scaffolding, and a third where the student needs to *subtract* to execute the decomposition plan. For example, while all three cells in column 3 of Figure 1 (with values 13.76, 55.04, and 123.94) were originally coded as compose-by-addition, in the new model the first (where 13.76 was entered) is coded as *decompose* and the next two (55.04 and 123.94) are coded as *subtract*. After these KCs were hypothesized, the Textbook-New student model was exported from DataShop and modified in Excel. Of the 20 steps that were previously labeled with the compose-by-addition KC, 6 were labeled with the new decompose KC and 8 were labeled with the subtract KC. The model was imported back as "DecomposeArith".

For model evaluation we use Bayesian information criterion (BIC) and root mean-squared error (RMSE) from a 3 fold cross-validation where the folds are computed with the constraint that each of the 3 training sets must have data points for each student and KC (See [1] for justification of use of the BIC metric). The fit metrics for this new model are lower (BIC 5,628 and cross validation RMSE of 0.4021), thus better, than those for the former model (5,677 and 0.4064) indicating that DecomposeArith student model improves on the previous model without over-fitting. The cross-validation results are from a 3 fold cross-validation

What did we learn toward an improved student model? Most importantly, we found that it is possible to distinguish, in geometry tutor data, the difference between the process of "planning" a problem decomposition (in an unscaffolded problem) and the "execution" of one (in a scaffolded problem).The planning process requires figuring out how the area of an irregular shape (e.g., Figure 1) can be found from the areas of regular shapes that make it up (e.g., the square and circle). In the context of this tutor, the execution involves seeing that available regular area values (present in provided columns) can be used to find the irregular area and performing the required arithmetic. The planning involved in Geometry decomposition may be reflective of more general student competence at problem decomposition and such problems are quite common in standardized mathematics tests. The intercept and slope estimates for the decompose KC in the new model indicate both that it is difficult (the initial success rate is only 40%) for students and that the tutor is helping. The slope parameter is 0.15 logits per opportunity, which is one of the higher learning rates in this unit. Most students, however, finished the tutor far from mastery of decomposition. Because the original student model confounded decomposition planning and execution, it over-estimated student progress on decomposition - in essence giving students credit for decompose when they correctly performed simpler scaffolded composition and subtraction.

To confirm the model discovered above, we performed a parallel analysis on a second Geometry Area data set also available in DataShop called "Geometry Area Hampton 2005-2006 Unit 34." The original Textbook student model associated with this data set has 13 KCs and the metric values: BIC 15,375.1 and cross validation RMSE of .4078. Based on the previous findings, the Textbook KC model was modified with the steps for compose-by-addition split into 3 KCs as suggested above. When the additional KCs were added, the new model (DecomposeArith) with 15 KCs showed improvement (BIC 15,176.7 and cross validation RMSE of .4042) further validating the existence of the new KCs. By demonstrating the success of the model changes on a new data set, not used to discover the model, we greatly reduce the chance that this model is not idiosyncratic or over-fit to the first data set.

3 Using a Discovered Student Model to Redesign a Tutor

Once an improved student model has been discovered, different kinds of changes in a student model can suggest redesign moves in the tutor. Potential changes include:

1) Resequencing – put problems requiring fewer KCs before ones needing more
2) Knowledge tracing – add/delete skill bars for better cognitive mastery
3) Creating new tasks – add problems to focus practice on new KCs
4) Changing instructional messages, feedback or hint messages

We applied the discovered student model to the Geometry area unit of a pre algebra course. The critical difference between the discovered and existing models is the new KCs for planning problem decomposition. As implied above, problem selection and knowledge tracing of a tutoring system is affected by changes in the model. We added 3 new skills to the tutor that make the distinction between unscaffolded decomposition, scaffolded, and simple subtraction. Students in this new version will not be able to get credit for the difficult decomposition step through success on simpler scaffolded or subtraction steps. New problems to target these newly identified skills were added. We identified 3 types of problem dimensions that would help isolate the new skills in area compositions problems. These are table scaffolded, area scaffolded, and problem statement scaffolded. Table scaffolded problems reflect the current setup in the tutor and include columns for intermediate areas. Area scaffolded problems go a step further and give the areas of the component shapes. Problem statement scaffolded problems provide less support in that the component area columns are not present, but there is an explicit hint in the problem statement directing the student to first find the areas of the individual shapes. Using these problem dimensions, four new problem types were created. In the new curriculum, there are more unscaffolded problems (and earlier in the curriculum), but and also problems that isolate just the decomposition step by giving students the component areas instead of requiring them to compute those areas. In general, changes in skills can lead to changes in the feedback and hint messages the tutor provides. The new problems give students focused instruction on these skills.

We performed a pilot *in vivo* experiment with the new student model and redesigned tutor in the Spring of 2010 with 5 classes working on the Carnegie Learning "Bridge to Algebra" cognitive tutor. 120 students were split into two conditions. The experimental condition received the new instruction driven by the new model while the control was given the original instruction with the old model.

Using the data collected, both the new student model with new skills and the old model without the new skills were fitted. The resulting model with 49 KCs had the following metrics: BIC 31,183.9 and cross validation RMSE of .3255. These were lower than the previous metric values of the model with 46 KCs (BIC 31,258.9 and cross validation RMSE of .3269) showing the model with the additional KCs labeled is better than the model without the new skills. The ultimate goal of the redesign was to show a better model leads to better learning. The experimental condition had better posttest means (M=.72, SD=.22) than the control (M=.64, SD=.20), but an ANCOVA analysis indicates only a marginal effect, F (1,78) = 3.47, p = .07, when accounting for pretest scores. Given the results are in the predicted direction, we would be justified to use a one-tailed test and reject the null hypothesis (p<.05). However, we intend future studies of this kind to get more solid evidence.

4 Conclusion and Future Work

We presented a human-machine discovery approach for improving student models by using DataShop tools. We demonstrated how this approach can produce non-trivial improvements in a student model, even in a domain (Geometry) where there has been considerable attention and prior cognitive analysis. We demonstrated how this new student model better fits student data, not only in the original data-set used to discover it, but also in two other data sets. We used the model to modify tutor instruction, particularly to create new problems that help isolate the difficult skill of planning problem decompositions. While not confirming the instructional benefits of this new student model, our initial experiment showed promise. At least one study has demonstrated that data-driven student model improvements can yield better instruction [7], but this work is novel in showing how aspects of student model improvement can be increasingly systematized and automated.

The approach we describe still has a significant human component. Unsupervised methods for model discovery [12][3] have the potential to produce better fitting models with less effort. However, it is not clear how to interpret the results of these models and apply them in improving tutor design. Further investigation is needed. More generally, using data to optimize student models and, in turn, improve instructional systems is a tremendous opportunity. The achievement are likely to be greater to the extent that the discovered models involve deep or integrative KCs not directly apparent in surface task structure, like the problem decomposition skill we identified in Geometry. This work was supported by the Pittsburgh Science of Learning Center (NSF award 0836012).

References

1. Cen, H., Koedinger, K.R., 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, pp. 164–175. Springer, Heidelberg (2006)
2. Clark, R.E., Feldon, D., van Merriënboer, J., Yates, K., Early, S.: Cognitive task analysis. In: Spector, J., Merrill, M., van Merriënboer, J., Driscoll, M. (eds.) Handbook of Research on Educational Communications and Technology, Mahwah, NJ, pp. 577–593 (2007)
3. Collins, M., Dasgupta, S., Schapire, R.: A generalization of PCA to the exponential family. In: Procs. of the 14th Conf. on Neural Info. Processing Systems, NIPS (2001)
4. Corbett, A.T., Anderson, J.R.: Knowledge tracing: Modeling the acquisition of Procedural knowledge. User Modeling and User-Adapted Interaction, 253–278 (1995)
5. Heffernan, N., Koedinger, K.: A developmental model for algebra symbolization: The results of a difficulty factors assessment. In: Gernsbacher, M.A., Derry, S.J. (eds.) Procs. of the 20th Annual Conf. of the Cognitive Science Society, Mahwah,NJ, pp. 484–489 (1998)
6. Koedinger, K.R., Baker, R.S.J.d., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the EDM commuity: The PSLC DataShop. In: Romero, V., Pechenizkiy, B. (eds.) Handbook of Educational Data Mining. CRC Press, Boca Raton (2010)
7. Koedinger, K., McLaughlin, E.: Seeing language learning inside the math: Cognitive analysis yields transfer. In: Procs. of the 32nd Ann. Conf. of the Cogitive Science Society (2010)
8. Koedinger, K.R., Nathan, M.J.: The real story behind story problems: Effects of representations on quantitative reasoning. The Jrnl of the Learning Sciences 13(2), 129–164 (2004)
9. Lee, R.L.: Cognitive task analysis: A meta-analysis of comparative studies. Unpublished doctoral dissertation, University of Southern California, Los Angeles, CA (2003)
10. Tatsuoka, K.K.: Rule space: An approach for dealing with misconceptions based on item response theory. Journal of Educational Measurement 20, 345–354 (1983)
11. VanLehn, K.: The behavior of tutoring systems. Intl Jrnl of AIED 16, 227–265 (2006)
12. Villano, M.: Probabilistic student models: Bayesian Belief Networks and Knowledge Space Theory. In: Procs. of the 2nd International Conference on ITS, pp. 491–498. Springer, Heidelberg (1992)
13. Wilson, M., de Boeck, P.: Descriptive and explanatory item response models. In: de Boeck, P., Wilson, M. (eds.) Explanatory Item Response Models, pp. 43–74. Springer, Heidelberg (2004)

Talk Like an Electrician: Student Dialogue Mimicking Behavior in an Intelligent Tutoring System

Natalie B. Steinhauser[1], Gwendolyn E. Campbell[1],
Leanne S. Taylor[2], Simon Caine[2], Charlie Scott[2],
Myroslava O. Dzikovska[3], and Johanna D. Moore[3,⋆]

[1] Naval Air Warfare Center Training Systems Division, Orlando, FL, USA
{natalie.steinhauser,gwendolyn.campbell}@navy.mil
[2] Kaegan Corporation, 12000 Research Parkway, Orlando, FL 32826-2944
{Leanne.Taylor.ctr,Simon.Caine.ctr}@navy.mil, CScott@kaegan.com
[3] School of Informatics, University of Edinburgh, Edinburgh, United Kingdom
{m.dzikovska,j.moore}@ed.ac.uk

Abstract. Students entering a new field must learn to speak the specialized language of that field. Previous research using automated measures of word overlap has found that students who modify their language to align more closely to a tutor's language show larger overall learning gains. We present an alternative approach that assesses syntactic as well as lexical alignment in a corpus of human-computer tutorial dialogue. We found distinctive patterns differentiating high and low achieving students. Our high achievers were most likely to mimic their own earlier statements and rarely made mistakes when mimicking the tutor. Low achievers were less likely to reuse their own successful sentence structures, and were more likely to make mistakes when trying to mimic the tutor. We argue that certain types of mimicking should be encouraged in tutorial dialogue systems, an important future research direction.

Keywords: Mimicking, Alignment, Intelligent Tutoring System (ITS), Human Computer Interaction (HCI).

1 Introduction

One component of learning a new domain is to learn the "language" of that domain. This includes not only the domain-specific vocabulary, but also the appropriate phraseology and knowledge of how to construct an argument or explanation in that domain. Being able to speak the language of a domain is necessary for effective communication with members of the relevant professional

[⋆] This work has been supported in part by US Office of Naval Research grants N000141010085 and N0001410WX20278. We would like to thank our sponsors from the Office of Naval Research, Dr. Susan Chipman and Dr. Ray Perez, and former Research Associates who worked on this project, Katherine Harrison, Elaine Farrow, and Charles Callaway for their contributions to this effort.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 361–368, 2011.

community. Students should begin to learn how to "talk like an electrician" (or doctor, or lawyer,etc.) in the classroom, by mimicking the teacher's or tutor's use of domain-specific language. In a computer tutoring context, it is even more important that the student copy the system's use of language, as none of the existing systems are able to understand a full range of natural language input.

It has long been observed that people modify their use of language to correspond more closely with the language used by the person or system that they are communicating with. This basic phenomenon has been studied in many contexts using a variety of different labels and definitions, in particular "alignment" [7], "convergence" [9], "lexical entrainment" [1], and "cohesion" [8].

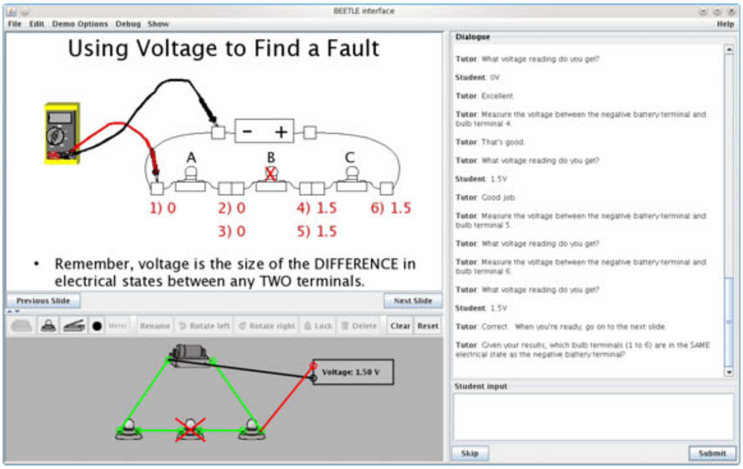
There is also evidence that the presence of this behavior in student dialogues is positively predictive of measures of student learning. Ward & Litman [8,9] defined lexical cohesion as the percentage of co-occurrence of individual words within consecutive pairs of dialogue turns, and lexical convergence as the rate of lexical change over a window of 5 to 50 turns. In their curriculum, for students with below average pre-test scores, higher lexical cohesion and convergence scores between the student and the tutor during the tutorial dialogue was predictive of a higher learning gain score. On the other hand, cohesion assessed on pairs of utterances made by the same speaker, whether it was the tutor or the student, was not correlated with learning gain. Ward & Litman concluded that a low level of convergence may indicate that the student is not aligning semantically with the tutor and therefore not learning.

To date, the majority of the research investigating the relationship between alignment and learning gain in computer-based tutoring environments has focused primarily on lexical alignment. Measures of lexical alignment are easy to compute automatically, and it has been theorized that alignment at one level leads to alignment at other levels [7]. In the current study, however, we attempt to extend the previous research by explicitly broadening our definition of linguistic overlap to incorporate features of both lexical and syntactical alignment. We hypothesize that this broader measure should be important in a training context because it reflects the extent to which students use the "language" of a new domain - i.e., not only repeating domain content words, but also organizing those words in meaningful and approriate sentences. In addition, whenever students align at both levels this is more likely to result in utterances that are easy to understand for current computer systems, give state of the art in Natural Language Processing (NLP). In the remainder of the paper we present our measure, which we call "mimicking," and describe our research testing our hypothesis that the amount of mimicking a student produces during a tutoring session will be positively correlated with their learning gain.

2 Method

2.1 Data Collection Environment

The Basic Electronics and Electricity Tutorial Learning Environment (BEETLE II)[6] was used for data collection. The BEETLE II curriculum of interest in



Using Voltage to Find a Fault

- Remember, voltage is the size of the DIFFERENCE in electrical states between any TWO terminals.

Dialogue

Tutor: What voltage reading do you get?

Student: 0V

Tutor: Excellent.

Tutor: Measure the voltage between the negative battery terminal and bulb terminal 4.

Tutor: That's good.

Tutor: What voltage reading do you get?

Student: 1.5V

Tutor: Good job.

Tutor: Measure the voltage between the negative battery terminal and bulb terminal 5.

Tutor: What voltage reading do you get?

Tutor: Measure the voltage between the negative battery terminal and bulb terminal 6.

Tutor: What voltage reading do you get?

Student: 1.5V

Tutor: Correct. When you're ready, go on to the next slide.

Tutor: Given your results, which bulb terminals (1 to 6) are in the SAME electrical state as the negative battery terminal?

Student input

Fig. 1. Screenshot of the Beetle II system

this study is a lesson on basic electricity and electronics that covers topics such as open and closed paths, voltage reading between components, and finding faults in a circuit with a multimeter. Students took approximately three hours to complete this lesson.

The screen of the Beetle II system can be seen in Figure 1. It contains lesson material in the form of a self-paced page-turning slide show, a circuit simulator which allowed the students to build and manipulate circuits as a complement to the lesson material, and a chat window where the participants and computer tutor interacted. All interactions with the tutor were typed.

2.2 Procedure

After reviewing the informed consent, participants filled out a demographic questionnaire and took a 22 question pre-test. The participants were then introduced to Beetle II and given a brief demonstration on the functionality of the learning environment. The students spent the majority of the experimental session working through the lesson materials. During the lesson, the computer tutor instructed the student to read slides, build circuits, and asked the student questions about the material. Every time the student responded to a question, the tutor would provide appropriate feedback.

When the student's answer was correct the tutor would reinforce the answer by either acknowledging that it was correct (e.g., "that's great") or by providing a better way to phrase the answer if the student was right, but not stating the answer in the ideal way (e.g., "Very good. Terminal 1 is connected to terminal 2."). We called the latter a "model better answer" strategy [4]. When the student answered incorrectly, the tutor responded with a a hint to help the student come up with the correct answer on their own. If the student could not get the answer after three increasingly detailed and specific remediations, the tutor would give the student the answer (e.g., “Almost. Here’s the answer. The positive battery terminal is separated by a gap from terminal 3.”). We called this a “bottom out”.

In about 13% of cases, the system was unable to interpret the student’s utterance. In those instances, the system produced an error message indicating that the student was not understood and the reason for misunderstanding, e.g., “I am sorry, I’m having trouble understanding. I didn’t understand the word ’power’ ” [3]. It then asked the student to rephrase their answer, providing a hint depending on the tutoring policy. We will refer to these errors as “uninterpretable utterances.” Similar to the case of multiple errors, the system used the “bottom out” strategy if the student made too many uninterpretable utterances.

After the students had completed the lesson, they took a 21 item post-test and filled out a satisfaction questionnaire.

3 Corpus Annotation

The original corpus was comprised of dialogues from forty-one participants. Previous research has shown that the relationship between lexical alignment and learning was strongest for the weakest students. Thus, for this preliminary investigation, we focused on the subset of the corpus that we believed to be most likely to demonstrate an effect of mimicking, the students at the extremes of our distribution. More specifically, we calculated a (normalized) gain score for each student as $\frac{(post-pre)}{1-pre}$. Next, we rank ordered our participants based on this gain score and selected the dialogues from the top ten and bottom nine students. A quick double-check confirmed that, as expected, the high gainers ($M = .75$, $SD = .04$) had a significantly higher learning gain score than the low gainers ($M = .40$, $SD = .17$), $t(17) = 8.67$, $p < .001$.

Of these 19 participants, nine were male and ten were female. Participants’ ages ranged from 18 to 25 years with an average age of 20. The final corpus included 770 student turns ($M = 40.5$ per student, $SD = 9.88$).

Our next step was to come up with an operational definition of mimicking that captured the majority of cases of both lexical and syntactical alignment within our corpus, and could be reliably coded by human raters. This is where we were able to rely upon a special feature of our curriculum, which is that many topics are addressed through a series of semi-repetitive questions. For example, in the exercise shown in Figure 1, the students are asked to measure voltage at 4 different points in a circuit. They are then asked a series of questions about the measurements they obtained: “Why did you get the voltage of 1.5 between terminal 1 and the positive battery terminal?”, “Why did you get the voltage of 0 between terminal 2 and the positive battery terminal?”, and so on. Once an acceptable answer to the first question has been established (either by the student or by the tutor), the student has an opportunity to “mimic” it, i.e., re-use that answer with minor changes for the following questions. For example, in the top left column in Table 1, the student gives a correct answer to the question. After the tutor acknowledges it as correct, the student uses exactly the same sentence to answer the next question, only modifying it to refer to terminal 6 instead of terminal 5.

Within this framework, we defined mimicking as re-using a complete previous answer, with two minor variations allowed: substituting the component being referenced (e.g., using “bulb A” instead of “bulb B”), and adding or removing negation (e.g., saying “not connected to” instead of “connected to”). Before beginning to code for mimicking, we identified 25 questions where the student has an opportunity to mimic the answer to a previous question. Three independent raters then coded the transcripts, coding each student answer to those questions as either (a) new, (b) a mimic of a previous statement made by the tutor, or (c) a mimic of a previous statement made by the student.[1] Two transcripts were coded by multiple coders to assess inter-rater reliability, which proved to be high (kappa = 0.88).

This way of defining mimicking as a re-use of a statement with only minor changes may seem stringent, but it works well within the context of our curriculum: it reflects strong lexical and syntactic alignment and can be unambiguously recognized by human raters. We return to this in Section 5.

As alluded to above, there are two potential sources of mimicking behavior. First, the students could mimic themselves, by repeating their own previous answers with minor modifications. We refer to this as “self-mimicking”. Second, the students can mimic the answers the tutor gives when either the “bottom-out” or “model better answer” strategy is used. We refer to this as “tutor-mimicking”.

4 Results

First, we tested our hypothesis that mimicking behavior is positively correlated with learning gains. The overall number of mimicked turns was not significantly correlated with learning gains, $r(18) = -0.17$. The number of self-mimics was also not significantly correlated with learning gains, $r(18) = 0.27$. The number of tutor-mimics was significantly negatively correlated with learning gains, $r(18) = -0.49$, $p < 0.05$.

Given that our results were in direct contradiction to findings from previous research, we extended our coding system to incorporate a feature that the raters had noted during the original coding process – specifically, the fact that sometimes the students' mimicking behavior produced correct answers to new questions, and sometimes it did not. The success or lack of success of each student turn coded as a mimic was determined by the computer tutor's response to that student turn. If the computer tutor accepted the student answer as correct, the mimic was categorized as successful. If the computer tutor was not able to interpret the student answer, or if the computer tutor determined that the student answer was missing essential information or contained one or more errors, then the mimic was categorized as unsuccessful. Examples of successful and unsuccessful, self-mimics and tutor-mimics can be found in Table 1.

[1] The Beetle II system was unable to change its phrasing, so only student turns were coded.

Table 1. Examples of successful and unsuccessful mimics

	Successful	Unsuccessful
Self-Mimic	**Tutor:** Explain why you got a voltage reading of 0 for terminal 5 and the positive terminal. **Student:** *terminal 5 is connected to the positive terminal of the battery.* **Tutor:** Very good. **Tutor:** Explain why you got a voltage reading of 0 for terminal 6 and the positive terminal. **Student:** *terminal 6 is connected to the positive terminal of the battery.* **Tutor:** Perfect.	**Student:** *There was a gap between terminal 2 and the positive terminal.* **Tutor:** Very good. **Tutor:** Explain why you got a voltage reading of 1.5 for terminal 3 and the positive terminal. **Student:** *There was no gap between terminal 3 and the positive terminal.* **Tutor:** Some of your answer is right, but some is wrong. Please try again.
Tutor-Mimic	**Tutor:** Here's the answer. *The positive battery terminal is separated by a gap from terminal 1.* **Tutor:** Explain why you got a voltage reading of 1.5 for terminal 2 and the positive terminal. **Student:** *The positive battery terminal is separated by a gap from terminal 2.* **Tutor:** That's great.	**Tutor:** Here's the answer. *The positive battery terminal is separated by a gap from terminal 1.* **Tutor:** Explain why you got a voltage reading of 0 for terminal 4 and the positive terminal. **Student:** *the positive battery terminal is separated by a gap from terminal 4.* **Tutor:** Some of your answer is right, but some is wrong. Please try again.

Once the transcripts were coded for mimicking success, the data were tabulated and summarized (see Figure 2). There was a significant difference between our high and low gainers in the percentage of self-mimicking they produced $t(17) = 2.17$, $p = .05$ and in the percentage of unsuccessful tutor-mimics $t(17) = -3.17$, $p = .01$.

Next, we investigated the relationship between mimicking and uninterpretable utterances. None of the existing dialogue systems are able to interpret the full range of human speech and we have previously shown that high frequency of uninterpretable utterances is negatively correlated with learning gain [5]. We found that overall percentage of mimicking was significantly negatively correlated with percentage of uninterpretables in dialogue ($r = -0.52$, $p = 0.02$), and this correlation was primarily explained by self-mimicking ($r = -0.46$, $p = 0.04$), while tutor-mimics were not significantly correlated with uninterpretables ($r = 0.04$, $p = 0.88$).

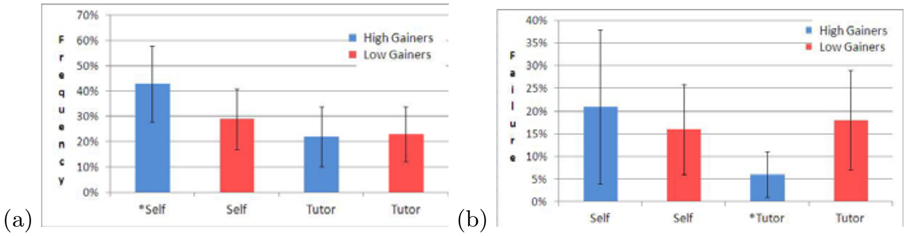


Fig. 2. Assessing Self and Tutor-mimicking between high and low gainers on (a) Frequency and (b) Failures

Finally, we looked at correlations between the reported overall satisfaction with the system and the amounts of various types of mimics. We found that self-mimicking was correlated with overall satisfaction with the system, $r(18) = 0.52$, $p < 0.05$. However, tutor-mimicking was not significantly correlated with overall system satisfaction, $r(18) = -0.36$, $p > 0.05$.

5 Conclusion

Past research, using a word-by-word method for assessing overlap in two language samples, has shown that the more a student's language converges towards the tutor's language, the higher the student's learning gains, especially among poorer students. In the current study, we moved to a new domain and learning task, and, more importantly, used a different measure of alignment (which we call mimicking), focusing on an amalgamation of lexical and syntactical alignment. We initially hypothesized, like past research, that student mimicking of the tutor would yield higher learning gains and improve communication with the system.

Our results produced a more complex pattern between the variables than was found in previous research. Students with the highest learning gains were more likely to mimic themselves and were more satisfied with the system. It appeared that these students found a strategy for responding to the tutor's questions that was successful and then stuck with it as much as possible. On the other hand, students with the lowest learning gains were less likely to mimic themselves, less able to successfully mimic the tutor and were less satisfied with the system. Moreover, for all students, the more they engaged in self-mimicking, the more successful they were in communicating with the tutor.

These results suggest that it may be advantageous to encourage certain types of mimicking behaviors in a tutorial setting and particularly with an ITS. Mimicking will help students to "talk like an electrician" (i.e., learn the proper way of speaking in the domain) and will help them to be understood by the system, which should make for better dialogue and a more enjoyable experience with the system. The current system incorporated features designed to facilitate tutor-mimicking. For example, the "bottom-out" and "model better answer" strategies were intended to provide patterns that students could imitate. Our results indicate, however, that self-mimicking is a more important predictor of learning gain. The best strategies to encourage self-mimicking are an open question for future work.

We used the special property of our curriculum, namely, the presence of semi-repetitive questions, to account for syntactic alignment without the need of syntactic parsing based on surface properties of student answers. Our definition was based on phenomena frequently observed in our corpus, and designed to achieve high inter-rater reliability. It would be possible to relax it slightly, in particular, to allow for use of pronouns and discourse connectives, and we are considering extending our analyses to cover those cases. The results also need to be replicated with different domains and curricula. However, in absence of similar questions, automated NLP tools would have to be used. Possibilities include using syntactic parsers or other automatically computable measures of cohesion (e.g. those used in Coh-Metrix [2]).

It is also possible that the importance of mimicking and the ease or difficulty of mimicking successfully may be affected by the quality of NLP and the nature of the domain. If the system error rate decreases, the percentage of successful mimics may increase and the relationship between successful mimicking and learning gain may change. Thus, these results should be re-examined as advances are made in the state of the art in natural language interpretation.

References

1. Brennan, S.E.: Lexical entrainment in spontaneous dialog. In: Proceedings of the 1996 International Symposium on Spoken Dialogue, pp. 41–44 (1996)
2. D'Mello, S.K., Dowell, N., Graesser, A.C.: Cohesion relationships in tutorial dialogue as predictors of affective states. In: Proceedings of AIED 2009, pp. 9–16 (2009)
3. Dzikovska, M.O., Callaway, C.B., Farrow, E., Moore, J.D., Steinhauser, N.B., Campbell, G.C.: Dealing with interpretation errors in tutorial dialogue. In: Proceedings of SIGDIAL-2009, London, UK (September 2009)
4. Dzikovska, M.O., Campbell, G.E., Callaway, C.B., Steinhauser, N.B., Farrow, E., Moore, J.D., Butler, L.A., Matheson, C.: Diagnosing natural language answers to support adaptive tutoring. In: Proceedings of the 21st International FLAIRS Conference (2008)
5. Dzikovska, M.O., Moore, J.D., Steinhauser, N., Campbell, G.: The impact of interpretation problems on tutorial dialogue. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (2010)
6. Dzikovska, M.O., Moore, J.D., Steinhauser, N., Campbell, G., Farrow, E., Callaway, C.B.: Beetle II: a system for tutoring and computational linguistics experimentation. In: Proceedings of the ACL-2010 Demo Session (2010)
7. Pickering, M.J., Garrod, S.: Toward a mechanistic psychology of dialogue. Behavior and Brain Sciences 27, 169–226 (2004)
8. Ward, A., Litman, D.: Cohesion and learning in a tutorial spoken dialog system. In: Proceedings of 19th International FLAIRS Conference (2006)
9. Ward, A., Litman, D.J.: Dialog convergence and learning. In: Proceedings of the 13th International Conference on Artificial Intelligence in Education (2007)

Dynamic Guidance for Task-Based Exploratory Learning

James M. Thomas and R. Michael Young

Digital Games Research Center
Department of Computer Science
North Carolina State University, Raleigh, NC USA
jmthoma5@ncsu.edu, young@csc.ncsu.edu

Abstract. This paper describes the implementation and evaluation of a new system to guide exploratory learning in arbitrary task-based domains. The system employs a knowledge representation borrowed from the field of automated planning to represent both the exploratory environment and the student's model of the tasks in the environment.

1 Introduction

This paper describes the implementation and evaluation of a new system to guide exploratory learning in arbitrary task-based domains. The ITS field has benefitted from a shared consensus of proven techniques for intelligent scaffolding [13,5], but common techniques to solve the unique challenges of exploratory or inquiry-based tutoring have proven more elusive [2,6,12].

Exploratory environments provide students with freedom to choose different courses of action. This complicates the tutor's ability to know what the student it trying to do, which introduces uncertainty in knowing whether or not a student has a misconception about the domain. When the tutor decides a misconception exists, it is difficult to know when is the right time to provide support to remediate that misconception, as the student may have changed focus to a different task. As others have noted [12], 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."

Our system addresses these problems by leveraging a well-understood computational model of actions and the causal relationships between them used in automated planning. We have previously published the details of the design for this system [11,10], and preliminary results of an evaluation of the accuracy of its student model [10], but this paper is the first to include a complete tutorial evaluation.

2 Related Work

Our work builds on two related threads of research: inquiry-based ITS and plan-based interactive pedagogical environments. The goal of what has been called inquiry-based or scientific discovery learning is to promote learning that is deep and conceptual. However, it has been repeatedly shown that exploration without competence guidance is insufficient to produce positive changes in learning outcomes [4,3]. Thus, the promise

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 369–376, 2011.

offered by inquiry learning is tempered by the problems students typically experience when using this approach. Fortunately, integrating supporting cognitive tools with computer simulations may provide a solution [2].

Steve [7] is an animated pedagogical agent who teaches human students to operate the engines of a naval surface ship in a virtual environment using on a deep and powerful model of task-based instruction. However, Steve has no student model. A student has the ability, but is never required, to seize the initiative from Steve and try to perform the next step(s) in the demonstration. Thus, although the task-based representation ensured that Steve's demonstrations were complete and correct, a student could finish the session having learned an entire procedure or nothing at all.

The Mission Rehearsal Exercise [9] extends the model of a pedagogical agent introduced by Steve to a more complex and dynamic environment. This simulation is embedded in an interactive narrative, with multiple non-player controlled agents each with its own set of goals and emotions. Like Steve, however, the content is cleverly presented to give more of an impression of student autonomy than actually exists. The pedagogical content is aligned within a tightly constrained progression through the story.

Crystal Island [8] shares Annie's goal of using planning to guide exploratory learning. Crystal Island produces a broad set of options for intervention, from lighting or sound changes that draw the user's attention toward a learning opportunity, direct or indirect dialogue supplied by non-player controlled (NPC) characters, and omnipotent environmental control that can be used to dynamically adjust world geography, obstacles, and other challenges. However, these guidance strategies are entwined with the planner, making it difficult to extend the design to new domains.

3 Design Overview

The system we have implemented is named "Annie", in recognition of Anne Sullivan, who used innovative and imaginative tutoring to guide the blind and deaf Helen Keller in learning to communicate with words. The idea that connects Anne with Annie is that both tutors have severely restricted communication bandwidth with their learners.

3.1 Architectural View

The architectural diagram shown in Fig. 1 emphasizes that our system runs as an independent peer of the learning environment through a messaging interface constrained to task execution. An architectural goal is for the tutorial algorithms in Annie to remain independent of any particular domain or exploratory environment. Therefore, all domain and environment knowledge is supplied as a run-time input to the system (shown in Fig. 1 as the **Learning Problem Description**, or **LPD**) prior to each tutorial session.

The LPD provides STRIPS-style [1] declarative descriptions of the pedagogically relevant tasks, including the preconditions that must be true for each task to execute and effects that can be expected upon successful execution. In addition, the LPD must provide explicit descriptions of the initial and goal states of the world, which, in effect, describes the learning challenge presented to the student. Annie considers it the student's job to discover a sequence of tasks that will transform the world from its initial state to the goal state.

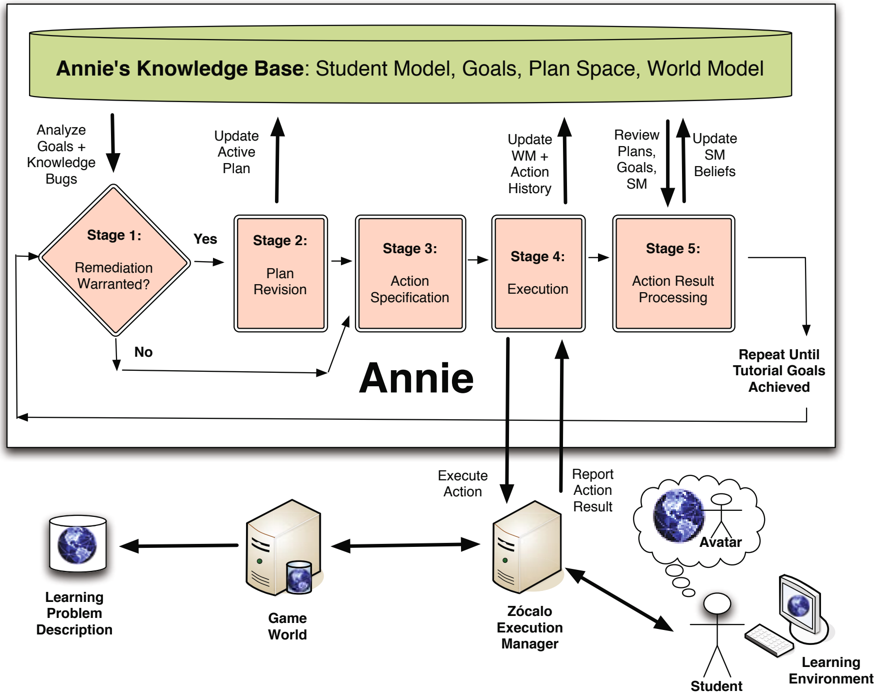

Annie's Knowledge Base: Student Model, Goals, Plan Space, World Model
Analyze Goals + Knowledge Bugs
Update Active Plan
Update WM + Action History
Review Plans, Goals, SM
Update SM Beliefs
Stage 1: Remediation Warranted?
Yes
No
Stage 2: Plan Revision
Stage 3: Action Specification
Stage 4: Execution
Stage 5: Action Result Processing
Repeat Until Tutorial Goals Achieved
Annie
Execute Action
Report Action Result
Avatar
Learning Problem Description
Game World
Zócalo Execution Manager
Student
Learning Environment

Fig. 1. Annie's Architecture

3.2 Plan-Based Reasoning

Annie uses the same task-based model to describe both the world and the learner's model of the world. Thus, the student model consists entirely of the task descriptions for each operator in the world, annotated with Annie's estimation of the likelihood that the student is knowledgeable of each of the preconditions and effects of each operator. These estimates are based entirely on the observation of the student's actions in the world. For example, if a student attempts to perform an action where some of the preconditions of that action have not yet been satisfied, Annie can choose to reduce its estimate of the likelihood the student is understands those preconditions.

Annie targets learning domains where the objective is to understand the relationships between tasks. For example, in the evaluation domain described later in this article, it is important for the student to learn which of three different tools is used in solving each of the first two learning challenges. Because Annie performs the cause-and-effect reasoning to deduce which tool is needed when, it can provide timely and appropriate guidance to the student.

Based on the LPD descriptions of the initial state and goal states of the world, Annie uses the Longbow planning algorithm [14] to generate at least one tutorial plan consisting of a plausible partially-ordered sequence of student and system-initiated actions designed to achieve a specific goal state for the world. The plan marks out the optimal tutorial path prior to the start of the session, but it is continually revised based on student actions that may or may not follow the plan's structure. In addition, Annie can generate alternative plans that highlight alternative exploratory routes the student might take to complete the given learning challenge.

Annie analyzes the space of potentially successful plans and ranks each of the unexecuted actions in those plans according to their potential proximity to the current state of the world. Annie cannot know which plan the student is most likely to be following, so it simply looks at all possible plans to recognize the set of most likely actions for the student to take next. This plan reasoning confers two advantages. First, it allows Annie to ignore consideration of actions unlikely to be executed soon. Second, it allows the more proximal tasks to be prioritized according to their proximity to more effectively target guidance.

4 Execution Loop

Like many ITSs [13], Annie's core tutorial reasoning is situated in a loop interleaving student and system-controlled actions. This loop consists of five stages as shown in the center of Fig. 1. Annie continuously updates its student model based on task success, failure or inaction. On each iteration through the loop it considers whether any of the observed misconceptions noted in the student model are sufficiently urgent as to require scaffolding based on the current state of plan execution. This allows Annie to move between tasks, or steps within tasks, to adapt to a user exploiting the exploratory nature of the environment.

Each time an action is taken in the world, either by the student or the system, Annie updates its student model by consulting a library of general **diagnostic** templates. 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. For example, if a student attempts an action for which some of the preconditions are not satisfied, a rule in one of these diagnostic templates fires to update the student model by lowering its confidence that the student is aware of those preconditions.

Annie uses the updated student model in consulting a second domain-independent library containing **remediation** templates that can be used to generate scaffolding. For example, if the plan shows that a particular task must be performed for the student to make progress toward plan goals, and Annie notes particular gaps in the student model pertaining to that action (e.g., student has an incorrect model of its effects), it will send a message to the execution environment to prompt the student about that action using text that is supplied for the action and effects as part of the LPD. Previous publications [10,11] describe Annie's design in more detail.

5 Experimental Evaluation

A previous experiment [10]) evaluated Annie's student model, providing evidence that a strong and statistically correlation exists between the predictions of learner knowledge made by Annie and those made by a human domain expert observing student behavior. This paper reports a subsequent evaluation of Annie's effectiveness in guiding exploratory learning. A total of 28 students enrolled in digital game design classes at North

Carolina State University were recruited to take part in the study. Each subject was tasked with finding and repairing problems in the fictional computer by using the appropriate tools on the appropriate objects and the appropriate times. A sequence of three learning challenges each required the student to perform several independent tasks.

5.1 Evaluation Environment

Annie's initial evaluation environment is called "FixIt," an operating system level game-based simulation of a computer under attack from various kinds of malware. The job of the student is to discover and fix problems using procedure-specific tools. For example, the first *learning challenge*, or *mission* is for the student to solve the problem of slow system response time. In this challenge, the student faces several learning tasks. First, the student uses a combination of observation and using the "Inspection" tool to deduce which of several processes is in a runaway state where it is consuming too much CPU. Then, the student must find, select, and use the "Nice" tool on the runaway process to reduce its resource consumption to normal levels. Three successive learning challenges are initiated within the simulation, where each challenge requires the student to perform a sequence of actions applying the appropriate tools to the appropriate entities in the environment.

5.2 Guidance Evaluation: Experimental Design

Each subject first completed a one page written pre-assessment, to gauge pre-test familiarity with the domain of operating systems concepts and computer malware. Then, a two minute narrated video walk-through of the FixIt was shown to each participant. The video ensured a consistent set of orientation instructions was given to each participant. In addition, the narration provided a natural conduit to associate the visual representations used in the system with their intended counterparts in the domain of operating system and malware.

After the video completed, Annie began the tutorial session and automatically assigned each subject to one of three treatment groups:

Control (no Annie-provided assistance): Subjects progressed through the FixIt environment, without any dynamically-generated guidance from Annie.

Ablated Annie: Annie exercised its full diagnostic and remediation capabilities with the exception that once a remediation (prompt, hint, etc.) was chosen for a given misconception, no further remediations (repeated prompts, more directive hints) were given for that particular misconception.

Full Annie: The highest level of scaffolding was identical to the Ablated Annie condition except that if the subject did not respond to the first issuance of a remediation, it was repeated after approximately a fifteen second delay. In addition, for the actions in the tutorial that required the subject to find tools, if the subject had not successfully found the tools after following the second prompt and delay, a five second video was played to show the general location of the tool.

Immediately following the FixIt session, subjects were given a post-assessment of their understanding of the subject domain. Thus, with two primary measures, and two methods for measuring them, the guidance study tested three main hypotheses:

Hypothesis 1. *Annie helps subjects complete more learning tasks.*
Hypothesis 2. *Annie helps subjects complete learning challenges (sets of tasks) faster.*
Hypothesis 3. *Annie increases pre-post domain knowledge gains.*

We tested each of these hypotheses for statistically significant group-wise differences between the three treatment treatment groups on the same measures.

5.3 Guidance Evaluation: Results Analysis

To test Hypothesis 1, a measure of the percentage of learning tasks completed by each subject was derived from the the FixIt logs. If a subject completed all three learning challenges, the learning task completion was set to 100. If the subject completed the first two learning challenges, the score was set to 70, and if the subject completed only the first challenge, the score was set to 40. Incremental points were given for students who perform some, but not all of the tasks required to complete an entire learning challenge. For example, if the student finds the "Nice" tool in learning challenge 1, selects it for use, but uses it on the wrong process, partial credit is given for completing learning challenge 1.

The SAS GLM procedure was used to measure variance between the treatment groups and the task completion percentages. This ANOVA revealed a highly significant effect for treatment: $F(2, 25) = 11.26, p < 0.0003$. A post-hoc Student-Newman-Keuls mean comparison test was run to find out which mean differences were responsible for this effect. This test found statistically significance differences for the mean task completion percentages between each the three treatment groups (34.44, 55, and 79.5). Together, these tests provide strong evidence for Hypothesis 1, in that the group of subjects receiving Full Annie treatment outperformed those in the Ablated Annie group, who in turn outperformed the No Annie control group.

Table 1. Hypothesis 2 Test Results

Measurement Value	ANOVA Findings	p-Value
LC 1 Seconds	$F(2, 25) = 9.20$	$p < 0.001$
LC 2 Seconds	$F(2, 25) = 10.73$	$p < 0.0004$
LC 3 Seconds	$F(2, 25) = 1.97$	$p < 0.1607$

Hypothesis 2: Group Differences Test. Hypothesis 2 measures the number of seconds required to complete each of the three major learning challenges. The ANOVA test results for Hypothesis 2, shown in the rightmost two columns of Table 1, mirror those of the correlation comparisons, in that statistically significant differences between treatment groups was observed for the completion times of learning challenges 1 and 2, but not for learning challenge 3. Post-hoc tests found statistically significant pairwise-differences for the mean completion times between the control group (No Annie) and the means of each of the two groups that received help from Annie. However, statistically significant mean differences were not seen between the Ablated Annie and Full

Annie groups on this measure. For challenge 2, the statistically significant mean differences were found between the Full Annie group as compared to the Ablated Annie and No Annie control groups, but no significant difference between the mean times for the Ablated Annie and No Annie control groups.

Hypothesis 3: Pre-post Learning Gains. Learning gains were measured by comparing the differences between post-test and pre-test written answers to the malware knowledge assessment questions for each subject to test the hypothesis that Annie has a positive impact on such gains. A weak, but statistically insignificant positive correlation $(r = 0.303, p < 0.1167)$ was found between the degree of Annie treatment and the pre-post learning gain measure, providing insufficient evidence was produced to accept or reject Hypothesis 3.

5.4 Guidance Evaluation: Discussion

The guidance evaluation results show that the automatic scaffolding generated by Annie helped subjects complete more tasks and complete those tasks in less time. Statistically strong support was found to accept Hypothesis 1: the full Annie group outperformed the Ablated Annie group who outperformed the Control group on the same measure, with statistically significant mean differences between the three treatment groups.

Support also found for Hypothesis 2, in that Annie helped subjects finish the first two learning challenges faster and that the group-wise differences in completion speed for these two challenges was significant. Insufficient evidence was found to support this hypothesis with regard to learning challenge 3, but this is likely to have been a result of the fact that only two of 28 subjects were able to complete the entire set of learning tasks in the time provided. In addition, unsuccessful completions introduced a floor effect because they were recorded as if they task was completed at the maximum time value.

Insufficient evidence was found to either support or reject Hypothesis 3, that Annie would have a positive impact on pre-test to post-test gains. Factors that likely contributed to this finding were the short (8 minute) duration of the learning experience, the coarse-grained assessment potential of just a few relevant questions on the survey, and the low degrees of freedom from having fewer than 12 subjects per group.

6 Conclusion

This paper describes the first full evaluation of Annie, a domain-independent platform for guiding exploratory learning in task-based environments. Our experimental evaluation showed Annie had strong, positive, and statistically significant impact on increasing the number of learning tasks students completed and reducing the time it took to complete them. Although we did not find a significant pre-post learning gain for domain knowledge, this initial study was limited to a short eight minute learning experience. In future work, we hope to evaluate Annie in a new domain, externalize our diagnostic and remediation libraries, and strengthen Annie's reasoning capabilities.

References

1. Fikes, R., Nilsson, N.: STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intellligence 2(3/4) (1971)
2. de Jong, T., van Joolingen, W.: Scientific discovery learning with computer simulations of conceptual domains. Review of Educational Research 68(2), 179–201 (1998)
3. 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)
4. Mayer, R.: Should there be a three-strikes rule against pure discovery learning. American Psychologist 59(1), 14–19 (2004)
5. Puntambekar, S., Hubscher, R.: Tools for Scaffolding Students in a Complex Learning Environment: What Have We Gained and What Have We Missed? Educational Psychologist 40(1), 1–12 (2005)
6. 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)
7. Rickel, J., Johnson, W.: Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. Applied Artificial Intelligence 13(4-5), 343–382 (1999)
8. Rowe, J., Mott, B., McQuiggan, S., Robison, J., Leed, S., Lester, J.: Crystal Island: A Narrative-Centered Learning Environment for Eighth Grade Microbiology. In: 14th International Conference on AI in Education Workshops Proceedings, p. 11 (2009)
9. Swartout, W., Hill, R., Gratch, J., Johnson, W., Kyriakakis, C., LaBore, C., Lindheim, R., Marsella, S., Miraglia, D., Moore, B., et al.: Toward the Holodeck: Integrating Graphics, Sound, Character and Story. Defense Technical Information Center (2006)
10. Thomas, J.M., Young, R.M.: Annie: Automated generation of adaptive learner guidance for fun serious games. IEEE Transactions on Learning Technologies 3, 329–343 (2010)
11. 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)
12. Van Joolingen, W., De Jong, T., Dimitrakopoulou, A.: Issues in computer supported inquiry learning in science. Journal of Computer Assisted Learning 23(2), 111–119 (2007)
13. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education 16(3), 227–265 (2006)
14. Young, R., Pollack, M., Moore, J.: Decomposition and causality in partial-order planning. In: Proceedings of the Second International Conference on AI and Planning Systems, vol. 48 (1994)

Clustering Students to Generate an Ensemble to Improve Standard Test Score Predictions

Shubhendu Trivedi, Zachary A. Pardos, and Neil T. Heffernan

Department of Computer Science, Worcester Polytechnic Institute,
Worcester, MA-01609 United States
{s_trivedi,zpardos,nth}@wpi.edu

Abstract. In typical assessment student are not given feedback, as it is harder to predict student knowledge if it is changing during testing. Intelligent Tutoring systems, that offer assistance while the student is participating, offer a clear benefit of assisting students, but how well can they assess students? What is the trade off in terms of assessment accuracy if we allow student to be assisted on an exam. In a prior study, we showed the assistance with assessments quality to be equal. In this work, we introduce a more sophisticated method by which we can ensemble together multiple models based upon clustering students. We show that in fact, the assessment quality as determined by the assistance data is a better estimator of student knowledge. The implications of this study suggest that by using computer tutors for assessment, we can save much instructional time that is currently used for just assessment.

Keywords: Clustering, Ensemble Learning, Intelligent Tutoring Systems, Regression, Dynamic Assessment, Educational Data Mining.

1 Introduction

Feng *et al.*[1] reported the counter-intuitive result that data from an intelligent tutoring system could better predict state test scores if it considered the extra measures collected while providing the students with feedback and help. These measures included metrics such as number of hints that students needed to solve a problem correctly and the time it took them to solve. That paper [1] was judged as best article of the year at User Modeling and User-Adapted Interaction and was cited in the National Educational Technology plan. It mentions a weakness of the paper concerning the fact that time was never held constant. Feng *et al.* go one step ahead and controlled for time in following work [2]. In that paper, students did half the number of problems in a dynamic test setting (where help was administered by the tutor) as opposed to the static condition (where students received no help) and reported better predictions on the state test by the dynamic condition, but the difference was not statistically reliable. This present work starts from Feng *et al.* [2] and investigates if the dynamic assessment data can be better utilized to increase prediction accuracy over the static condition. We use a newly introduced method that clusters students, creates a mixture of experts and then ensembles the predictions made by each cluster model to achieve a reliable improvement.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 377–384, 2011.

2 Literature Review

The Bayesian knowledge tracing model [3] and its variants [4] [5] have become the mainstay in the Intelligent Tutoring System (ITS) community to track student knowledge. This knowledge estimate is used for calibrating the amount of training students require for skill mastery. One of the most important aspects of such modeling is to ensure that performance on a tutoring system is transferred to actual post tests. If this is not the case, then that implies over-training within the tutoring system. In fact, it is reasonable to say that one of the most important measures of success of a tutoring system is its ability to predict student performance on a post-test. Since such a transfer is dependent on the quality of assessment, a tension exists between focusing on quality of assessment and quality of student assistance.

Traditionally, performance on a post-test is predicted by using practice tests. Practice tests based on past questions from specific state tests can give a crude estimate of how well the student might perform in the actual state test. Improving this estimate would be highly beneficial for educators and students. For improving such assessment, dynamic assessment [6] has long been advocated as an effective method. Dynamic assessment is an interactive approach to student assessment that is based on how much help a student requires during a practice test. Campione *et al.* [7] compared the traditional testing paradigm, in which the students are not given any help, with a dynamic testing paradigm in which students are given graduated hints for questions that they answer incorrectly. They tried to measure learning gains for both the paradigms from pre-test to post-test and suggested that such dynamic testing could be done effectively with computers. Such assessment makes intuitive sense as standard practice tests simply measure the percent of questions that a student gets correct. This might not give a good estimate of a student's knowledge limitations. If a student gets a question wrong, it might not necessarily imply absence of knowledge pertaining to the question. It is likely that the student has some knowledge related to the question but not enough to get it correct. It is thus desirable to have a fine grained measure of the knowledge limitations of the student during assessment. Such a measure might be obtained by monitoring the amount of help the student needs to get to a correct response from an incorrect response. ITS provide the tools for doing dynamic assessment more effectively as they adapt while interacting with individual students and make it easier to provide interventions and measure their effect. Fuchs *et al.* [9] studied dynamic assessment focusing on unique information, such as how responsive a user is to intervention. Feng *et al.* [1][2] used extensive information collected by the ASSISTments tutor [13] to show that the dynamic assessment gives a relatively better prediction as compared to static assessment. This work effectively showed that dynamic assessment led to better predictions on the post test. This was done by fitting a linear regression model on the dynamic assessment features and making predictions on the MCAS test scores.

They concluded that while dynamic assessment gave good assessment of students, the MCAS predictions made using those features lead to only a marginally statistically significant improvement as compared to the static condition. In this paper we explored the dynamic assessment data to see if we could make significantly better predictions on the MCAS test score. A significant result would further validate the use of ITS as a replacement to static assessments.

2 Data

The dataset that we considered was the same as used by Feng *et al.*[2]. It comes from the 2004-05 school year, the first full year when ASSISTments.org was used in two schools in Massachusetts. ASSISTments is an e-learning and e-assessing research platform developed at Worcester Polytechnic Institute. Complete data for the 2004-05 year was obtained for 628 students. The data contained the dynamic interaction measures of the students and the final grades obtained in the state test (MCAS) taken in 2005. The dynamic measures were aggregated as students used the tutor.

2.1 Metrics

The following metrics were developed for dynamic testing by Feng *et al.* [2] and were used in these experiments. They try to incorporate a variety of features that summarize a student's performance in the system. The features were as follows: 1) the student's percent correct on the main problems 2) number of problems done 3) percent correct on the help questions 4) average time spent per item 5) average number of attempts per item and 6) average numbers of hints per item. Out of these, only the first was as a static metric and was used to predict the MCAS score in the static condition. The other five and a dynamic version of student's percent correct on the main problems were used to make predictions in the dynamic condition.

The predictions were made on the MCAS scores. The MCAS or the Massachusetts Comprehensive Assessment System is a state administered test. It produces tests for English, Mathematics, Science and Social Studies for grades 3 to 10. The data set we explore is from an 8th grade mathematics test.

3 Methodology

The data was split into randomly selected disjoint 70% train and 30% test sets. Feng *et al.*[2] fit a stepwise linear regression model using the dynamic assessment features on the training set to make a prediction on the MCAS scores on the test set. They reported an improvement in prediction accuracy with a marginal statistical significance relative to the predictions made only using data from the static condition. Fitting in a single linear regression model for the entire student data might be a bad idea for two reasons. First, the relationship between the independent variables (dynamic assessment features) and the dependent variables (MCAS test scores) might not be a linear one. If so, training a linear model would have high bias for the data and no matter how much data is used to train the model, there would always be a high prediction error. The second conceivable source of error is related to the first. A student population would have students with varying knowledge levels, thus requiring different amounts of assistance. Thus it might be a bad idea to fit the entire population in a single model. Students often fall into groups having similar knowledge levels, assistance requirements, etc. It is thus worth attempting to fit different models for different groups of students. It, however, must be noted that while such groups could be identified using clustering, the groups obtained may not be easily interpretable.

3.1 Clustering

The previous section mentions that it might not be a good idea to fit in a single model for the entire student population and that there might exist groups of students having similar knowledge levels and nature of responses to interventions. A natural method to find such patterns in the data is by clustering. If data was generated by a finite set of distinct processes, then clustering methods are maximum likelihood methods to identify such underlying processes and separating them. The idea in this work is to fit in a linear regression model for each such group in the training set. The prediction for the MCAS score for each student from the test set would thus involve two steps: identification of the cluster to which the student from the test set belongs and then using the model for that cluster to make the prediction of the MCAS score for the student.

We used K-means clustering for the identification of K groups. The initialization of cluster centroids was done randomly and the clusters were identified by using Euclidean distance. K-means finds out the best separated clusters by trying to minimize a distortion function. The distortion function is a non-convex function and thus implies that K-means is susceptible to getting stuck in local optima. This means that when K-means is run with random cluster centroids; we might not reach the best solution possible. To reduce the chances of getting a sub-optimal clustering we restarted K-means 200 times with random initialization.

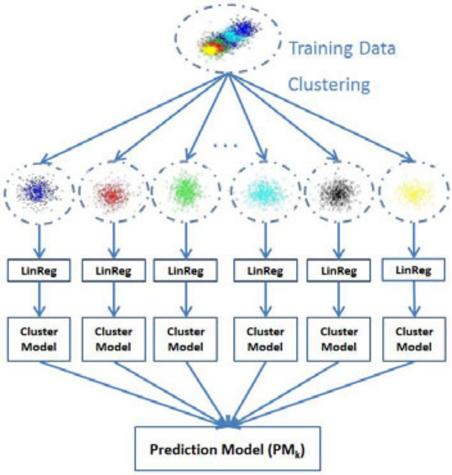


Fig. 1. Schematic illustrating the steps for obtaining a prediction model (PM_K). There would be one such prediction model for each value of K chosen (1 to K would give K prediction models).

For each cluster identified we trained a separate linear regression model (Fig. 1). We call such a linear regression model (for each cluster) a cluster model. For data separated into K clusters there would be K cluster models. All of these K cluster models taken together make predictions on the entire test set. These K cluster models together can be thought to form a more complex model. We call such a model a prediction model i.e. PM_K, with the subscript K identifying the number of cluster models in the prediction model. Feng *et al.* [2] used the prediction model PM_1, since only a single linear regression model was fit over the entire data-set. The value of K can be varied from 1 to K to obtain K prediction models. For example: if K = 1, 2 and 3, there would be three prediction models - PM_1 having a single cluster model (K=1), PM_2 having two different cluster models (K=2) and PM_3, that is the prediction model with three different cluster models (K=3). It is noteworthy that the cluster models in different prediction models would be different.

If K prediction models are constructed from the data, there would be a set of K different predictions on the test data. These predictions are compared with those obtained on $PM_{1,}$ i.e. a linear regression model fit over the entire data-set to see if there is an improvement in prediction accuracy. An improvement would indicate a strong result that dynamic assessment indeed gives a much better assessment of student learning.

3.2 Ensemble Learning

Section 3.1 described how, by using K as a controllable parameter, we can obtain a set of K prediction models and K corresponding predictions. The training data is first clustered by K-means and K clusters are obtained. For each of the clusters we fit a linear regression model, which we called the cluster model. The cluster models together are referred to as a prediction model. This prediction model makes a prediction on the entire test set. But since K is a free parameter, for each value of K we get a different prediction model and a different set of predictions. For example when K=2, the prediction model will have two cluster models. When K=7, the prediction model will have 7 cluster models. Thus, by means of clustering, we generate a number of prediction models.

While we are interested in looking at how each prediction model performs. It would also be interesting to look at ways in which the K predictions can be combined together to give a single prediction. Such a combination of predictors leads to ensembling. Ensemble methods have seen a rapid growth in the past decade in the machine learning community [12][13][14].

An ensemble is a group of predictors each of which gives an estimate of a target variable. Ensembling is a way to combine these predictions with the hope that the generalization error of the combination is lesser than each of the individual predictors. The success of ensembling lies in the ability to exploit diversity in the individual predictors. That is, if the individual predictors exhibit different patterns of generalization, then the strengths of each of the predictors can be combined to form a single stronger predictor. Dietterich [12] suggests three comprehensive reasons why ensembles perform better than the individual predictors. Much research in ensembling has gone into finding methods that encourage diversity in the predictors.

3.2.1 Methodology for Combining the Predictions

We have a set of K predictors. The most obvious way of combining them is by some type of averaging. The combination could also be done using Random Forests [10], but they have not been explored in this work as we are extending work that simply used linear regression. We explored two methods for combining these predictors.

1. Uniform Averaging: This is the simplest method for combining predictions. The K predictions obtained (as discussed in section 3.1) are simply averaged to get a combined prediction. In addition to averaging all predictions we could also choose to average just a subset of the predictions together.
2. Weighted averaging: In uniform averaging, each predictor is given the same weight. However, it is possible that the predictions made by some model are more important than the predictions made by another model. Thus, it is reasonable to combine the models by means of a weighted average. Such weighted averaging could be done by means of a linear regression. Since we did not find an improvement with weighted averaging, the methodology and results are not discussed in detail.

4 Results

4.1 Prediction Models

The data was first clustered with K taken from 2 to 7. Clustering beyond 7 clusters was problematic as it returned empty clusters. Hence the experiments were restricted to a maximum of K=7 for this dataset. The prediction on the MCAS was made first by using PM_1. Then, K was varied from 2 to 7 and a set of six more predictions on the MCAS were obtained (all dynamic features were used). The Mean Absolute Difference (MAD) and the Root Mean Square Errors (RMSE) of the MCAS in the test set were found. This section summarizes these results. It also compares the results with the static condition.

Table 1. Prediction errors by different prediction models

Model	**MAD**	**p-value (with PM_1)**	**p-value (with static)**	**RMSE**
Static	10.4900	0.0180	-	12.7161
PM_1	9.6170	-	0.0180	11.5135
PM_2	9.3530	0.1255	0.0036	11.4286
PM_3	9.3522	0.2005	0.0074	11.4377
PM_4	9.3005	0.1975	0.0062	11.5243
PM_5	9.3667	0.3375	0.0067	11.7291
PM_6	9.3518	0.2347	0.0052	11.5100
PM_7	9.4818	0.6138	0.0134	11.6762

Almost all Prediction Models (Table 1) showed a statistically significant improvement in prediction as compared to the static condition demonstrating greater assessment power using the dynamic condition. However, though there is an improvement in the error as compared to the Prediction Model 1, the improvement is not statistically significant, as was previously found to be the case [1].

4.2 Averaging Predictions

As reported in section 4.1 the prediction models do not show a statistically significant improvement in prediction accuracy of the MCAS score relative to the PM_1. As discussed in section 3.2, combining them might lead to improved predictions. This section reports these results.

Table 2. Prediction errors by different prediction models averaged. The subscripts refer to the models whose predictions were used in averaging.

Model	**MAD**	**p-value (with PM_1)**	**p-value (with static)**	**RMSE**
Static	10.4900	0.0180	-	12.7161
PM_1	9.6170	-	0.0180	11.5135
$PM_{1\text{ to }4}$	9.2375	0.0192	0.0013	11.3042
$PM_{1\text{ to }5}$	9.2286	0.0251	0.0012	11.3405
$PM_{1\text{ to }6}$	9.2268	0.0260	0.0012	11.3412
$PM_{1\text{ to }7}$	9.2398	0.0365	0.0013	11.3511
$PM_{2\text{ to }4}$	9.2604	0.0526	0.0022	11.3379
$PM_{2\text{ to }5}$	9.2406	0.0540	0.0018	11.3818
$PM_{2\text{ to }6}$	9.2348	0.0475	0.0016	11.3753
$PM_{2\text{ to }7}$	9.2507	0.0630	0.0017	11.3830

Averaging across prediction models clearly improves predictions as compared to the prediction models taken alone (Table 2). The improvement is not just in the error but also in terms of statistical significance and thus improves the results reported in 4.1. These results validate the idea that clustering helps in predictions. These results show how the dynamic assessment prediction accuracy can be further improved.

4 Contributions

This paper makes one clear contribution. This is the first paper we know of that clearly demonstrates that not only can an Intelligent Tutoring System allow students to learn while being assessed but also indicates a significant gain in assessment accuracy. This is important, as many classrooms take away time from instruction to administer tests. If we can provide such a technology it would save instruction time and give better assessment and would thus be highly beneficial to students and instructors. The second contribution of this paper is the application of clustering student data and ensembling predictions that we are introducing to the field in a KDD paper [15]. In that paper we applied this approach to a number of datasets from the UC Irvine Machine Learning repository and reported a prediction improvement in all datasets.

Acknowledgments

For the full list of over a dozen funders please see http://www.webcitation.org/5xp605MwY.

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) (2009)
2. Feng, M., Heffernan, N.T.: 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). In: Proceedings of the 3rd International Conference on Educational Data Mining?, pp. 41–50 (2010)
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. Pardos, Z.A., Heffernan, N.T.: Using HMMs and bagged decision trees to leverage rich features of user and skill from an intelligent tutoring system dataset. Journal of Machine Learning Research C & WP (in press 2011)
5. Baker, R.S.J.d, Corbett, A. T., Aleven, V.: More Accurate Student Modeling Through Contextual Estimation of Guess and Slip Probabilities in Bayesian Knowledge Tracing. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, Brightion, UK, pp. 531–538.
6. Grigerenko, E.L., Steinberg, R.J.: Dynamic Testing. Psychological Bulletin 124, 75–111 (1998)
7. Campione, J. C., Brown, A. L.: Dynamic Assessment: One Approach and some Initial Data. Technical Report. No. 361. Cambridge, MA. Illinois University, Urbana, Center for the Study of Reading. ED 269735 (1985)
8. 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' of Mathematical Problem Solving. Journal of Educational Psychology 100(4), 829–850 (2008)
9. Fuchs, D., Fuchs, L.S., Compton, D.L., Bouton, B., Caffrey, E., Hill, L.: Dynamic Assessment as Responsiveness to Intervention. Teaching Exceptional Children 39(5), 58–63 (2007)
10. Breiman, L.: Random Forests. Machine Learning 45(1), 5–32 (2001)
11. Baker, R.S., Corbett, A.T., Koedinger, K.R., Wagner, A.Z.: Off-task behaviour in the Cognitive Tutor Classroom: When Students "game the system". In: Proceedings of the ACM CHI 2004: Computer - Human Interaction, pp. 383–390. ACM, New York (2004)
12. Dietterich, T.G.: Ensemble Methods in Machine Learning. In: Kittler, J., Roli, F. (eds.) First International workshop on Multiple Classifier Systems. LNCS, pp. 1–15. Springer, New York (2000)
13. Dietterich, T.G.: An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. Machine Learning 40, 139–157 (2000)
14. Brown, G., Wyatt, J.L., Tino, P.: Managing Diversity in Regression Ensembles. Journal of Machine Learning Research 6, 1621–1650 (2005)
15. Trivedi, S., Pardos, Z.A., Heffernan, N.T.: The Utility of Clustering in Prediction Tasks. In: Submission to the 17th Conference on Knowledge Discovery and Data Mining (in submission, 2011)

Using Automated Dialog Analysis to Assess Peer Tutoring and Trigger Effective Support

Erin Walker[1], Nikol Rummel[2], and Kenneth R. Koedinger[1]

[1] Carnegie Mellon University
[2] Ruhr Universität Bochum
erin.a.walker@gmail.com, nikol.rummel@rub.de, koedinger@cmu.edu

Abstract. Intelligent tutors have the potential to be used in supporting learning from collaboration, but there are few results demonstrating their positive effects in this domain. One of the main challenges in automated support for collaboration is the machine classification of dialogue, giving the system an ability to know when and how to intervene. We have developed an automated detector of conceptual content that is used as a basis for providing adaptive prompts to peer tutors in high-school algebra. We conducted an after-school study with 61 participants where we compared this adaptive support to two nonadaptive support conditions, and found that adaptive prompts significantly increased conceptual help and peer tutor domain learning. The amount of conceptual help students gave, as determined by either human coding or machine classification, was predictive of learning. Thus, machine classification was effective both as a basis for feedback and predictor of success.

Keywords: intelligent tutoring, peer tutoring, adaptive collaboration support.

1 Introduction

Computer-mediated collaborative learning activities have been demonstrated to improve student domain learning [1]. When students articulate their reasoning as part of interacting with others, they can engage in *beneficial cognitive processes*; they may reflect on misconceptions, elaborate on existing knowledge, and generate new knowledge [2]. However, without guidance, students may not collaborate in ways that lead them to benefit. One potential remediation is to add intelligent tutoring technologies that can assess the quality of collaboration as it occurs and provide targeted support. This support might lead students to engage in more beneficial cognitive processes as they try to collaborate better, causing an improvement in domain learning [3]. In a small number of studies, adaptive support for collaboration quality has indeed shown to be better than no support and nonadaptive support at increasing domain learning [e.g., 4]; in another small set, adaptive support has been shown to improve collaboration quality directly [e.g., 5]. However, there are no studies that have demonstrated an effect on both collaboration *and* learning. Thus, a causal link between adaptive support, improved collaboration, and learning has yet to be established. We explore that link by investigating three hypotheses (Figure 1): Adaptive support improves student

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 385–393, 2011.

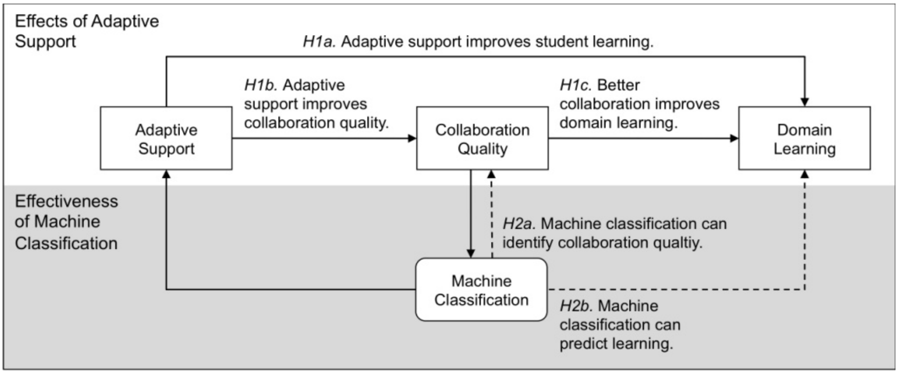

Effects of Adaptive Support
H1a. Adaptive support improves student learning.
H1b. Adaptive support improves collaboration quality.
H1c. Better collaboration improves domain learning.
Adaptive Support
Collaboration Quality
Domain Learning
Effectiveness of Machine Classification
H2a. Machine classification can identify collaboration qualtiy.
Machine Classification
H2b. Machine classification can predict learning.

Fig. 1. Hypotheses investigated. We explore the link between support, collaboration, and learning, and test how well our system classifies collaboration quality and learning.

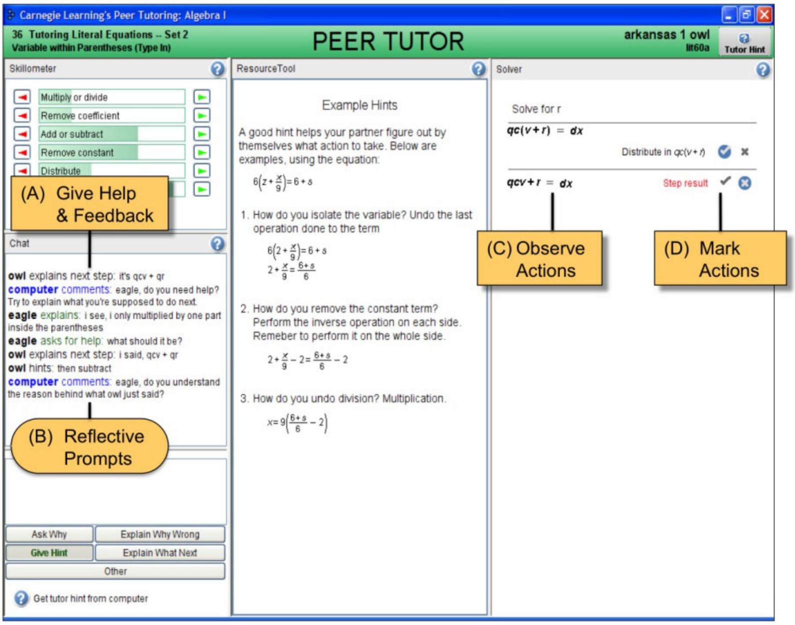
learning (*H1a*), improves collaboration quality (*H1b*), and better collaboration quality relates to improved domain learning (*H1c*).

One reason these hypotheses have not been fully explored might be that tutoring systems for collaborative learning are hard to construct. Collaboration quality is linked to properties of student dialogue, and to adaptively support this dialogue its properties need to be classified in real-time. In many existing systems, dialogue is assessed by having students self-classify their utterances [6]. For example, students may select a sentence starter like "I disagree, because…" in order to signal an instance of constructive conflict. However, students do not consistently select sentence starters that match the content of their statements, and therefore the inferences that the system makes can be inaccurate [7]. Consequently, researchers are starting to use machine classification to label student dialogue as it occurs, with goals ranging from determining the conversation topic to labeling a student's argument [8, 9]. As the quality of dialogue relates to whether students benefit from collaboration [10], improving our ability to automatically classify properties of student utterances would have two potential benefits: a) It would increase our ability to target support to those utterances, b) Given a relationship between collaboration and domain learning, it would enable us to predict learning based on the machine classification. Thus, this paper also investigates two technical hypotheses (Figure 1): Machine classification can identify collaboration quality (*H2a*) and predict domain learning (*H2b*).

We investigated these hypotheses in the context of an intelligent tutoring system for reciprocal peer tutoring in algebra, called the Adaptive Peer Tutoring Assistant (*APTA*). Reciprocal peer tutoring is a type of collaborative learning activity where two students of similar abilities take turns tutoring each other [11]. The goal of *APTA* is to improve peer tutors' domain learning by providing adaptive support for their help. In giving help, peer tutors benefit from reflecting and elaborating on their knowledge [2]. These beneficial cognitive processes can be triggered when peer tutors construct high quality help [10], but peer tutors tend to need support to do so. One type of high-quality help is conceptual help, in that it references domain concepts as part of a hint or explanation. For example, the phrase "You need to subtract the *ax* to get the two *x*'s on the same side" would be considered conceptual. Fuchs and colleagues trained peer tutors to give conceptual help, and found that tutors that received this training learned more than tutors that did not [12]. In *APTA*, we follow up on these results by using a machine classification of conceptual help to support peer tutors in giving more conceptual help. We discuss a study where we assessed whether *APTA* improved the conceptual content of peer tutor help and peer tutor domain learning. We then examine the effectiveness of *APTA* for classifying peer tutor conceptual help and serving as a basis for feedback. Although there are other aspects of student dialogue that are supported by our system and may relate to learning, given the length of this paper we focus here on conceptual help.

2 The Adaptive Peer Tutoring Assistant (APTA)

APTA is a peer tutoring addition to the Cognitive Tutor Algebra, a successful individual intelligent tutoring system for high school algebra [13]. In *APTA*, one student tutors another on literal equation solving problems where they are given an equation like "$ax + by = cx + dy$" and a prompt like, "Solve for x". Students are seated at different computers. Using menus, the tutee can select operations like "subtract from both sides" and then type in the term they would like to subtract. Peer tutors can see the tutee's actions, but are not able to perform actions in the problem themselves (C in Figure 2). Instead, they mark the tutee's actions right or wrong (D in Figure 2). Students discuss the problem in a chat window (A in Figure 2).



owl explains next step: it's qcv + qr
computer comments: eagle, do you need help? Try to explain what you're supposed to do next.
eagle explains: i see, i only multiplied by one part inside the parentheses
eagle asks for help: what should it be?
owl explains next step: i said, qcv + qr
owl hints: then subtract
computer comments: eagle, do you understand the reason behind what owl just said?

Example Hints

A good hint helps your partner figure out by themselves what action to take. Below are examples, using the equation:

$6\left(z+\frac{x}{9}\right)=6+s$

1. How do you isolate the variable? Undo the last operation done to the term

$6\left(2+\frac{x}{9}\right)=6+s$
$2+\frac{x}{9}=\frac{6+s}{6}$

2. How do you remove the constant term? Perform the inverse operation on each side. Remeber to perform it on the whole side.

$2+\frac{x}{9}-2=\frac{6+s}{6}-2$

3. How do you undo division? Multiplication.

$x=9\left(\frac{6+s}{6}-2\right)$

Solve for r

$qc(v+r) = dx$

Distribute in $qc(v+r)$ ✓ ✗

$qcv+r = dx$

Step result ✓ ✗

Fig. 2. Peer tutor's interface in APTA. The peer tutor watches the tutee take problem-solving steps, and marks them correct or incorrect. The peer tutor helps the tutee in the chat window.

APTA provides peer tutors with prompts in the chat in order to encourage them to reflect and elaborate on their domain knowledge while providing more conceptual help. The computer prompts the peer tutor to reflect in the chat window (e.g., “Owl, think about the last help you gave. Why did you say that? Can you explain more?”), where “owl” is the peer tutor). These prompts are visible to both students (B in Figure 2), and might include positive reinforcement (“Good work! Hinting or explaining the reason for a step can help your partner learn how to do the step”), or tips for giving better help (“Owl, when helping, use examples or facts your partner already understands”). *APTA* incorporates prompts related to four different skills, namely (1) giving help when needed, (2) giving help targeting errors, (3) giving conceptual help, and (4) using the interface appropriately. Here, we focus on conceptual help.

Our assessment of whether students were giving conceptual elaborated help was based on an automated classification of student dialogue, described in [14]. We generated a baseline machine classifier for *conceptual content* using Taghelper Tools, state of the art text-classification technology designed for coding collaborative dialogue [9]. We then improved the accuracy of the classifier by adding three different types of domain features: problem-solving context (e.g., whether the tutee has just made an error), text substitutions (e.g., whether the peer tutor uses a domain-related word, like “add” or “isolate”), and substitution history (e.g., how many times in the past a given peer tutor has used a domain-related word). Training our automatic classification on previous study data, we achieved a kappa of 0.72 when compared to human raters. We expected accuracy to be lower when we deployed the system in the current study, given the change of population. Nevertheless, we used the machine classification of each dialogue utterance as part of a knowledge tracing model that assessed whether peer tutors knew how to give conceptual help, and, if not, triggered reflective prompts at relevant moments.

3 Method

As described in the introduction, we were interested in evaluating effects of adaptive support on the conceptual content of peer tutor help (*H1a*) and domain learning (*H1b*), with the hypothesis that conceptual help relates to learning (*H1c*). In a controlled study, we compared an adaptive support condition to two nonadaptive conditions. In the *real adaptive condition*, students received relevant prompts based on the automated assessment (using *APTA*). They were told that the prompts they received were adaptive (“The computer will watch you tutor and give you targeted advice when you need it based on how well you tutor”). In the *told adaptive condition,* we still told students that support was adaptive, using the above instructions. However, students were actually given nonadaptive support, where they received randomly selected prompts at moments when they would not have received the adaptive prompts. We ensured that the adaptive and random prompts appeared with the same frequency. In the *real nonadaptive condition,* students received the nonadaptive support and were told the support was not adaptive (“From time to time, the computer will give you a general tip chosen randomly from advice on good collaboration”). Including these two control conditions was an attempt at separating the cognitive effects of receiving support tailored to one’s collaborative actions from the motivational effects of believing support is adaptive. If receiving adaptive support is indeed beneficial for improving help given by tutors, the *real adaptive condition* would have a better effect than the *told adaptive* and *real non-adaptive* conditions.

Participants were 130 high-school students (49 males, 81 females) from one high school, currently enrolled in Algebra 1, Geometry, or Algebra 2. While the literal equation solving unit was one that all students had (in theory) received instruction on in Algebra 1, the teacher we were working with nevertheless identified it as a challenging unit for the students. The study was run at the high school, either immediately after school or on Saturdays. All students were paid 30 dollars for their participation, and as a result, appeared to be highly motivated during the study activities. Students participated in sessions of up to 9 students at a time. Each session was randomly assigned to one of the three conditions. Students came with partners that they had chosen, except for 4 students to whom the researchers then assigned partners. Within each pair students were randomly assigned to the role of tutee or tutor. Eight students worked alone and were not included in the analysis, leaving 122 students. For the purposes of this paper, we focus on peer tutor interaction and learning, and thus analyze data from 61 peer tutors.

Students first took a 20-minute domain pretest, and then spent 20 minutes working individually using the CTA to prepare for tutoring. They were then assigned either the tutor or tutee role. Students spent a total of 60 minutes in a tutoring phase, with one student tutoring another student. Finally, students took a 20-minute domain posttest. The pretests and posttests were counterbalanced, and contained conceptual and procedural items relating directly to the literal equation solving domain. To assess help quality and the accuracy of the automated classification, we human coded peer tutor help during tutoring for *conceptual content* by scoring whether each peer tutor utterance contained a reference to one or more domain concept. For example, "add *ax* to cancel out the -*ax*" and "cancel out the –*ax*" were conceptual, while "add *ax*" and "add *ax* so you can factor" were not. A total of 3105 utterances were made by peer tutors, and coded. To compute interrator reliability two independent raters coded 647 utterances separately, and achieved a kappa of 0.79.

4 Effects of Adaptive Support

To investigate the effects of condition on peer tutor learning (*H1a*), we conducted a one-way ANCOVA, with posttest score as the dependent measure, condition as a between subjects variable and pretest score as a covariate (see Table 1). Condition had a significant effect on posttest score ($F[2,57] = 4.47$, $p = 0.016$), and pretest was also significantly predictive of posttest score ($F[1, 57] = 33.24$, $p < 0.001$). Post-hoc contrasts revealed that students in the real adaptive condition learned significantly more than students in the real nonadaptive condition ($p = 0.019$) and marginally more than students in the told adaptive condition ($p = 0.077$), controlling for pretest. Overall, providing adaptive support led peer tutors to learn more, suggesting that the adaptive support triggered beneficial cognitive processes related to domain learning.

Next, we tested *H1b*, examining whether condition had an effect on conceptual content of tutor help. Here, we used negative binomial regression, because the outcome variable, conceptual content, was a count variable that was not normally distributed. We included two dummy coded condition variables in the regression, one representing the *told adaptive* condition and one representing the *real nonadaptive* condition, so that both could be compared to the *real adaptive* condition. We controlled for total help given by the peer tutor (all utterances that contained any domain information), which, using an ANOVA, was not significantly different between conditions (*F*[1,58] = 1.82, $p = 0.17$). The told adaptive condition was negatively related to the amount of conceptual help compared to the real adaptive condition ($\beta = -0.922$, $\chi 2(1, N = 61) = 3.976$, $p = 0.046$), and the real fixed condition was not significantly different from the real adaptive condition ($\beta = -0.310$, $\chi 2(1, N = 61) = 0.565$, $p = 0.452$). Essentially, when all else is held constant, the *real adaptive* condition is responsible for roughly 2.51 more instances of conceptual help per student than the *told adaptive* condition, and 1.36 more instances of conceptual help per student than the *real nonadaptive* condition. The total help was also related to the amount of conceptual help ($\beta = 0.039$, $\chi 2(1, N = 61) = 5.841$, $p = 0.016$).

Table 1. Domain learning scores and amount of conceptual help

	Pretest Score	**Posttest Score**	**Conceptual Help**	**Total Help**
Real Adaptive	0.27 (0.15)	0.39 (0.17)	4.16 (5.89)	26.00 (12.10)
Told Adaptive	0.24 (0.12)	0.27 (0.14)	1.77 (2.76)	32.55 (12.51)
Real Nonadaptive	0.29 (0.16)	0.28 (0.18)	3.15 (4.58)	29.85 (7.56)

Finally, we wanted to determine whether the conceptual help peer tutors gave was related to their domain learning (*H1c*). We conducted a linear regression with posttest score as the dependent measure, and conceptual content and pretest score as predictor variables. We also included the dummy coded condition variables to separate the overall effects of condition from the effects of conceptual content. We found that the conceptual content of help was marginally predictive of learning ($\beta = 0.199$, $t(60) = 1.95$, $p = 0.071$). As in our test of *H1a*, taking part in the actually adaptive condition significantly influenced learning compared to the *real nonadaptive* condition ($\beta = 0.308$, $t(60) = 2.64$, $p = 0.011$), and marginally influenced learning compared to the *told adaptive* condition ($\beta = 0.227$, $t(60) = 1.92$, $p = 0.060$). In sum, increased conceptual help partially mediated the effect of condition and learning, but there are likely other (yet unknown) interaction factors that had positive effects on learning.

5 Effectiveness of Machine Classification

We then examined how accurately our system assessed conceptual help. First, we compared the machine classification to the human codes on an utterance level (*H2a*). Table 2 displays the confusion matrix for the conceptual help codes. While the percent accuracy of the codes is 94%, with the vast majority of non-conceptual help correctly classified, *Cohen's kappa* is 0.53, as only 50% of the conceptual help instances were correctly classified. On the surface, this result would indicate that our classifier was less successful than we might have hoped. However, we can also explore the relationship between the human and computer coding on a student level, rather than on an utterance level, in order to assess more generally whether a given student has developed the ability to give conceptual help. The correlation between the human and machine count of instances of conceptual help *per student* was significant ($r[59] = 0.855$, $p < 0.001$), suggesting that the computer classification is overall accurate at determining whether students know how to give conceptual help. Thus, two goals of our classifier were met: a) It could identify instances of nonconceptual help in order to provide relevant support, and b) it could determine if a given student had the ability to give conceptual help by looking at the overall machine classifier count for that student. Further, *H2b* asked whether the machine classification of conceptual help could predict domain learning. Running the same regression as in Section 4, with posttest score as the dependent measure, and computer coded conceptual content, pretest score, and condition as predictor variables, we found that the computer classification was as predictive of student learning as the human classification ($\beta = 0.225$, $t(60) = 2.15$, $p = 0.036$). Using the machine classification of conceptual help, we can predict whether peer tutors will learn from the activity.

Table 2. Confusion matrix for machine and human classification of conceptual content

		Computer Codes	
		not conceptual	conceptual
Human Codes	not conceptual	2793	117
	conceptual	66	116

6 Discussion

In this paper, we described *APTA*, a system for adaptively supporting peer tutors in high school algebra. We discussed the component of *APTA* that detects peer tutor use of conceptual help and provides relevant prompts. We found that the prompts significantly increased peer tutor learning and the conceptual help peer tutors gave their partners. The amount of conceptual help given was marginally predictive of peer tutor learning, suggesting that there was indeed a causal link between the adaptive support provided, the increase in peer tutor conceptual help, and the increase in peer tutor learning. However, the relative weakness of the relationship between conceptual help and learning, and the differing pattern of results between the control conditions (the *real nonadaptive* condition learned the least but the *told adaptive* condition gave the least amount of conceptual help) suggested that there were other mediating factors at play. In fact, some of these factors may be relatively undetectable; the adaptive support, by prompting peer tutors to reflect at relevant moments, might increase their beneficial cognitive processes without having a tangible effect on the help they give. Further, it is likely that certain aspects of the peer tutor and tutee *interaction* (such as how much the tutee builds on peer tutor ideas) might have a positive effect on peer tutor learning. Nevertheless, this paper takes a step towards identifying the mechanisms by which adaptive support might lead to greater learning.

The second contribution of this paper is a technical one, examining the effectiveness of our machine classifier for conceptual help in this domain. On an utterance level the classifier was not as accurate as we might have hoped at positively identifying instances of conceptual help. However, on a practical level, the classifier was successful, and the support based on the classifier proved to be effective at improving conceptual help and domain learning. Indeed, accurate detection of non-conceptual help instances may be more valuable than accurate detection of conceptual help instances. Interestingly the classifier was accurate at assessing whether a given student was overall able to give conceptual help, and successfully predicted learning based on these classifications. This result suggests that these machine classifiers can function effectively as broader assessments of collaborative skill and domain learning.

This paper has focused on supporting conceptual help in a peer tutoring activity. However, we believe our results generalize to other collaborative learning activities, as conceptual elaboration and help exchange are key elements of collaboration in general. One might also extend this technology to support a student in interacting with teachable agents or companion agents. By developing an understanding of how adaptive support can assess student collaboration, influence collaboration quality, and improve student domain learning, we can build powerful intelligent support systems for human-human and human-agent collaborative learning activities.

Acknowledgments. This work was supported by the Pittsburgh Science of Learning Center, NSF Grant #SBE-0836012. Thanks to Ruth Wylie, Ido Roll, Amy Ogan, Sean Walker, Carolyn Rosé, and the mathematics coordinator who made this possible.

References

1. Lou, Y., Abrami, P.C., d'Apollonia, S.: Small group and individual learning with technology: A meta-analysis. R. Ed. Res. 71(3), 449–521 (2001)
2. Ploetzner, R., Dillenbourg, P., Preier, M., Tram, D.: Learning by explaining to oneself and to others. In: Dillenbourg, P. (ed.) Collaborative Learning: Cognitive and Computational Approaches, pp. 103–121. Elsevier Science Publishers, Amsterdam (1999)
3. Rummel, N., Weinberger, A.: New Challenges in CSCL: Towards Adaptive Script Support. In: Kanselaar, E., Jonker, V., Kirschner, P.A., Prins, F. (eds.) Proc. ICLS 2008, pp. 338–345. International Society of the Learning Sciences (2008)
4. Kumar, R., Rosé, C.P., Wang, Y.C., Joshi, M., Robinson, A.: Tutorial dialog as adaptive collaborative learning support. In: Luckin, R., Koedinger, K.R., Greer, J. (eds.) Proc. AIED 2007, pp. 383–390. IOS Press, Amsterdam (2007)
5. Baghaei, N., Mitrovic, A., Irwin, W.: Supporting Collaborative Learning and Problem-Solving in a Constraint-Based CSCL Environment for UML Class Diagrams. IJCSCL 2(2-3), 159–190 (2007)
6. Soller, A., Martinez, A., Jermann, P., Mühlenbrock, M.: From mirroring to guiding: A review of state of the art technology for supporting collaborative learning. IJAIED 15, 261–290 (2005)
7. Israel, J., Aiken, R.: Supporting collaborative learning with an intelligent web-based system. IJAIED 17(1), 3–40 (2007)
8. Kumar, R., Rosé, C.P., Wang, Y.C., Joshi, M., Robinson, A.: Tutorial dialogue as adaptive collaborative learning support. In: AIED 2007, pp. 383–390 (2007)

9. Rosé, C., Wang, Y.-C., Cui, Y., Arguello, J., Stegmann, K., Weinberger, A., Fischer, F.: Analyzing collaborative learning processes automatically: Exploiting the advances of computational linguistics in computer-supported collaborative learning. IJCSCL 3(3), 237–271 (2008)
10. Webb, N.M., Mastergeorge, A.: Promoting effective helping behavior in peer-directed groups. International Journal of Educational Research 39, 73–97 (2003)
11. Dillenbourg, P., Jermann, P.: Designing integrative scripts. In: Fischer, F., Mandl, H., Haake, J., Kollar, I. (eds.) Scripting Computer-Supported Communication of Knowledge - Cognitive, Computational and Educational Perspectives, pp. 275–301. Springer, Heidelberg (2007)
12. Fuchs, L., Fuchs, D., Hamlett, C., Phillips, N., Karns, K., Dutka, S.: Enhancing students' helping behavior during peer-mediated instruction with conceptual mathematical explanations. The Elementary School Journal 97(3), 223–249 (1997)
13. Koedinger, K., Anderson, J., Hadley, W., Mark, M.: Intelligent tutoring goes to school in the big city. IJAIED 8, 30–43 (1997)
14. Walker, E., Walker, S., Rummel, N., Koedinger, K.R.: Using problem-solving context to assess help quality in computer-mediated peer tutoring. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 145–155. Springer, Heidelberg (2010)

Evaluating a General Model of Adaptive Tutorial Dialogues

Amali Weerasinghe[1], Antonija Mitrovic[1], David Thomson[1],
Pavle Mogin[2], and Brent Martin[1]

[1] Intelligent Computer Tutoring Group, University of Canterbury, New Zealand
[2] Victoria University of Wellington, Wellington New Zealand
{amali.weerasinghe,david.thomson}@pg.canterbury.ac.nz,
pavle.mogin@ecs.vuw.ac.nz,
{tanja.mitrovic,brent.martin}@canterbury.ac.nz

Abstract. Tutorial dialogues are considered as one of the critical factors contributing to the effectiveness of human one-on-one tutoring. We discuss how we evaluated the effectiveness of a general model of adaptive tutorial dialogues in both an ill-defined and a well-defined task. The first study involved dialogues in database design, an ill-defined task. The control group participants received non-adaptive dialogues regardless of their knowledge level and explanation skills. The experimental group participants received adaptive dialogues that were customised based on their student models. The performance on pre- and post-tests indicate that the experimental group participants learned significantly more than their peers. The second study involved dialogues in data normalization, a well-defined task. The performance of the experimental group increased significantly between pre- and post-test, while the improvement of the control group was not significant. The studies show that the model is applicable to both ill- and well-defined tasks, and that they support learning effectively.

Keywords: adaptive tutorial dialogues, constraint-based tutors, Ill-defined tasks, well-defined tasks.

1 Introduction

One of the aspirations of AIED research is to explore how intelligent systems can achieve the same effectiveness as in human one-on-one tutoring. One of the major factors contributing to the effectiveness of human tutors is the conversational aspect of instruction. Dialogues provide opportunities for students to reflect on their existing knowledge and to construct new knowledge. Some of the existing dialogue-based tutoring systems are Why2-Atlas [1], Auto Tutor [2], CIRCSIM-Tutor [3], Geometry Explanation Tutor [4] and KERMIT-SE [5]. Why2-Atlas and Auto Tutor use dialogues as the main learning activity, while the others provide problem-solving as the main activity and use tutorial dialogues as a way of remediating student errors. For example, CIRCSIM-Tutor is a natural language tutor that helps students learn cardiovascular physiology related to regulation of blood pressure. The Geometry Explanation Tutor requires students to justify the problem-solving steps in their own words. KERMIT-SE, a database design tutor, engages students in dialogues when

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 394–402, 2011.

their solutions are erroneous. All these tasks except database design are well-defined: problem solving is well-structured, and therefore explanations expected from learners can be clearly defined. In contrast, database design is an ill-defined task: the final result is defined only in abstract terms, and there is no algorithm to find it [6].

Our goal is to develop a general model for supporting dialogues across domains. Based on the findings of two Wizard-of-Oz studies [7], we developed a model consisting of three parts: an error hierarchy, tutorial dialogues and rules for adapting them. The error hierarchy categorizes all error types in a domain. At the leaf level, an error type is associated with one or more violated constraints. (The knowledge bases of our constraint-based tutors are represented in terms of constraints.) The error types are then grouped into higher-level categories. Remediation is facilitated through tutorial dialogues, one of which is developed for each error type. When there are multiple errors in a student solution, the hierarchy is traversed to select the error most suitable for discussion and the corresponding dialogue is then initiated. Finally, the adaptation rules are used to individualize the dialogues to suit the student's knowledge and reasoning skills by controlling their timing and the exact content. In response to the generated dialogue learners are able to provide answers by selecting an option from a list. For a detailed discussion of the model see [7].

In this paper we discuss how we evaluated the effectiveness of our model supporting an ill-defined and a well-defined task. The first study investigated the effectiveness of our model in database design (an ill-defined task), in the context of EER-Tutor [8]. In database design, students design database schemas using the EER model. Students need to know the concepts of the EER data model, use world knowledge about different real-world scenarios (i.e. enrolling students in a university etc.) and be able to handle the ill-definedness of the task. In the second study, we evaluated our model in data normalization, using NORMIT [8]. Data normalization is the process of refining a relational database schema in order to ensure that all relations are of high quality. This task requires normalizing a given database schema using the specified procedure. NORMIT contains a page for each step of this procedure, and students are requested to complete one step before continuing with the next one. The following two sections present the results of the study, followed by discussion and conclusions.

2 EER-Tutor Study

We conducted a study with the EER-Tutor in March 2010 at the University of Canterbury, which involved volunteers from an introductory database course. The objective of the study was to investigate whether adaptive dialogues are more effective in improving learning than non-adaptive dialogues in database design.

The participants were randomly assigned to groups. The experimental group received adaptive dialogues, while the control group had non-adaptive dialogues. The differences between the two groups were in dialogue selection, dialogue prompts and additional support. Dialogues for the control group were selected using the depth-first traversal of the error hierarchy. The first violated constraint that was found in the traversal was selected for discussion. As the errors in the hierarchy are ordered from simpler to more complicated errors, the depth-first search results in the simplest error for the control group.

The dialogues in our model consist of four stages [7]: (i) a problem-independent prompt discusses the relevant domain concept for the selected error; (ii) a problem-dependent prompt discusses the error in the context of the current problem; (iii) a corrective action prompt provides an opportunity to understand how to correct the error and (iv) a reinforcement prompt, providing another opportunity to learn the related domain concept. The control group saw the entire dialogue regardless of the number of times they have seen the dialogue previously or their responses to the dialogue prompts. As the result, the same solution submitted by two different students with different knowledge levels in the control group received identical dialogues. In contrast, an experimental group participant receives the problem-dependent prompt (prompt (ii)) the first time a mistake is done. If s/he makes this type of error repeatedly, the dialogue will start from the problem-independent prompt. The exit point of the dialogue for the experimental group is customized based on the student's past interactions with the dialogues. For a detailed description, see [7].

When an experimental group participant abandons a problem (i.e. changes a problem without attempting it) or has been inactive for a period of time, they were asked whether they needed help. If they requested help then their solution was evaluated and an error was selected for discussion based on their student model. The control group did not receive this support.

The study consisted of four stages: pre-test, interactions with EER-Tutor, post-test and questionnaire. The pre- and post-tests had 6 questions each, of similar difficulty. We wanted to evaluate whether students' problem-solving abilities as well as explanation skills improved after interacting with the system. One question asked the participants to provide the database schema for the given requirements. This is a typical question that can be found in examinations, text books etc. The other three questions were aimed to understand the effect the system had on students' explanation skills.

The participants used EER-Tutor for the first time in their regular lab sessions during the third week of the course, for a single 2-hour session. At the beginning of the session students were given about 10 minutes to complete the pre-test, after which they interacted with the system. Towards the end of the session, they were given 10 minutes to complete the post-test and 5 minutes to answer a questionnaire.

Out of 104 students enrolled in the course, 77 participated in the study. There was no significant difference in the pre-test performance between the control and the experimental groups. Some students have not completed the post-test. Table 1 reports some statistics about the 65 participants who completed both pre-and post-tests.

Table 1. Some statistics from the EER-Tutor study (sd given in parentheses)

	Control (34)	Experimental (31)	p
Pre-test (%)	54.5 (18.1)	51.3 (16.1)	ns
Post-test mean (%)	61.2 (14.9)	69.9 (11.5)	0.005
Gain	6.8(15.6)	18.6 (16.8)	0.002
Normalised gain	0.002 (0.7)	0.3 (0.4)	0.01
Interaction time (min)	62.8 (22.1)	62.9 (24.1)	ns
Attempted Problems	8.6(4.8)	10.6(4.8)	ns
Solved problems	9.0(4.8)	7.9 (4.7)	ns
Total Dialogues received	12.1 (7.3)	14.0 (8.3)	ns
Questions answered	34.4 (25)	23.6 (14.6)	0.01
% of correct answers	61.4 (23.1)	59 (16.9)	ns

There were 31 participants in the experimental group and 34 in the control group, with no significant difference on the pre-test performances. The post-test performance of the experimental group was significantly better compared to their peers who received non-adaptive dialogues. Both the learning gain (post-test score – pre-test score) and the normalised learning gain[1] of the group who received adaptive dialogues was also significantly higher than the gains of the control group.

There were no differences between the times spent with the system, the numbers of attempted and solved problems, and the number of dialogues received. The control group answered a significantly higher number of questions than their peers. This was expected, as the control group had to go through the entire dialogue before resuming problem-solving. However, percentages of correct answers are similar for both groups.

The effect size[2] (Cohen's d) for learning gains of the two groups is 0.69 (the effect size based on the normalized gain is 0.51). The effect size obtained here is remarkable because the only difference between the two groups was the adaptivity of the dialogues. In order to investigate how the students learnt the database design concepts in terms of constraints, we analyzed how frequently constraints were violated. Figure 1 illustrates the learning curves for both groups. The probabilities of violating a constraint on the first and subsequent attempts were averaged over all students. The x-axis represents the attempt number (first, second and so on) when a student violated a constraint. The y-axis shows the probability of violating these constraints. The probability of making a mistake is initially higher for the experimental group than the control group even though not significantly. Figure 1 indicates that both groups learnt the constraints in a similar manner.

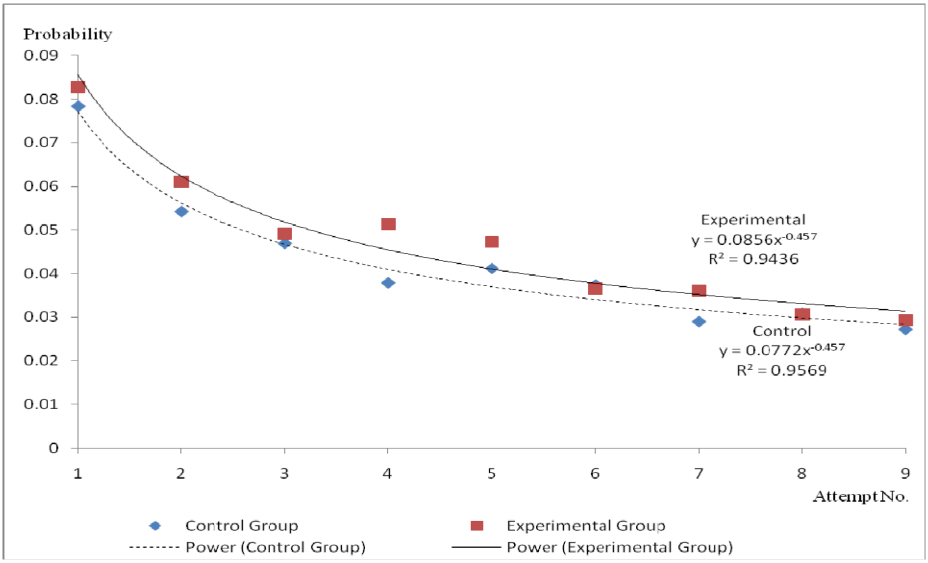


Fig. 1. Probability of constraint violations – EER-Tutor study

[1] Normalised learning gain =learning gain/(1-pre-test score).

[2] Effect size = (Experimental Mean – Control Mean) /Standard Deviation of both groups.

We also investigated the number of constraints learnt by both groups. We used the first five attempts and the last attempts on each constraint to decide whether the status of the constraint changed from ‘not known’ to ‘learnt’ for a given student. If the probability of violating a constraint is below a pre-defined threshold then the constraint was deemed not known. Similarly, if the probability of violating a constraint is above the same pre-defined threshold then it was considered to be learnt. This analysis revealed that the experimental group learnt a significantly higher number of constraints than the control group (2.3 vs 1.2, p= 0.02).

Table 2 presents the subjective responses about various aspects of the dialogues. The impression about the quality of the dialogues and the ease of understanding the questions were similar between the groups. However there was clear evidence that the control group did not like having to go through the entire dialogue.

Table 2. Subjective responses about tutorial dialogues (sd given in parentheses)

Question	Likert scale	Control	Experimental	p
Quality of the dialogues	Poor to Excellent (1 to 5)	3.5 (1.0)	3. 7(0.8)	ns
Length of the dialogues	Too long to Too short (1 to 5)	2.6 (0.9)	3.2 (0.5)	0.002
Ease of understanding the questions	Very Hard to Very Easy (1 to 5)	3. (1.0)	3.4 (0.8)	ns

3 NORMIT Study

We conducted a study with NORMIT in September 2010 at the Victoria University of Wellington, which involved 20 volunteers from a database system engineering course in a single, 1-hour session. The objective and the experimental setup for this study are similar to that of EER-Tutor study. Pre-and the post-tests were designed to explore the system’s effect on both the students’ problem-solving abilities and explanation skills. Both pre- and post-tests had 4 questions each, of similar difficulty. Two questions requested students to solve very simple problems, and explain their solutions. The other two questions requested students to specify definitions of concepts. Some students have not completed the post-test. Table 3 reports some statistics about the 18 participants who completed both tests. Each group had 9 students.

Table 3. Some statistics from the NORMIT study (sd given in parentheses)

	Control (9)	Experimental (9)	p
Pre-test (%)	68.1 (30.0)	69.4 (29.4)	ns
Post-test (%)	72.2 (24.0)	86.1(15.9)	ns
Gain	4.2 (32.4)	16.7 (27.2)	ns
Interaction time (min)	60.1(24.7)	47.7 (16.8)	ns
Attempted Problems	7.1 (3.0)	5.9 (2.1)	ns
Solved problems	6.1 (3.0)	5.4 (2.0)	ns
Total Dialogues received	27.8 (14.6)	23.6 (11.3)	ns
Questions answered	55.7 (37.4)	23.9 (11.5)	0.01
% of correct answers	6.9 (4.1)	8.2 (4.7)	ns

There were no significant differences between the pre-test and post-test performances of the two groups, as well as between the gains. The performance of the experimental group increased significantly between pre- and post-test (paired t-test, t=1.84, p=0.052), while the improvement of the control group was not significant. The effect size for learning gains of the two groups is 0.4.

As the study was limited to a single lab session, the two groups spent a similar time interacting with the system. The groups attempted and solved a similar number of problems, and received a similar number of dialogues.

The control group participants answered significantly more questions than their peers, as was the case in the EER-Tutor study. This can be expected as the control group had to go through the entire dialogue every time a dialogue is given to the student. However, percentages of correct answers are similar for both groups.

Figure 2 presents the learning curves for both groups. The probability of making a mistake is initially higher for the experimental group than the control group even though not significantly. The learning curves indicate that the learning rate of the experimental group is higher than that of the control group. Similar to the EER-Tutor study, we also investigated the number of constraints learnt by both groups. There was no significant difference between the numbers of constraints learnt.

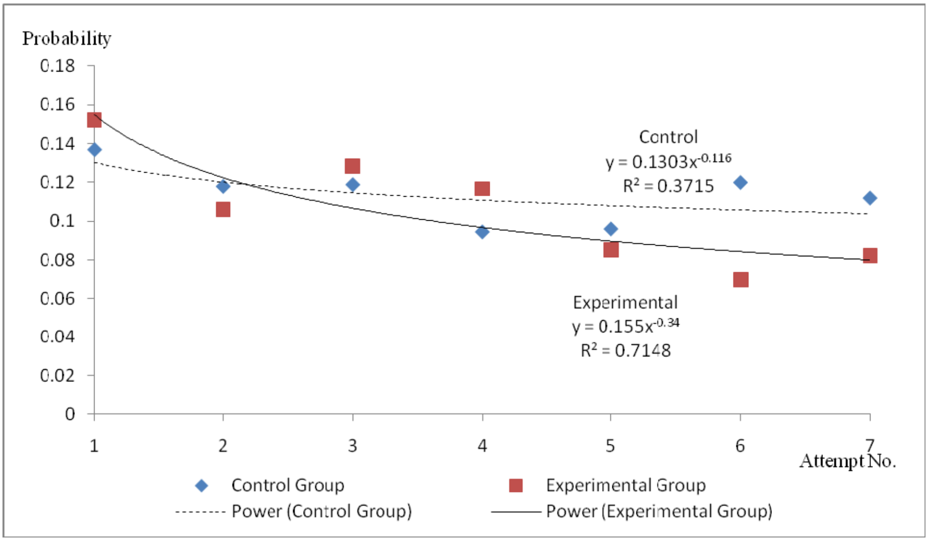


Fig. 2. Probability of constraint violations – NORMIT study

We also explored the users' impressions about various aspects of tutorial dialogues using questionnaires (Table 4). The questions used for the EER-Tutor study were used here. The impression about the quality of the dialogues and the ease of understanding the questions were similar between the groups. Unlike the EER-Tutor study, there was no evidence from the control group that the non-adaptive dialogues were too long.

Table 4. Subjective responses about tutorial dialogues (sd given in parentheses)

Question	Likert scale	Control	Experimental	p
Quality of the dialogues	Poor to Excellent (1 to 5)	3.3 (0.5)	3.1(1.0)	ns
Length of the dialogues	Too long to Too short (1 to 5)	3.1 (0.8)	3.3(0.5)	ns
Ease of understanding the questions	Very Hard to Very Easy (1 to 5)	3.4(0.7)	3.1(0.7)	ns

4 Discussion and Conclusions

We presented how we evaluated the effectiveness of our model for supporting tutorial dialogues in two very different tasks. Our model facilitates adaptive dialogues based on a student's knowledge and their interaction with the dialogues. The dialogues discuss a student's mistake in the current context and the relevant domain concepts.

In EER-Tutor study the learning gain of the experimental group (that received adaptive dialogues) is significantly higher than the gain of their peers, with the effect size of 0.69. The experimental group also learnt a significantly higher number of constraints. These results strongly suggest that adaptive dialogues had a positive effect on learning database design. This is a significant result because (i) the difference between the two groups was minimal (i.e. the only difference was the adaptivity of the dialogues) and (ii) the study was limited to a single 2- hour session.

In the NORMIT study, there were no significant differences between the pre-test and post-test performances of the two groups, as well as between the gains. This might be due to the small number of participants (20 vs 65 in EER-Tutor study). However, we can observe similar trends in learning in both studies: significantly higher number of constraints learnt in EER-Tutor study, and a higher learning rate in NORMIT study by the respective experimental groups compared to their peers.

In both studies we used dialogues to discuss the errors in the problem-solving process, and not as the main activity to learn the domain knowledge. The task facilitated in EER-Tutor requires world knowledge about different real-world scenarios such as enrolling students in a university, or customers interacting with a bank. In the EER-Tutor study, the model was used to support dialogues in an ill-defined task with the well-defined domain theory. In the NORMIT study, dialogues facilitated learning a well-defined task with the well-defined domain theory. Therefore, our model has shown evidence of enhancing learning of a domain in the WDIT quadrant (well-defined domain, ill-defined task) and WDWT quadrant (well-defined domain, well-defined task) [6]. As the next step, we plan to explore the possibility of developing the model for a task such as essay writing or legal argumentation in the IDIT quadrant (Ill-defined domain, Ill-defined task).

The three highest levels of the error-hierarchy (the first component of the model) are domain-independent. The top level node is *All Errors*, which is then further divided into *Basic Syntax Errors* and *Errors dealing with the main problem solving activity*. The latter is further divided into (i) *Using an incorrect solution component type*, (ii) *Extra solution components,* (iii) *Missing solution components,* (iv) *Associations* and (v) *Failure to complete related changes*. Further divisions of these nodes and the node *Basic Syntax Errors* deal with domain-specific concepts. Even though tutorial dialogues consist of domain-specific prompts, the structure is domain-independent. Adaptation rules (the last component) which customise dialogue prompts are domain-independent except for the time period of inactivity the tutor waits before intervening.

We also investigated whether our model can be used in other domains. We tried to fit the errors from two different domains: logical database design and fraction addition into our model. Logical database design involves mapping high-level, conceptual ER schemas to relational schemas using the 7-step mapping algorithm [9]. We used the constraint-base of ERM-Tutor [10], a constraint-based tutor for teaching logical database design and developed the error hierarchy categorizing all the constraints. Then we explored whether we could develop dialogues for each type of error. All these were done on paper and the model could be developed for logical database design. We repeated the steps of (i) developing the error hierarchy using the constraints developed for fraction addition and (ii) developing dialogues for each type of error. The outcome of our attempt is a model that could be implemented to support dialogues in fraction addition. Therefore we have developed models for four different domains: (i) database design (ii) data normalization (iii) logical database design and (iv) fraction addition. The first two were implemented and evaluations indicate that the model can enhance learning the domain knowledge. The last two were done on paper and our attempt provides evidence that the model can be used in different domains.

For a newly created constraint-based tutor, developing our model to support dialogues involves (i) developing the error hierarchy to categorize the errors in the domain using the constraint-base (ii) designing the dialogues for each type of error and (iii) customizing the domain-dependent features (i.e. inactive time period) in the adaptation rules. Furthermore, even though this model was developed for constraint-based tutors, it can be used in any ITS with a problem-solving environment. In such ITSs, a student solution is evaluated and feedback is provided on errors regardless of the mechanism/methodology used for diagnosis. Therefore, the error hierarchy (the first component of the model) could be developed using the error types of that domain. Tutorial dialogues (the second component of the model) need to be written for each type of error based on the dialogue structure. The third component of the model, rules for adapting dialogues, are domain independent (except for the inactive time period), and can be used across domains.

The future work includes conducting a larger NORMIT study and exploring the possibility of developing a model for an ill-defined task in an ill-defined domain.

References

1. 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), 3–52 (2007)
2. Graesser, A.C., Lu, S., Jackson, G.T., Mitchell, H.H., Ventura, M., Olney, A., et al.: AutoTutor: A tutor with dialogue in natural language. Behavioral Research Methods, Instruments and Computers 36, 180–193 (2004)
3. Evens, M., Michael, J.: One-on-One Tutoring By Humans and Computers. Lawrence Erlbaum Associates, Mahwah (2006)

4. Aleven, V., Ogan, A., Popescu, O., Torrey, C., Koedinger, K.: Evaluating the Effectiveness of a Tutorial Dialogue System for Self-Explanation. In: Lester, J., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 443–454. Springer, Heidelberg (2004)
5. Weerasinghe, A., Mitrovic, A.: Facilitating Deep Learning through Self-Explanation in an Open-ended Domain. Knowledge-based and Intelligent Tutoring Systems 10(1), 3–19 (2006)
6. Weerasinghe, A., Mitrovic, A., Martin, B.: Towards Individualized Dialogue Support for Ill-Defined Domains IJAIED. Special Issue on Ill-Defined Domains 19(4), 357–379 (2009)
7. Mitrovic, A., Weerasinghe, A.: Revisiting the Ill-Definedness and Consequences for ITSs. In: Dimitrova, V., et al. (eds.) Proc. Artificial Intelligence in Education, Frontiers in Artificial Intelligence and Applications, vol. 200, pp. 375–382 (2009)
8. Mitrovic, A., Martin, B., Suraweera, P.: Intelligent Tutors for All: Constraint-based Modeling Methodology, Systems and Authoring. IEEE Intelligent Systems 22(4), 38–45 (2007)
9. Elmasri, R., Navathe, S.: Fundamentals of Database Systems, 5th edn. Addison-Wesley, Boston (2007)
10. Milik, N., Marshall, M., Mitrovic, A.: Teaching logical database design in ERM-Tutor. In: Ikeda, M., Ashley, K. (eds.) Proc. of ITS 2006, pp. 707–709 (2006)

A Reflective Tutoring Framework Using Question Prompts for Scaffolding of Reflection

Longkai Wu and Chee-Kit Looi

Natioanl Institue of Education, Nanyang Technological Univeristy, Singapore
{longkai.wu,cheekit.looi}@nie.edu.sg

Abstract. In the context of tutoring, question prompts can be important in enhancing a learner's reflection in learning. This paper describes the design and implementation of a reflective tutoring framework within an inquisitive simulated tutee environment that seeks to scaffold learner's reflection in pursuing tutoring activities. The results of an empirical study suggest that such a framework could afford the investigation of incorporating question prompts in a simulated tutee system to foster reflection and learning.

Keywords: Reflective Tutoring Framework, Question Prompts, Scaffolding of Reflection.

1 Introduction

Historically, the educational research literature has suggested that question prompts (from teachers, peers, software, or texts) can promote reflection and learning by eliciting explanations. Rothkopf [1] investigates the ways in which questions inserted in texts affected subjects' understanding of the texts. Chi, deLeeuw, Chiu, & LaVancher [2] indicate that questions that elicited self-explanations led to improved understanding of texts. Students who provide explanations to other students' questions or who explain examples they find in their textbooks seem to strengthen connections among their ideas [3]. Moon [4] suggests structuring reflection with questions to deepen the quality of reflection.

Such a view has led to the incorporation of question prompts into ILE (Intelligent Learning Environment) designs (e.g., [5]). Question prompts are used as scaffolds to help direct students towards learning-appropriate goals, such as focusing student attention and modeling the kinds of questions students should be learning to ask [6]. Positive evidences are found for question prompts to help students with various aspects, such as knowledge integration [7] and ill-structured problem-solving processes [8]. It is also suggested that self-efficacy and metacognitive prompting could increase problem-solving performance and efficiency separately through activation of reflection and strategy knowledge [9]. However, not enough research has looked at how question prompts can be used as a scaffolding strategy to elicit reflection when pursuing tutoring activities with a computer simulated tutee.

This paper explores how a reflective framework addresses the challenge of facilitating learners' reflection within an inquisitive simulated tutee enabled by the

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 403–410, 2011.

generation of question prompts. Here, a learner's reflection mainly refers to an intermingled process of knowledge construction and metacognition as a direct result of his/her engagement in instructional activities inherent to the tutoring process with simulated tutee, such as explaining, answering questions from the tutee, correcting errors of the tutee and asking questions to the tutee [10, 11]). The opportunity for reflection enables learners to monitor their own understanding, recognize and repair knowledge gaps and misconceptions, integrate new knowledge with prior knowledge, and generate new ideas for self-evaluation and reflection [12].

We argue that a reflective tutoring framework, with consideration of tutoring stages, reflection types and self-efficacy levels, is needed to incorporate question prompts into simulated tutee environment systematically to foster reflection and learning. Question prompts can be effective to provide support for the cognitively complex ways learners think about, feel, and make connections with experience (e.g. [7]). By engaging in reflective activities such as responding to the different types of question prompts (generic and specific prompts in this study), learners with different levels of self-efficacy could build their understanding and locate the significance of their activity in a larger context. Thus a learner is enabled to observe the meaning he has drawn from the experience and excavate the underlying qualities that made the experience significant [13].

2 Reflective Tutoring Framework

2.1 Proposed Framework

Based on a reflection assistant model proposed by Gama [14], three aspects were considered to construct a reflective tutoring framework by incorporating question prompts to guide learners for tutoring activities: Tutoring Stages, Reflection Types and Self-Efficacy Levels (Fig. 1). We discuss the three aspects respectively in the following sections.

2.2 Tutoring Stages

The reflective tutoring framework offered a structured approach to help learners proceed through the tutoring activities within four stages (Fig. 2), which follow the

Aspects		Value
Tutoring Stages	→	Familiarization Production Evaluation Post-Task Reflection
Reflection Types	→	Double-Loop Reflection Single-Loop Reflection
Self-Efficacy Levels	→	High Self-Efficacy Low Self-Efficacy

Fig. 1. Reflective Tutoring Framework

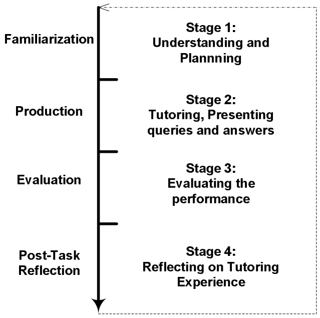

Familiarization
Stage 1: Understanding and Plannning
Production
Stage 2: Tutoring, Presenting queries and answers
Evaluation
Stage 3: Evaluating the performance
Post-Task Reflection
Stage 4: Reflecting on Tutoring Experience

Fig. 2. Tutoring Stages

conceptual stages in the practice of tutoring [14], and provide different question prompts through their actions in accordance with the structured stages.

As shown in Figure 2, the *Familiarization* stage allows the learner to self-assess his/her understanding of domain knowledge and learning difficulties, as well as selecting his/her metacognitive strategies. The *Production* stage enables the learner to teach the simulated tutee what they have learned (e.g., constructing concept maps) and monitoring the simulated tutee's understandings. The *Evaluation* stage provides the learner with opportunities to evaluate the performance of the simulated tutee, as well as their own performance. The *Post-Task Reflection* stage is to promote post-practice reflection on the tutoring experiences and the strategies being implemented.

2.3 Reflection Types

This study adopts and adapts the double learning theory proposed by Argyris and Schön [15], which pertains to learning to change underlying values and assumptions, as the theoretical framework to support the system and experimental design. The single-loop reflection refers to increase efficiency of reaching an objective. It is task oriented and about the design of the process to retain reliability. It is simple reflection that may challenge assumptions and strategies to alter the plan of action but always 'in ways that leave the values of a theory of action unchanged' [16]. Comparatively, the double-loop reflection is described by Courtney et al. [17] as a higher level of reflection than single-loop reflection. This second loop focuses on the examination and reflection of the theory or perspective in use [18] or the evaluation of an experience using explicit and varied concepts [19].

Considering from the perspective of double loop learning theory, we design question prompts into the inquisitive simulated tutee environment to elicit two major types of reflection for students in tutoring.

- **Generic Prompts eliciting Double-Loop Reflection** lead students to examine their perspectives, assumptions and experiences by reflecting on metacognitive strategies and beliefs in learning and teaching. Sample generic prompts are: "Before starting to teach, can you think about what you are supposed to learn from it?", "Can you reread the learning objectives and resources and ask if the map really meets the description in the learning objectives and resources?"
- **Specific Prompts eliciting Single-Loop Reflection** lead students to reach certain learning objectives by reflecting on task-specific and domain-related skills regarding their activities and to articulate their explanatory responses. Sample specific prompts are: "Can you explain the concepts you just taught me?", "Can you tell me if my reasoning process is correct and give me a further explanation?"

2.4 Self-Efficacy Levels

The double-loop learning process that appears in the self-efficacy model has been found to occur in individual knowledge sharing activities and could positively affect performance [20]. Zimmerman [21] notes that self-efficacy has emerged as a highly effective predictor of students' motivation and learning. Bandura [22] notes that high self-efficacy in one's ability to share tacit knowledge may result in challenging personal goals, as well as higher effort, persistence, satisfaction, and performance. These positive outcomes could fuel the self-beliefs that one can perform even better when self-efficacy is estimated again [23].

3 Empirical Study

3.1 Participants and Procedure

The goal of this study is to investigate whether the proposed framework is able to help learners in reflection and learning. Such a framework was implemented in a simulated tutee system developed and described in our previous studies [24, 25].

Participants were 29 students from two local secondary schools (ages ranged from 13 to 15) who took part in the experiments on a voluntary basis for two two-hour sessions within one week (Table 1). They were randomly assigned to one of the three conditions to study elementary economics topics of demand and supply. Economics is both a theoretical and applied domain, seldom studied in class by secondary school students and seldom adopted as the domain in ILE research. The domain materials were provided to participants before the sessions.

Table 1. Procedure of Empirical Study

Phases	Activities	Description
Phase 1	Pre-test	MSLQ and Knowledge Pre-test
Phase 2.1	Tutoring: Familiarization	Get familiar with materials and simulated tutee
Phase 2.2	Tutoring: Production	Teach simulated tutee by concept mapping
Phase 2.3	Tutoring: Evaluation	Check the performance of simulated tutee
Phase 2.4	Tutoring: Post-Task Reflection	Reflect upon own performance
Phase 3	Post-test	MSLQ and Knowledge Post-test (1 week later)

During the tutoring phases, participants were working with the simulated tutee system to teach what they learn from materials by constructing concept maps. The NP group (n=10) worked with the basic version of simulated tutee without prompts. The SP group (n=10) worked with the version embedded with specific prompts. The GP group (n=9) worked with the version embedded with generic prompts. Both the SP and GP groups were required to write down their reflection statements in the dialog window to respond to the simulated tutee prompts to proceed with their tutoring activities.

Sample response statements from participants to two types of prompts are as follows.

…

[Simulated tutee detects decreasing of missing expert propositions in the production phase] Can you pick up some concepts and explain to me the relationship among them? (**Specific Prompts**)

[SP Student] According to law of demand, the higher the price of the product, the fewer amounts of people will consume this product. According to law of supply, the higher the price the higher is the quantity supplied.

…

[Simulated tutee detects start of the post-reflection phase] What is your thinking after teaching me? **(Generic Prompts)**

[GP Student] You are a curious student by asking a lot of questions to me. But sometimes, I don't quite understand what you are asking me to do. I need to learn more about demand and supply to teach you better.

…

We further categorized the participants into *High* and *Low* group as to their MSLQ (Motivated Strategies for Learning Questionnaire, [26]) pre- and post- test scores (Table 1). Participants scored above the mean score in MSLQ pre-test were included in the High group and the rest were included in the Low group.

3.2 Effects of Question Prompts on Development of Self-Efficacy

To test students' development of self-efficacy, we compared the students' pre-to-post scores in the pretest on self-efficacy. The data is reported in Table 2. A Tukey's test, performed to compare the difference between groups, reveals that there is a significant difference between the GP Low group and the Control Low group (*MD* = 21.75, *p* = 0.003), the GP High group and the Control Low group (*MD* = 22.40, *p* = 0.001), the SP High group and the Control Low group (*MD* = 16.40, *p* = .024). There was no significant difference between the GP Low group and the SP Low group or between the SP Low group and the Control Low group. This result suggests the GP Low group has experienced most prominent progress in than other groups.

Table 2. Result of MSLQ Pre-/Post- Test

Groups	N	SE Pre-test (Mean/SD)	SE Post-test (Mean/SD)
Control Low	5	23.00 (10.36)	27.00 (10.36)
Control High	5	43.89 (7.08)	45.00 (8.00)
GP Low	4	20.50 (4.20)	48.75 (4.78)
GP High	5	47.00 (7.00)	49.40 (4.87)
SP Low	5	25.40 (8.93)	35.40 (8.82)
SP High	5	45.20 (8.29)	43.40 (6.07)

Table 3. Result of Domain Knowledge Pre-/Post- Test

Groups	N	Pre-test (Mean/SD)	Post-test (Mean/SD)
GP High	5	4.25 (0.96)	8.25 (0.96)
GP Low	4	3.20 (0.84)	7.00 (1.41)
SP High	5	4.00 (1.58)	7.60 (2.30)
SP Low	5	3.80 (0.83)	7.40 (1.34)

3.3 Effects of Self-Efficacy on Adoption of Question Prompts

Table 3 shows the result of domain knowledge pretest to posttest. A Games-Howell test, which is to eliminate the uncertainty of equivalent population variances, indicated that there is a significant difference between the GP High group and Control Low group (*MD* = 0.013, *p* = 0.013), which implies that students high self-efficacy who received generic prompts have significantly outperformed the students with low-efficacy in the control group. The result indicates that students with high self-efficacy might benefit more from generic prompts than others.

3.4 Effects of Question Prompts on Elicited Levels of Reflection

A tripartite coding scheme [27], which includes *reaction*, *elaboration*, and *contemplation*, was adopted to analyze participant' response statements to simulated tutees' prompts to indicate their elicited levels of reflection.

An initial observation of the transcript excerpts of prompt-response exchange showed that the GP participants exhibited a higher level of reflection than the SP participants. We noted that the GP participants gradually tended to present more completive response statements than the SP participants after being exposed to the question prompts for a period of time. A sample transcript of participants' response statement is as follows to demonstrate both SP participant and GP participant responded to the purposefully designed same question prompts but showed different levels of reflection in their response statements.

[Simulated Tutee] What is the most important thing you tried to teach me?
[SP Participant] What is economics? **(Reaction: Report facts)**
[GP Participant] The universality of these links, causes and implications
(Contemplation: Indicate a shift in thought).

[Simulated Tutee] What is/are the most important thing(s) you learn from me?
[SP Participant] Demand, supply and price are inter-related and contribute to the idea of economics in total. **(Elaboration: Provide interpretive analysis of course concepts)**
[GP Participant] What to do when approaching economics in real life.
(Contemplation: Indicate a shift in attitude about himself)

[Simulated Tutee] Did you learn something from me?
[SP Participant] Yes, I learnt how to organize my thoughts better.
(Contemplation: Indicate a shift in attitude about himself)
[GP Participant] How to teach better and more clearly. **(Contemplation: Indicate a shift in attitude about himself).**

A combined qualitative and quantitative analysis of participants' response statements to simulated tutees' question prompts showed the difference in the levels of reflection between groups (Table 4). An ANOVA test shows significant difference between the groups as to reactive statements (F (1, 17) = 36.747, p <.05) and contemplative statements (F (1, 17) =19.472, p < .05). The number of elaborative statements was not significantly different between the groups. Such a result shows that the participant of

GP group, whether with high or low efficacy, was more likely to respond with contemplative statements representing a higher level of reflection. Comparatively, the participants of SP group, whether with high or low efficacy, responded more with reactive statements representing a lower level of reflection which means they pay more attention to report issues with no development than the GP group.

Table 4. Result of Response Statements Analysis

Levels of Reflection	GP	SP	ANOVA-Test[1]
Reaction	6.00 (1.41)	9.60(1.17)	36.75*
Elaboration	7.56 (1.54)	9.50(2.17)	4.19
Contemplation	9.00 (1.80)	5.10 (2.82)	19.47*

[1], p < .05

4 Conclusion and Future Work

Overall, the preliminary results indicate that the proposed reflective tutoring framework, with consideration of tutoring stages, reflection types and self-efficacy levels, has the potential to help us systematically understand learner's reflection and learning when interacting with a simulated tutee environment and further exploit the potential of question prompts. In future studies, we will further work on the development of simulated tutee systems that encourage students to do both reflection-in-action and reflection-on-action and incorporate them into regular classroom use.

References

1. Rothkopf, E.: Learning from written instructive materials: An Exploration of the control of inspection by test-like events. American Educational Research Journal 3, 241–249 (1966)
2. Chi, M.T.H., et al.: Eliciting self-explanations improves understanding. Cognitive Science 18, 39–477 (1994)
3. Davis, E.A.: Scaffolding students' reflection for science learning. University of California, Berkeley (1998)
4. Moon, J.: A Handbook of Reflective and Experiential Learning. Routledge, London (2004)
5. Hmelo, C., Day, R.: Contextualized questioning to scaffold learning from simulations. Computers & Education 32, 151–164 (1999)
6. Azevedo, R., Hadwin, A.F.: Scaffolding self-regulated learning and metacognition - Implications for the design of computer-based scaffolds. Instructional Science 33, 367–379 (2005)
7. Davis, E.A., Linn, M.: Scaffolding students' knowledge integration: Prompts for reflection in KIE. International Journal of Science Education 22(8), 819–837 (2000)
8. Ge, X., Land, S.M.: A conceptual framework of scaffolding ill-structured problem solving processes using question prompts and peer interactions. Educational Technology Research and Development 52(2), 5–27 (2004)
9. Hoffmana, B., Spatariu, A.: The influence of self-efficacy and metacognitive prompting on math problem-solving efficiency. Contemporary Educational Psychology 33(4), 875–893 (2008)

10. Cohen, J.: Theoretical considerations of peer tutoring. Psychology in the Schools 23, 175–186 (1986)
11. Gartner, A., Kohler, M., Riessman, F.: Children teach children: Learning by teaching. Harper & Row, New York (1971)
12. Roscoe, R.D., Chi, M.T.H.: Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors' explanations and questions. Review of Educational Research 77(4), 534–574 (2007)
13. Amulya, J.: What is Reflective Practice? Center for Reflective Community Practice, Massachusetts Institute of Technology, Boston (2004)
14. Gama, C.A.: Integrating Metacognition Instruction in Interactive Learning Environments. University of Sussex (2004)
15. Argyris, C., Schön, D.: Organizational learning II: Theory, method and practice, in Reading. Addison Wesley, Mass (1996)
16. Brockbank, A., McGill, I.: Facilitating Reflective Learning in Higher Education, in Society for Research into Higher Education. Open University Press, Buckingham (1988)
17. Courtney, J., Croasdell, D., Paraadice, D.: Inquiring Organisations. Australian Journal of Information Systems 6(1) (1998)
18. Lynch, M., Joham, C.: Reflection in Self-organised Systems, Information Systems Foundations: Constructing and Criticising. In: ISF Proceedings. ANU, Canberra (2004)
19. Lynch, M., Metcalfe, M.: Reflection, Pragmatism, Concets and Intuition. Journal of Information Technology Theory and Application 7(4), 1–10 (2006)
20. Jashapara, A.: Cognition, culture and competition: an empirical test of the learning organization. The Learning Organization 10(1), 31–50 (2003)
21. Zimmerman, B.J.: Self-Efficacy: An Essential Motive to Learn. Contemporary Educational Psychology 25(1), 82–91 (2000)
22. Bandura, A.: Self-efficacy: The exercise of control. Freeman, New York (1997)
23. Endres, M.L., et al.: Tacit knowledge sharing, self-efficacy theory, and application to the Open Source community. Journal of Knowledge Management 11(3), 92–103 (2007)
24. Wu, L., Looi, C.-K.: Econie: An Inquisitive Virtual TuteePrompting Student's Reflection in Tutoring. In: Kong, S.C., et al. (eds.) Proceedings of the 17th International Conference on Computers in Education, pp. 35–42. Asia-Pacific Society for Computers in Education, Hong Kong (2009)
25. Wu, L., Looi, C.-K.: Use of agent prompts to support reflective interaction in a learning-by-teaching environment. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 302–311. Springer, Heidelberg (2008)
26. Pintrich, P.R., De Groot, E.: Motivational and self-regulated learning components of classroom academic performance. Journal of Educational Psychology 82(1), 33–50 (1990)
27. Ortiz, J.: Reflective Practice and Student Learning in the Introductory Interpersonal Communication Course. Maricopa Institute for Learning (2006)

A Simple Method of Representing Reusable Strategic Knowledge for MT Tutoring

Amir Abdessemed and André Mayers

Université de Sherbrooke, Québec, Canada
{amir.abdessemed,Andre.Mayers}@USherbrooke.ca

Abstract. Several popular model-tracing tutors are available today, but they are usually limited by common general compromises, such as (1) between the complexity and the variety of the domains taught, or (2) between the precision of student modeling and the degree of easiness to model new domains. We present an improved version of the ASTUS formalism, allowing the modeling of more complex knowledge while simultaneously increasing the diversity of pedagogical behavior, with no loss in diversity or time consumption.

Keywords: knowledge representation, model-tracing tutors, strategies.

1 Introduction

Knowledge representation is an essential element of any model-tracing tutor design. A good number of model tracing tutors, such the Cognitive Tutors [1] use a knowledge representation involving production rules. We tend to question the extensive use of cognitive theories, and therefore production rules, in knowledge representation for ITS. We work under the assumption that the design of representation formalisms must primarily focus on the way a teacher depicts and transmits the knowledge to his students, focusing on the way the knowledge is stored in students' minds being secondary. ASTUS [2] was created in that spirit.

In our experience of modeling complex domains, we encountered some properties that couldn't be easily represented. Consider a domain where there is usually more than a solution, but each step of each solution might be correct on its own. It is the choices of the steps, which may be resulting of the use of a higher level of knowledge, that make a solution as more or less optimal. The idea behind our proposed approach is to separate procedural errors, which will generally result in an erroneous step, from strategic errors, which produce a correct step leading to a non-optimal solution. We aimed to separate the modeling of procedural and higher-level thinking. The task of using an interface to add a sequence of natural numbers is an excellent introductory example of this approach, as adding numbers of a list two by two will lead to the completion of the addition task in all cases, but different solutions will be characterized by different higher-level thinking (strategies).

2 Representation of Strategies

Common model tracing tutors tend not to distinguish between procedural and strategic knowledge, resulting in one strategic approach being taught implicitly. On

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 411–413, 2011.

production rule based tutors, the strategies employed are implicitly encoded in the priority levels assigned to the production. Researches showed that although the explicit teaching of strategies didn't raise the scores of tested students, it improved their understanding and their ability to justify and explain their answers [3].

In ASTUS, knowledge is depicted in the form of goals and procedures [2]. Goals represent intentions, and procedures symbolize the ways to achieve them. Procedures can be classified into primitive procedures, which represent actions on the interface, and complex procedures, which depict scripts of new sub-goals. This method of representation has the property of providing easy access to the elements of procedural knowledge where choices are made. Until recently, there was no mechanisms influencing these choices in ASTUS, let alone giving these mechanisms some form of pedagogical meaning. We designed an improved version of ASTUS, allowing the explicit representation of strategies and the use of the latter to influence the solving.

3 A New Formalism of Strategy Representation

We define a strategy as "semantic knowledge, based on a partial observation of the state of the problem, that influence the solving process of a problem when a decision (a choice) is to be made". It influences the decision by selecting the most appropriate candidates (goals or procedures) after examining their relevant properties (metadata). In our addition lab, a strategy could be "to concentrate on multiples of 5".

Strategies can be pertinent or not, in the sense that their application is considered useful or not. For example, the strategy of adding multiples of 5 might cease to be pertinent when dealing with large numbers. Strategies can also be effective or not. For example, the same strategy is not effective if there are no multiples of 5 to add. Since the primitive procedures are usually the result of several decisions on the solution tree, each step will be characterized by the strategies that influenced the decisions.

On any decision during the solving process, all pertinent strategies may be applied to ensure a selection of candidates. The fact that several strategies led to the selection of an action ensures a stronger justification and allows richer pedagogical help.

Our formalism allows the representation of complex strategies. For example, the complex strategy of "concentrating on multiples of 5" has two substrategies: "use existing multiples of 5" and "create new multiples of 5". The ability to construct complex and abstract strategies and the fact that strategic and procedural knowledge are separated allows the reuse of the strategies across different domains.

The introduction of a new strategy and the related pedagogical material into the knowledge of a domain is time consuming. If we consider that the representation of a new element (be it strategy, goal or procedure) and its pedagogical material takes n time units using the proposed formalism, it would require $2^n m$ time units using a purely procedural approach to represent a strategy.

4 Validation Experiments

The experiments were undertaken with the help of sixteen students of the Ecole Polytechnique de Montreal, deliberately chosen from different levels.

First, students were asked to produce a list of “methods” that they would use to add a sequence of natural numbers in the most optimal way, with the aim of representing them later as strategies. The goal of this phase was to test to what extent “rough knowledge” could be represented in terms of strategies, and it was a 100% success.

During the second phase, the students were given the established list of strategies, asked to study it, and then invited to solve a number of addition problems. During these activities, scores related to the correct use of strategies were issued. The goal of the phase was to determine (1) if the students found the scores fair regarding their performance, and (2) if a progression in the score could be noticed. The results varied greatly, the main factor being the number of occurrences a strategy could be applied in a solving session. For strategies with frequent occasions of being applied, the scores were sound. A graph of the evolution of the score showed a clear progression. For rarely applicable strategies, the progress was too erratic due to the lack of data.

5 Discussion

The modeling phase of the experiments showed excellent results and confirm that strategies can be used to depict knowledge of domains that allow multiple solutions. The second phase provide much less perfect results, as we could observe that efficient model tracing require a sufficient amount of occurrences for the strategy to be used. The use of strategies as knowledge units remains very advantageous, provided that some requirements are met. For example, a careful choice of problems will ensure that the taught strategies will have a sufficient number of occasions to be applied.

6 Conclusion

In this paper, we described a simple way to represent reusable strategic knowledge for model tracing tutoring. We described some of the experiments leading to its validation as a teachable detectable knowledge unit.

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. Paquette, L., Lebeau, J.F., Mayers, A.: Authoring Problem-Solving Tutors: A Comparison Between ASTUS and CTAT. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) Advances in Intelligent Tutoring Systems, pp. 377–405. Springer, Heidelberg (2010)
3. VanLehn, K., et al.: Implicit versus explicit learning of strategies in a non-procedural cognitive skill. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 521–530. Springer, Heidelberg (2004)

COMET: Context Ontology for Mobile Education Technology

Sohaib Ahmed and David Parsons

Institute of Information & Mathematical Sciences, Massey University,
Auckland, New Zealand
{s.ahmed,d.p.parsons}@massey.ac.nz

Abstract. The use of mobile devices is increasingly prevalent in education. These devices provide the convenience of supporting access to learning anytime, anywhere. Further, mobile learning provides opportunities to tailor the learning experience to dynamically changing contexts. Major challenges for constructing context-aware models to support this kind of learning include defining the contextual information and adapting to dynamic changes. Ontology-based context models exhibit features such as expressiveness, extensibility, ease of sharing, and logic reasoning support, thus show promise in this area. In this paper, we propose COMET (Context Ontology for Mobile Education Technology) in order to provide a semantically rich model for mobile learning. More specifically, we have demonstrated an example application to show how we can retrieve contextual data from different participating entities within the ontology by using their semantic understanding.

Keywords: Mobile learning, Context, Ontologies.

1 Classifying Context Information

Context can be viewed from different perspectives but there is no definite agreement about what should be modeled in the area of context. Most previous work on context in mobile computing focuses on a common core that includes environment and human dimensions [1]. Kurti et al. [2] suggested a conceptual framework in which activity is one of the three dimensions of context including environment and personal dimensions. Our approach to context modeling builds on their work.

The development of context-aware applications deals with a number of technological challenges and requires the existence of a suitable context model that can be represented and understood between different entities like devices and applications. Some other context-aware systems have been developed in terms of device adaptability including tourist applications [3] and Innsbruck. Mobile [4]. In both applications, adaptation is used in a single direction, from resource to device, as contents are adapted according to different device types. However, such uni-directional transformations do not fully explore the application of context-aware systems that adapt from multiple perspectives. For comprehensive adaptivity support, we need an approach which can deliver adaptive contents from any platform, in any format, to any device, through any network, at anytime, anywhere [5].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 414–416, 2011.

In recent years, ontologies have emerged as one of the most popular and widely accepted tools for modeling contextual information in mobile computing domains [6]. Several context ontologies have been proposed but thus far they have not been able to capture all the relevant information needed for technology enhanced mobile education. We have developed the COMET context ontology based on three key concepts [2]. The purpose of defining such ontology is to demonstrate how different entities can be inter-related and used in order to extract specific information in a mobile learning environment.

2 Usage Scenario

As a proof of concept, we consider two key scenarios; one in which an educator wishes to identify suitable mobile applications for their students according to the availability of specific mobile devices, and another in which she wishes to identify devices that can run a chosen application. We have taken 11 recent mobile applications and around 60 mobile devices and their supported versions with some other related information to show how they are semantically inter-connected (fig. 1). To test the utility of our ontology we built a prototype ontology-driven web application that demonstrates information retrieval using SPARQL queries from multiple perspectives.

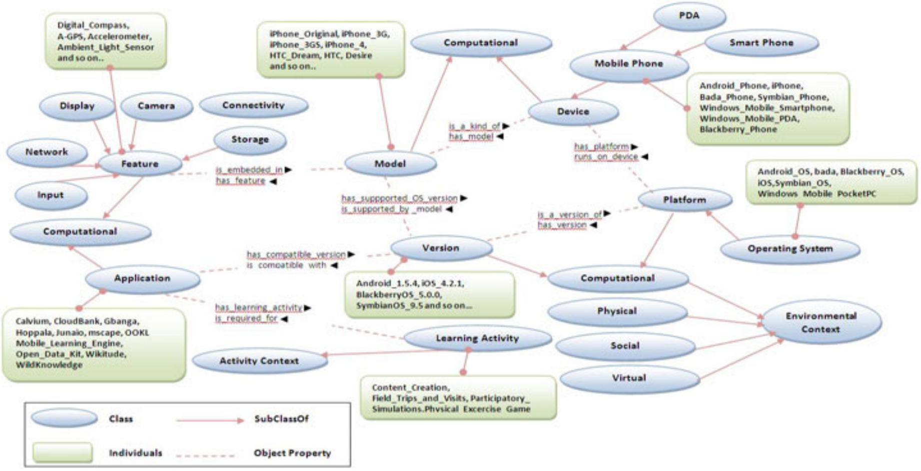
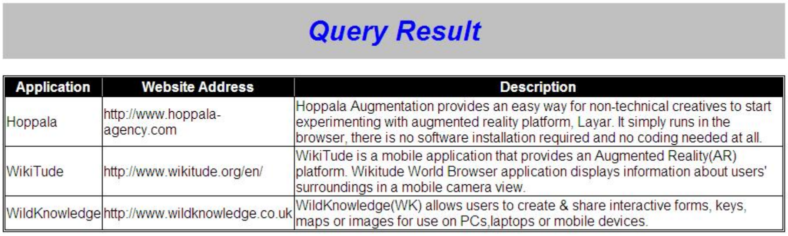

Digital_Compass, A-GPS, Accelerometer, Ambient_Light_Sensor and so on..
iPhone_Original, iPhone_3G, iPhone_3GS, iPhone_4, HTC_Dream, HTC, Desire and so on..
Computational
PDA
Smart Phone
Mobile Phone
Android_Phone, iPhone, Bada_Phone, Symbian_Phone, Windows_Mobile_Smartphone, Windows_Mobile_PDA, Blackberry_Phone
Display
Camera
Connectivity
Device
Storage
is_a_kind_of ▶
has_model ◀
has_platform ▶
runs_on_device ◀
Network
Feature
is_embedded_in ▶
has_feature ◀
Model
Android_OS, bada, Blackberry_OS, iOS,Symbian_OS, Windows Mobile PocketPC
Input
has_suppported_OS_version ▶
is_supported_by _model ◀
Platform
is_a_version_of ▶
has_version ◀
Computational
Version
Operating System
has_compatible_version ▶
is compatible with ◀
Application
Android_1.5.4, iOS_4.2.1, BlackberryOS_5.0.0, SymbianOS_9.5 and so on...
Computational
Physical
has_learning_activity ▶
is_required_for ◀
Calvium, CloudBank, Gbanga, Hoppala, Junaio, mscape, OOKL Mobile_Learning_Engine, Open_Data_Kit, Wikitude, WildKnowledge
Environmental Context
Learning Activity
Social
Activity Context
Virtual
Content_Creation, Field_Trips_and_Visits, Participatory_ Simulations.Phvsical Excercise Game
Class
SubClassOf
Individuals
Object Property

Fig. 1. Excerpt from COMET

For instance, if we want to extract information such as a list of mobile devices which can support a particular application (e.g. Hoppala) or list of applications which can run on a particular mobile model (e.g. Android Phone) to support a given learning activity (e.g. Field Trips and Visits), these defined relationships between entities can help us to extract the relevant information from multiple complementary perspectives (e.g. fig. 2).



Query Result

Application	Website Address	Description
Hoppala	http://www.hoppala-agency.com	Hoppala Augmentation provides an easy way for non-technical creatives to start experimenting with augmented reality platform, Layar. It simply runs in the browser, there is no software installation required and no coding needed at all.
WikiTude	http://www.wikitude.org/en/	WikiTude is a mobile application that provides an Augmented Reality(AR) platform. Wikitude World Browser application displays information about users' surroundings in a mobile camera view.
WildKnowledge	http://www.wildknowledge.co.uk	WildKnowledge(WK) allows users to create & share interactive forms, keys, maps or images for use on PCs,laptops or mobile devices.

Fig. 2. Partial query result from one perspective of the ontology

3 Conclusion and Future Work

In this paper, we have discussed the need for an underlying context model for mobile education technology which we provide in the form of COMET. The work presented here is still in early stages. We are currently working on the design of context and domain ontologies. In future, we may leverage these ontologies to develop an adaptive learning environment. Further, adaptation of the learning contents may be explored by using more real life scenarios. That might help us to understand how ontology-driven applications can possess the necessary flexibility to support mobile learner activities in varying contexts.

References

1. Brusilovsky, P., Millan, E.: User Models for Adaptive Hypermedia and Adaptive Educational Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web 2007. LNCS, pp. 3–53. Springer, Heidelberg (2007)
2. Kurti, A., Spikol, D., Milrad, M.: Bridging outdoors and indoors educational activities in schools with the support of mobile and positioning technologies. Int. J. of Mob. Learning and Organisation 2(2), 166–186 (2008)
3. Mantovaneli Pessoa, R., Zardo Calvi, C., Pereira Filho, J.G., Guareis de Farias, C.R., Neisse, R.: Semantic Context Reasoning Using Ontology Based Models. In: Pras, A., van Sinderen, M. (eds.) EUNICE 2007. LNCS, vol. 4606, pp. 44–51. Springer, Heidelberg (2007)
4. Hopken, W., Scheuringer, M., Linke, D., Fuchs, M.: Context-based Adaptation of Ubiquitous Web Applications in Tourism. LNCS, pp. 533–544. Springer, New York (2008)
5. Yang, S.J.: Context-aware Ubiquitous Learning Environments for peer-to-peer Collaborative Learning. J. of Edu. Tech. & Soc. 9(1), 188–201 (2006)
6. Ye, J., Coyle, L., Dobson, S., Nixon, P.: Ontology-based Models in Pervasive Computing Systems. The Know. Engg. Rev. 22(4), 315–347 (2007)

Scaffolding to Support Learning of Ecology in Simulation Environments

Satabdi Basu, Gautam Biswas, and Pratim Sengupta

Vanderbilt University, Nashville, TN, USA
{satabdi.basu,gautam.biswas,pratim.sengupta}@vanderbilt.edu

Abstract. This paper presents a semi-clinical interview-based empirical study for identifying effective scaffolds to support inquiry learning in a Multi-Agent based simulation of a desert ecosystem. Our preliminary results based on Sherin et al.'s Δ-shift framework show that all five categories of identified scaffolds contributed to students' conceptual shifts and overall learning gains. This paper lays the foundation for future research on designing scaffolds in multi-agent, simulation-based learning environments for study of ecological processes.

Keywords: Inquiry Learning, Simulation-based Learning, Scaffolding, Multi-Agent Simulations, Conceptual Change.

1 Introduction

Students at all levels perceive ecology as a difficult subject to learn – in particular the concepts of population and population frequencies, organization in an ecosystem, and the relationship between individuals, populations and species [3][5]. Multi-Agent-Based-Models (MABMs) have been successful in teaching ecological concepts to novices [4][5]. Rather than describing relationships between properties of populations, MABMs require students to focus on individuals and their interactions [4], thereby engaging in intuitive "agent-level thinking" (i.e., thinking about the actions and behavior of individual actors in the ecosystem). In this paper, our emphasis is on scaffolding in MABM-based learning environments, a topic that has received significantly less attention, but is essential to support novice learning [1][2].

Scaffolding is particularly important in inquiry based learning environments where learners tend to face a multitude of problems ranging from generating hypotheses, setting up experiments, to interpreting simulation results for inferring the underlying models [1]. Quintana et al. describe a set of scaffolding guidelines and strategies organized around the three primary components of scientific inquiry: sense making, process management, and articulation and reflection [1]. Building on Quintana et al, we identify categories of scaffolds to help middle school students learn about the interactions between entities in an ecosystem. We analyze the effectiveness of our scaffolds using Sherin et al.'s Δ-shift framework [2]. A scaffolding analysis in this framework is defined as a comparison of an unassisted situation S_{base} and a scaffolded situation S_{scaf}. Δs, the difference between S_{scaf} and S_{base}, can be provided by teachers, software agents, and other tools. The change in performance (P) due to Δs is

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 417–419, 2011.

calculated as $\Delta p = P_{scaf} - P_{base}$. P_{target} is defined as an idealized target performance, and the goal of the scaffold Δs is to make P_{scaf} match P_{target}.

2 Method

The model used for this study (Figure 1) was a Netlogo based simulation [4] of a Saguaran desert ecosystem containing five species: two plants (ironwood trees and cacti), their fruits (pods) and seeds, and three animals (rats, doves and hawks). The five species are characterized by sets of simple rules that define their behavior and their interactions with other species in the environment. Besides the simulation, a set of graphs display the aggregate population for each species over time. Learners manipulate a set of sliders to regulate the initial number of each species, and they can start, stop or regulate the speed of a simulation run at any point.

We report an interview-based study conducted with 7th and 8th grade students (n = 10 in each grade), uniformly distributed by achievement profile. The experimenter conducted semi-clinical interviews with each student by periodically asking students for mechanistic explanations of their observations and predictions. Additionally, she also verbally guided dialog segments to scaffold students' reasoning, wherever necessary, to address their difficulties. Each interview lasted about 35 minutes, and was video-recorded and later transcribed for analysis.

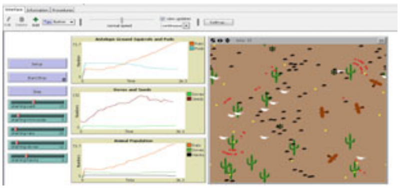


Fig. 1. The user interface (UI) of the Saguaran desert ecosystem simulation environment

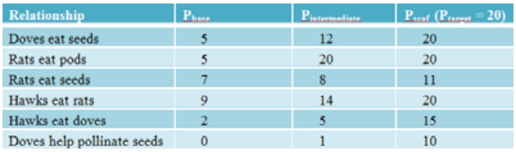
3 Results and Conclusion

The five categories of scaffolds we identified to help students overcome difficulties are described below (The numbers in parentheses indicate the number of students who needed the type of scaffold): *S1. Scaffolds for setting up a simulation run* (18) through prompts for choosing initial population parameters, regulating the speed of the simulation, deciding how long to observe, which set of species to observe, etc; *S2. Scaffolds for interpreting results of a simulation run* by prompting to notice the plotted graphs, relating them with the simulation window, and drawing conclusions about the interrelatedness of the species involved; *S3. Scaffolds for controlling variables and planning the construction of the underlying model of the simulation* by suggesting a vary-one-variable-at-a-time and/or vary-one-pair-at-a-time approach to study relationships between different pair of variables/species, deciding the ordering for such studies, and keeping track of which pairs have been studied and what relationships have been found; *S4. Scaffolds through self-explanations and predictions (20)* by posing general and directed queries and asking the student to make predictions about simulation results; *S5. Scaffolding by creating cognitive conflict* (20) by reminding students about previous contradictory findings or statements made, or by making them re-run simulations with different parameters.

Most students required a combination of scaffolds to interpret and understand the relations between the species modeled in the simulation. The change in performance due to Δs (S1 though S5) is $\Delta p = P_{scaf} - P_{base}$, where P_{base} and P_{scaf} describe performances at the 'Initial Ideas' phase and at the end of the scaffolding phase, respectively. Initially, only general scaffolds S3 and S4 were provided which were independent of the relations being scaffolded. The performance at this stage is referred to as $P_{intermediate}$. Later a combination of S1 through S5 was administered. It was noticed that the average number of correct relationships contained in students' responses increased from 1.4 in the 'Initial Ideas' phase to 3 in the 'Intermediate' phase and 4.8 at the end of the 'Scaffolding' phase (Δp=3.4). The effects of the scaffolds on number of students who could find each relationship have also been summarized in Table 1.

In conclusion, we have identified five categories of scaffolds required in inquiry learning involving MABMs, and shown their effectiveness using Sherin et al's Δ-shift framework [2]. As we move forward, we envision designing a learning environment using such MABM simulations along with the necessary set of scaffolds.

Table 1. Effect of scaffolds on number of students who could find each relationship



Relationship	P_{base}	$P_{intermediate}$	P_{scaf} (P_{target} = 20)
Doves eat seeds	5	12	20
Rats eat pods	5	20	20
Rats eat seeds	7	8	11
Hawks eat rats	9	14	20
Hawks eat doves	2	5	15
Doves help pollinate seeds	0	1	10

Acknowledgements

Satabdi Basu and Gautam Biswas were supported by NSF IIS Award #0904387. Pratim Sengupta was supported by NSF DRL Award #1020101.

References

1. Quintana, C., et al.: A Scaffolding Design Framework for Software to Support Science Inquiry. Journal of the Learning Sciences 13(3), 337–386 (2004)
2. Sherin, B., Reiser, B.J., Edelson, D.: Scaffolding Analysis: Extending the Scaffolding Metaphor to Learning Artifacts. Journal of the Learning Sciences 13(3), 387–421 (2004)
3. Chi, M.T.H., Ferrari, M.: The nature of naïve explanations in natural selection. International Journal of Science Education 20(10), 1231–1256 (1998)
4. Wilensky, U., Resiman, K.: Thinking like a wolf, a sheep, or a firefly: learning biology through constructing and testing computational theories – an embodied modeling approach. Cognition and Instruction 24(2), 171–209 (2006)
5. Dickes, A., Sengupta, P.: Learning Natural Selection in 4th Grade With Multi-Agent-Based Computational Models. In: Sengupta, P., Hall, R. (eds.) Models, Modeling, and Naïve Intuitive Knowledge in Science Learning. Symposium Presented at the 41st Annual Meeting of the Jean Piaget Society, Berkeley, CA (2011)

Context-Dependent Help for the DynaLearn Modelling and Simulation Workbench

Wouter Beek, Bert Bredeweg, and Sander Latour

University of Amsterdam, The Netherlands
{beek,bbredeweg,latour}@uva.nl

Abstract. We implemented three kinds of context-dependent help for a qualitative modelling and simulation workbench called DynaLearn. We show that it is possible to generate and select assistance knowledge based on the current model, simulation results and workbench state.

Keywords: Qualitative reasoning, support knowledge, help systems.

1 Introduction

DynaLearn is a conceptual modelling workbench that allows learners to build and simulate causal models [1]. It is based on Qualitative Reasoning (QR) and provides a domain-independent and formal means to externalize thought, capturing the learner's believes of how and why a system behaves. Since the modelling language is very powerful, it introduces a host of concepts and tools [2], resulting in a steep learning curve for learners. In order to support these learners in their modelling attempt, we have implemented three kinds of context-sensitive support facilities.

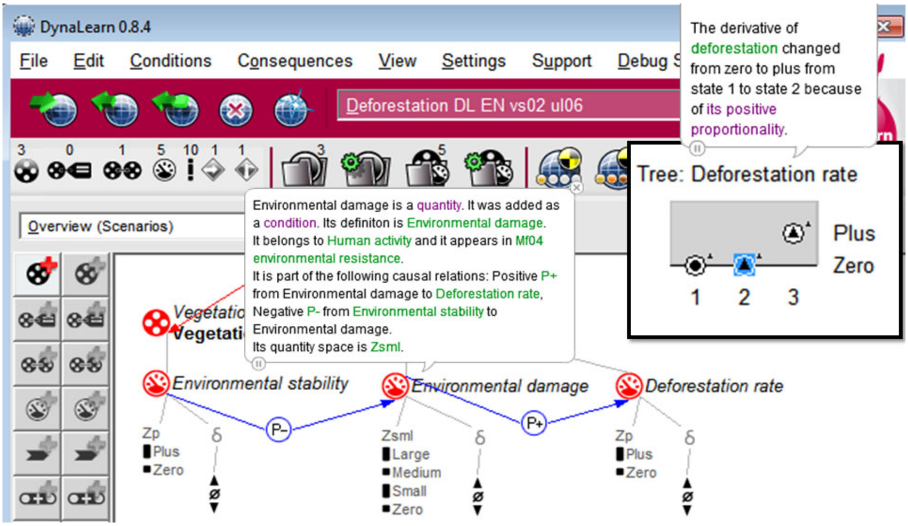
2 Help Modes

Figure 1 gives an impression of the support features within the DynaLearn interface. It shows a sample model and simulation results. The left hand-side text balloon explains the properties of the selected model ingredient *Environmental damage*. The right hand-side text balloon explains why the selected value change (i.e. *Environmental damage*'s decrease) occurred.

In DynaLearn there are three modes of basic help. The "What is"-mode gives information about the learner-created model, explaining for every model ingredient what it is in QR terms (the conceptual modelling language), what its properties are, and what the relations in which it partakes. The "Why"-mode gives information about simulation results, explaining why a certain change (or event) occurred in terms of the causal model. The "How to"-mode explains how to perform modelling tasks within the workbench.

The models and simulation results change as the learner edits them. Meanwhile the applicable help requests are continuously generated in the form of a hierarchically structured menu, from which the learner can choose. Within each help message

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 420–422, 2011.



The derivative of deforestation changed from zero to plus from state 1 to state 2 because of its positive proportionality.

Environmental damage is a quantity. It was added as a condition. Its definiton is Environmental damage. It belongs to Human activity and it appears in Mf04 environmental resistance. It is part of the following causal relations: Positive P+ from Environmental damage to Deforestation rate, Negative P- from Environmental stability to Environmental damage. Its quantity space is Zsml.

Fig. 1. A portion of the DynaLearn interface, showing a model and simulation results. The text balloons show help messages for the model ingredient *Environmental damage* (left) and for *Environmental damage*'s decrease (right).

dynamically generated follow-up links (as in [3]) are added that allow additional help requests into more detailed or related information.

3 Implementation

The help information is captured in ontological descriptions of the model (What is), the simulation results (Why) or the workbench tasks (How to). Generating the answer to a question amounts to filtering the relevant support knowledge from these descriptions. The generated answers are themselves mini RDF-documents, allowing for easy natural language generation.

The purpose of the three help modes is to give local information within a global context. This means that individual help messages are concise and to-the-point, covering the aspects of an individual model, simulation or application element. At the same time, the embedding of the individual element within a broader context is included by providing possibilities for posing follow-up questions. All knowledge can be reached by allowing the learner to traverse the graph of interconnected model ingredients (What is), prior causes (Why) and related tasks (How to). The three support modes also link to each other (in figure 1 the green links in the right hand-side text balloon allow "What is"-requests to be posed from within a "Why"-item).

If the learner issues a "What is"- or "Why"-request, the appropriate information is generated on the fly, based either on the constructed model or on the simulation results. The "How to"-representation is itself a static ontology of tasks, but which task requests are displayed to the learner depends on the state that the workbench is in. Each model construction task consists of a sequence of subtasks with set preconditions (what allows a task to be performed) and postconditions (what is brought about by performing a task). Only tasks that can be performed are shown, and only subtasks that are not yet performed are included in the help messages. A subtask is communicated once the learner has satisfied its preconditions. Performing a subtask brings about its postconditions, potentially triggering new preconditions, etc. In this way the right task information is communicated at precisely the right moment.

All support messages are communicated by a virtual teacher character that uses speech, text, gesticulation, facial expression and a laser pointer in order to communicate the help message verbally, non-verbally, and in written form. The teacher is one of the virtual characters that operate within the DynaLearn workbench [4], blending in with the other pedagogical use cases.

4 Concluding Remarks

We showed that it is possible to integrate context-dependent assistance knowledge inside a complex modelling and simulation environment such as DynaLearn. When it comes to these basic help modes, existing qualitative workbenches often resort to (searchable) hypertext resources that do not dynamically adapt to what the learner is working on right now (e.g. Betty's Brain [5:188], VModel [6:824]).

We believe that these basic help facilities provide an important scaffold for learners, especially those that are new to and/or exploring the workbench. These assistance modes can support other, more advanced feedback facilities, such as a Teachable Agent. Evaluation studies with learners are planned in the near future.

References

1. Bredeweg, B., Liem, J., Linnebank, F., Bühling, R., Wißner, M., del Río, J.G., Salles, P., Beek, W., Gómez Pérez, A.: DynaLearn: Architecture and approach for investigating conceptual system knowledge acquisition. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 272–274. Springer, Heidelberg (2010)
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. Mittal, V., Moore, J.: Dynamic Generation of Follow up Question Menus: Facilitating Interactive Natural Language Dialogues. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 1995, pp. 90–97 (1995)
4. Mehlmann, G., Häring, M., Bühling, R., Wißner, M., André, E.: Multiple Agent Roles in an Adaptive Virtual Classroom Environment. In: Intelligent Virtual Agents, pp. 250–256 (2010)
5. 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)
6. Forbus, K., Ureel, L., Carney, K., Sherin, B.: Qualitative Modeling for Middle-School Students. In: Proceedings of the 18th International Qualitative Reasoning Workshop, pp. 81–87 (2004)

Evaluation of WebxPST: A Browser-Based Authoring Tool for Problem-Specific Tutors

Stephen B. Blessing[1], Shrenik Devasani[2], and Stephen Gilbert[2]

[1] University of Tampa, 401 W. Kennedy Blvd., Tampa, FL 33606, USA
[2] VRAC, Iowa State University, 1620 Howe Hall, Ames, IA 50011, USA
sblessing@ut.edu, shrenik@iastate.edu, gilbert@iastate.edu

Abstract. Authoring tools enable the more rapid creation of intelligent tutoring systems. Such tools are essential for tutors to become more widespread. In this study we evaluate WebxPST, a browser-based authoring system that enables non-programmers to create model-tracing-like intelligent tutors. Five authors, two course instructors and three undergraduates, created 74 problems suitable for use in an undergraduate statistics curriculum. A subset of these problems was deployed in a classroom. These authors quickly mastered the authoring interface showing the feasibility of the tool.

Keywords: authoring tools, problem-specific tutors, statistics.

1 Introduction

If intelligent tutoring systems (ITSs) are to be used more broadly, then the tools used to create them need to become easier to use and more widespread themselves. In our own work we have sought to lower the bar of tutor creation. The present work attempts to continue that tradition and produce an authoring tool, as well as the subsequent tutor, that works within a common web browser.

The past decade has contained some amount of work on authoring tools for ITSs, some of which is discussed in an edited volume [1] and also a recent special edition of the International Journal of Artificial Intelligence in Education. Much of this work is motivated by the observation that, historically, creating ITSs is time consuming. An often cited statistic is that for earlier ITSs it took 200 hours of development time create 1 hour of instruction [2]. Not only is this work time consuming, it requires much expertise, with team members needing experience in interface design, cognitive science, pedagogy, and programming. This increases the costs of creating the ITS and also limits the number of ITSs that can be created within the various domains.

We developed the Extensible Problem-Specific Tutor system (xPST) to allow us to more rapidly develop model-tracing-like tutors [3]. The xPST system involves two design goals: 1) the interface is separable from the tutoring component, and 2) the syntax, while having power, is easy and not very code-like. Separating the interface from the tutoring component allows for swapping in and out of interfaces (e.g., custom or existing third-party software) while retaining the same tutoring backend.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 423–425, 2011.

Having a simple syntax to create the instruction lowers the cost of entry by non-programmers. We developed a plugin, WebxPST, for xPST that allows it to use a web page for the student interface. The plugin allows the Firefox browser to mark webpages for correct and incorrect answers and to display messages to the student.

2 The xSTAT Project

We started the xSTAT project to create a set of tutored problems for college-level statistics. As a case study of the WebxPST system, 5 non-programmers developed problems to be used in such a college course. Two were instructors of the course, and three were undergraduates who had successfully completed the course in the past.

To develop the materials for the xSTAT tutor, both the problems and the instruction had to be authored. The authors used JotForm (www.jotform.com) to easily create the webform that students would use. JotForm allows a wide variety of widgets to be dragged-and-dropped onto a webform for easy layout. Participants then used the WebxPST authoring website to create the instruction and tutoring for the problems they created using the JotForm form builder.

We created a small set of instructional materials for the participants to learn both JotForm and WebxPST. Throughout the month that the participants spent authoring problems, we had four 1 hr meetings where we gave them the initial instruction, discussed problems they encountered, authoring strategies they discovered, and other relevant issues. This was the total amount of their instruction.

The WebxPST website logged the time each participant spent coding (i.e., actually typing in code in the xPST code box) in addition to the amount of time spent logged into the system. Participants were asked to create 15 problems apiece dealing with *z*- and *t*-tests. All problems consisted of a real-world-scenario, a data table, and then 8-10 questions relating to the scenario.

2.1 Results

One participant authored 12 problems, three others authored 15 and another authored 17. In total these 5 participants authored 74 problems over the course of 1 month.

We took two time measures: total time logged into the system and time spent typing xPST code. The total time measure captures the time spent typing the form into JotForm, performing calculations in SPSS, and formulating the problem itself. The xPST code time is included in the total time, and is a relatively exact estimate of how long participants spent typing the instructional code. Authors averaged 28.57 hr logged on to the system across the month, and a mean of 7.37 hr editing the xpst file (see Figure 1; problems are binned in groups of 3 to show the average for that group). At the end of the experiment, participants were spending less than 45 min authoring a problem total, with just under 18 min of that time spent writing the code. Students on average spent 10.67 minutes solving these problems. The ratio of the time spent authoring at the end of the experiment (44.50 min) to the student time spent solving, a 4.2:1 ratio, compares quite admirably to the earlier estimates of ITS creation time.

Participants created a variety of problem types and questions. Each participant authored 4 problem types (*z*-tests, and 3 different *t*-test types), with most producing between 4 and 6 of each type. The 74 problems averaged 8.50 subgoals apiece. Each of the problems had a unique scenario and data. We tested all problems to ensure hints worked and answers accepted. This was a meaningful and usable set of problems the participants authored, very suitable for multiple homework assignments across the three targeted text chapters.

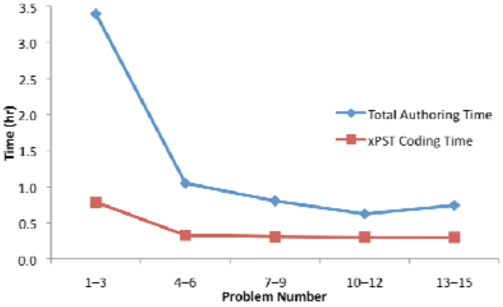


Fig. 1. Number of problems authored versus time

3 Discussion

Non-programmers and non-cognitive scientists created a substantial set of meaningful tutored problems with minimal training with the WebxPST authoring tool. The tutor's feedback is similar to that of a model-tracing tutor. These problems were used in a college-level course as a homework assignment, and were authored and delivered in a standard web browser. Though instantiated within statistics, the tools used were very general purpose and could be adapted to a number of domains. With these advantages, we feel this case study was a success, and shows the viability of our approach.

References

1. Murray, T., Blessing, S., Ainsworth, S. (eds.): Authoring Tools for Advanced Technology Educational Software. Kluwer Academic Publishers, Dordrecht (2003)
2. Woolf, B.P., Cunningham, P.: Building a community memory for intelligent tutoring systems. In: Forbus, K., Shrobe, H. (eds.) Proceedings of the Sixth National Conference on Artificial Intelligence, pp. 82–89. AAAI Press, Menlo Park (1987)
3. Gilbert, S., Blessing, S.B., Kodavali, S.: The extensible problem-specific tutor (xpst): Evaluation of an api for tutoring on existing interfaces. In: Dimitrova, V., et al. (eds.) Proceedings of the 14th International Conference on Artificial Intelligence in Education, pp. 707–709. IOS Press, Brighton (2009)

Gesture-Based Affect Modeling for Intelligent Tutoring Systems

Dana May Bustos, Geoffrey Loren Chua, Richard Thomas Cruz,
Jose Miguel Santos, and Merlin Teodosia Suarez

Center for Empathic Human-Computer Interactions, De La Salle University
2401 Taft Avenue, 1004 Manila, Philippines
{danabustos,t4u842_geoff,tom.cruz17,
josemiguelsantos2000}@yahoo.com,
merlin.suarez@delasalle.ph

Abstract. This paper investigates the feasibility of using gestures and posture for building affect models for an ITS. Recordings of students studying with a computer were taken and an HMM was built to recognize gestures and posture. Results indicate distinctions can be achieved with an accuracy of 43.10% using leave-one out cross validation. Results further indicate the relevance of hand location, movement and speed of movement as features for affect modeling using gestures and posture.

Keywords: Gesture recognition, affect modeling, intelligent tutoring systems, emotions in gestures.

1 Introduction

Intelligent Tutoring Systems (ITSs) are computer-based educational systems that provide individualized instructions like a human tutor [1]. An ITS capable of providing affective support will presumably enrich the student's learning experience. Most works on affective modeling for ITSs use specialized equipment, or sensors that measure physiological data. Because sensors are obtrusive when worn, it may affect student concentration and may be distracting. The use of specialized equipment is also expensive, difficult to deploy and duplicate.

Gestures are movements of the body or the limbs that express an idea or a sentiment [2]. Studies suggest that gestures are indicative of a person's affective state [3]. In this work, we study the feasibility of using posture and gestures for student affect modeling using a web camera because it is inexpensive, ubiquitous, and unobtrusive. It provides the student freedom of movement and spontaneity. In the succeeding sections, details about data collection, model building and preliminary results are presented.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 426–428, 2011.

2 Gesture Modeling, Tests and Preliminary Results

2.1 Data Collection, Feature Extraction and Model Building

Most gesture-based affect recognition systems use acted gestures. In this work, spontaneous gestures were used as these occur more frequently in real life, are subtler and less dramatic. Spontaneous gestures were collected from students using the computer for academic-related tasks. Three students were asked to sit in front of a computer with a web camera in front to record the session, studying his notes or researching online. They annotated their own videos with the following discrete academic emotion labels: boredom, flow, confusion, frustration. A total of 60 gestures were collected with 25 instances of flow, 22 instances bored, 11 instances confused, and 2 instances frustrated clips.

2.2 Feature Extraction and Model Building

Manually segmented gesture recordings were pre-processed, applying Gaussian blur filter to smoothen and remove noise. OpenCV libraries were used to detect the face and track its movement. Aside from the head, the shoulders were detected, and as well as its movement. Simple Expectation-maximization (EM) algorithm was used to cluster the data. The data set was re-labeled, changing its class to the cluster number from the EM algorithm. The frame sequences were then converted into discrete observations and saved into a text file. *K*-means was used to initially build an HMM for each emotion. To pick the best number of states for the model, a trial-and-error approach was used, iterating over models with different number of states and using the model that returned the highest probability in classifying its training set. To determine this, the forward-backward algorithm was used to compute the probability of the training set, given the created HMM.

3 Observations, Preliminary Results and Analysis

3.1 Gestures and Academic Emotions

Based on initial data that was collected, dominant emotions exhibited were flow and boredom, comprising 78% of the entire data set. The absence of frustration might indicate that an unstructured approach to data collection is unwise because sufficient coverage is not achieved. Therefore, inducing specific academic emotions is advisable. It was also observed that a majority of gestures were compound, i.e., it was a combination of more than one movement. For instance, the test subjects leaned forward and touched their faces when they felt confused. It was also interesting to discover that some gestures convey different emotions even for the same person. For instance, the gesture of scratching the face conveys frustration, boredom and confusion for one test subject. The difference in the gestures lies in how quickly they perform the movement between these.

4.2 User-Specific and Stereotype Gesture-Based Models

To test the accuracy of the model, leave-one out cross validation method was used. A user-specific model was built, and this resulted to an accuracy score of 24.99%. A stereotype model was built afterwards, with confused having an accuracy score of 36.36%, flow with a score of 36%, boredom with a score of 57.14%, achieving an average accuracy score of 43.10%[1].

The improvement in the results from a user-specific model to a stereotype model may be explained by the fact that all the test subjects expressed their emotions in very similar manner. For instance, transitioning from bored (crossing arms and fidgeting on the seat) to flow (crossed arms but no movement) was the same for two subjects. Expressing confusion as change in posture (from leaning forward to back or vice-versa, including touching the face using the dominant hand) was also the same for two subjects.

While there was an improvement in the results (due largely to the increase size of the data set), and that the accuracy is better than chance, there leaves a lot of room for improvement. Specifically, the hand and its position need to be recognized and tracked. Likewise, the shiftiness of the subject on the chair is also relevant. Additional data needs to be collected to improve the average accuracy score.

4 Concluding Remarks

This paper presented how gesture and posture can be used to to distinguish academic emotions, specifically confusion, flow, boredom and frustration. A gesture corpus containing 60 video segments was used to create emotion models for confusion, flow, and boredom. At best, the accuracy of the model was at 43.10% using leave-one out cross validation.

References

1. Perez, Y., Gamboa, R., Ibarra, O.: Modeling affective responses in intelligent tutoring systems. In: Proceedings of IEEE International Conference on Advanced Learning Technologies 2004, pp. 747–749 (2004)
2. Merriam-webster's collegiate dictionary. MerriamWebster, Springfield, MA (2004)
3. Ekman, P., Friesen, W.: Detecting deception from the body and face. Journal of Personality and Social Psychology 29, 288–298 (1974)

[1] No model was built for frustration due to insufficient data.

Turn on the TV to turn off the TV: An Application of Adaptive Learning Television to Discuss the Television

Ana C.A. de Campos and Fábio N. Akhras

Renato Archer Center of Information Technology
Rodovia Dom Pedro I, km 143,6 - 13089-500 Campinas, São Paulo, Brazil
fabio.akhras@cti.gov.br, carops@gmail.com

Abstract. This paper presents an approach to the development of adaptive learning television programs to raise awareness of social matters. Interacting with a simple adaptive television program people is led to discuss issues of social matters in order to learn about these issues. The adaptive television program that has been created stimulate the spectator to discuss issues of social relevance through the interaction with a series of media objects of two main kinds: videos and choices. The focus of the application developed is the discussion of the television. One of the objectives of the project is to explore the use of adaptive television to provide independent learning programs that can reach marginalized populations that live in the more underdeveloped and isolated regions of the country, as a way of promoting the social inclusion of this population, which have no access to the internet but has plenty access to the television.

Keywords: television, adaptivity, interactivity, learning, social inclusion.

1 Introduction

Digital television promises structural, aesthetic and linguistic innovations that have not been fully explored yet. In our work, we have analyzed these aspects of the analog television and propose to address the production of digital television programs to support learning. The objective is to explore the interactivity and adaptivity made possible by the digital technology, without losing the fundamental characteristic of the television: entertainment.

Interactivity is a central aspect of learning. The main change that will come with the digital television is the transformation of a passive spectator in someone that participates in the television program. This can be used to promote the learning of contents that can make the spectator more critical in relation to social matters. In addition, adaptivity offers many possibilities for enhancing interaction [1].

Therefore, the work developed involved the study of the characteristics of television language, its codes and aesthetics, to create a television program for learning that introduces the interactivity and the adaptivity without losing the television role of providing information and entertainment.

A discussion structure based on the Socratic method has been created to support the adaptation of the television program according to the way the spectator reacts to

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 429–431, 2011.

the videos. In the approach, the questioning, which is the central issue of the Socratic method, comes from the videos. (instead of coming from a teacher). After each video that is presented, the spectator can reflect on the questioning that is presented in the video, and answer a question related to the video content which is presented to the spectator for a choice of an answer. According to the answer, the spectator is led to watch videos that discuss or contra pose to the answer given by the spectator promoting a discussion of the issues addressed by videos and questions.

2 Television and Learning

To understand television it is important to study its language, patterns, divisions, stereotypes, types of programs and so on [2]. Among the types of television programs, those that are more effective as learning programs are the documentaries. Documentaries can be compared to expositive classes in which the students learn by observing and listening. A documentary is even more effective than an expositive class because it uses images. Other important sources of learning in television are the programs of debate, in which some themes are discussed trying to work out diverse viewpoints. During these debates the spectator tends to organize their own reflection but do not have the possibility of expose it.

In trying to join these two kinds of programs of established formats, the challenge was how to make the spectator participate in the discussion and learn at the same time. The digital television offers means to approach this issue by making questions during the program. These questions lead the spectator to reflect on the subject and express their opinion, at the same time that they watch the exhibited contents.

Therefore, the subject needed to be a controversial subject that could address a plurality of perspectives, and at the same time be a daily subject so that it could have a strong connection with the spectator. Analyzing these issues we decided to address the television as the subject of our interactive television program. The result is the development of an application of adaptive learning television to discuss the television.

3 An Adaptive Television Program to Discuss the Television

In order to raise the awareness about the passivity of the television spectator and how this will change with the introduction of the interactivity in the television, the program created intends to promote the discussion of several aspects of television to develop a critical view on the spectator. As a basis to create the discussion structure of our adaptive television program we used the argumentation method known as “Socratic Method”. According to this method the participants in a dialogue are led to a discussion process through the exposition to a series of questions which reveal conceptions and associations that do not make sense, leading them to revise their reasoning and beliefs, and to reflect more deeply on the questions discussed.

In our program, through the exposition to a series of videos the spectator is led to revise their television habits. To do that the program always present a video that challenges an opinion given by the spectator with regard to a previously watched video. The interaction of the spectator with the program is based on questions that were inserted between videos in the discussion structure. In order to avoid that the choices made by the spectator could lead to the set of videos not watched being larger than the set of videos watched, we defined three kinds of interaction patterns to use in the discussion structure. These patterns provide a better use of the material produced without losing the diversity of choices. The patterns are:

- Two different paths after the question, with different videos according to the answer given. After the initial differentiation the two videos lead to the same video, which will introduce a new question and continue the program.
- The two answers lead to the same video, which will be interpreted in different ways, confirming the beliefs of the spectator or trying to show an opposite point of view, or even complicating the previous question.
- One of the paths lead to an additional video then both paths lead the same video.

An example of the first interaction pattern is the following. The initial video argues that "we live in television, we watch television, it is always there" and questions how much we really think of it. The question presented after the first video is "How frequent do you think on the TV you watch?" The answers are "sometimes" and "always". If the answer is "sometimes" the next video argues that the images shown on television become part of how we perceive reality, through a process of cultivation of values that the TV is continually emphasizing. If the answer is "always" the next video presents a deeper analysis arguing that the histories shown on television are not neutral, they form a coherent system that give us stable ways of looking at the world. The final video of the interaction pattern makes a kind of synthesis of the two paths an explores aspects associated with media and society. The program was implemented using the Ginga-NCL language [3].

This program when exhibited made it very evident its potential to provoke reflection about the television, reducing the passivity of the act of watching and introducing the act of making choices. Programs like this when broadcasted on digital TV will have a lot to contribute to the development of a critical view on the spectators about themes of social relevance, like the role of television in society.

Acknowledgments. The work reported in this paper is being sponsored by the National Research Network (RNP) of Brazil.

References

1. Masthoff, J., Pemberton, L.: Adaptive hypermedia for personalized TV. In: Chen, S., Magoulas, G. (eds.) Adaptable and Adaptive Hypermedia Systems, pp. 246–263. IDEA group publishing (2005)
2. Marshall, J., Werndly, A.: The Language of Television. Routledge, New York (2002)
3. Laiola Guimarães, R., Monteiro de Resende Costa, R., Gomes Soares, L.F.: Composer: Authoring tool for iTV programs. In: Tscheligi, M., Obrist, M., Lugmayr, A. (eds.) EuroITV 2008. LNCS, vol. 5066, pp. 61–71. Springer, Heidelberg (2008)

Scaffolding Metacognitive Processes in the Ecolab: Help-Seeking and Achievement Goal Orientation

Amanda Carr (nee Harris)[1], Rosemary Luckin[2],
Katerina Avramides[2], and Nicola Yuill[3]

[1] Department of Psychology, Roehampton University, London, UK
amanda.carr@roehampton.ac.uk
[2] London Knowledge Lab, Institute of Education, London, UK
{r.luckin,k.avramides}@ioe.ac.uk
[3] School of Psychology, University of Sussex, Brighton, UK
nicolay@sussex.ac.uk

Abstract. Ecolab is an interactive learning environment designed to support 10 - 11 year old learners' understanding of ecology. The system offers help at different levels of specificity and invites users to consider what level of help they need – a form of metacognitive assistance. In this paper we report results from an empirical study which investigates how learners respond to metacognitive assistance as provided by two different versions of the Ecolab according to differing achievement goal orientations.

1 Introduction

Ecolab is an interactive learning environment for 10 - 11 year old learners designed to support their understanding of ecology concepts such as food chains and webs. In a series of studies [1, 2, 3] we have used the software to test the design of metacognitive tools to support learning. In particular, we have focused on help-seeking and task selection as aspects of metacognition that vary a great deal between young learners.† We have identified achievement goal orientation, whether learners pursue mastery or performance goals, as one important influence on help-seeking behaviour within the Ecolab environment [3].

In this paper we report some early results of an empirical study with a new version of Ecolab that was developed for performance-oriented learners. This data is not published elsewhere and fits well with the conference theme of "Next Generation Learning Environments: Supporting Cognitive, Metacognitive, Social and Affective Aspects of Learning". The Ecolab software aims to support learners both cognitively and metacognitively and offers a test-bed for the exploration of learner model design to enable systems to adapt to learner motivation.

2 Ecolab

Ecolab builds a software-based model of the learner and scaffolds their interactions with timely interventions. This model represents the system's interpretation of the

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 432–434, 2011.

learner's understanding of a small curriculum of ecology knowledge, and the learner's ability in two metacognitive processes: help seeking and task selection. Concerning help seeking, the model assesses the frequency and level of help requested by the learner in relation to their level of success at each action. The system then provides scaffolding prompts aimed to make learners aware of their help seeking behaviour. For example, when a learner needs help at the domain level but is not using the help facility (a choice of clue at one of four levels of specificity), the system responds by providing a meta-level prompt reminding the child that help is available (Meta-Level 1: Don't forget that you can ask Ecolab for help). Alternatively a learner might consistently select high-level clues even though they are performing successfully. In this case the system will prompt the learner to select a lower level clue (Meta-Level 2: Why not ask Ecolab for less help). In the current study we were interested in whether metacognitive assistance that can adapt to learners' achievement goal orientation is more effective in promoting learning gains than assistance which does not distinguish between mastery and performance orientations.

3 Method

Participants

Participants were 49 (28 males and 21 females) Year 5 children (10 years old) attending a semi-rural primary school in the South of England. A pre-test [1] was completed first followed by individual achievement goal assessments [for details on materials see 3]. These were followed by two Ecolab sessions about a week apart, each lasting 30 minutes. A further week later children completed the Ecolab post-test.

Design and measures

Based on previous observations of mastery and performance learners, we developed a new performance-oriented version of Ecolab for this study. The main revision was the addition of encouraging comments in the metacognitive suggestions such as "Good Try!" or "Keep on with the activity". These were included to provide extra support to performance-oriented learners who show a tendency to give up more readily on difficult activities [3, 4]. As mastery learners had not shown these tendencies, the 'mastery' version of the software remained unchanged. The current analysis is based on 14 mastery- and 12 performance-oriented learners who used the version of the software that matched their goal orientation. Learning outcomes were measured by calculating the difference between pre- and post-test scores. System logs were used to measure the type of help learners used during the interaction and the extent to which learners made use of the metacognitive assistance.

4 Results

We started by creating two groups of higher and lower ability learners using a median split based on pre-test scores. An ANOVA tested the effect of ability and achievement goal orientation on learning gains. This revealed no main effect of achievement goal orientation but a significant effect of ability group (F (1, 40) = 386.01, $p < 0.02$); low ability learners showed the most learning gains overall while high ability learners showed no significant change. This was not due to a ceiling effect as no learner in the high ability group scored higher than 48 out of 53 on the pre-test and the mean for this group was much lower (M = 38.62, SD = 5.5). A second ANOVA used the proportion of metacognitive assistance used by the learner as the dependent variable. This showed no main effect for goal orientation but a difference between higher and lower ability learners at a level approaching significance (p = .1); the lower ability group used a higher proportion of metacognitive assistance.

5 Discussion

These results partly replicate previous findings; low ability learners show the greatest learning gains [1]. However, they are not entirely consistent with our previous findings on goal orientation which showed differences between mastery- and performance-oriented learners' use of the help facility [3]. One possibility for this is that by adapting the software to match the individual learner's achievement goal orientation previous differences between mastery and performance oriented learners are no longer evidenced. To test this hypothesis follow up analysis will compare the matched group to a group of learners in which the software was mismatched to their individual achievement goal orientation.

References

[1] Luckin, R., Hammerton, L.: Getting to know me: Helping learners understand their own learning needs through metacognitive scaffolding. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 759–771. Springer, Heidelberg (2002)

[2] Martinez-Miron, E., Harris, A., Du Boulay, B., Luckin, L., Yuill, N.: The role of Learning goals in the design of ILEs: Some issues to consider. In: 12th International Conference on Artificial Intelligence in Education (AIED), pp. 427–434. IOS Press, Amsterdam (2005)

[3] Harris, A., Bonnett, V., Luckin, R., Yuill, N., Avramides, K.: Scaffolding effective help-seeking behaviour in mastery and performance oriented learners. In: Dimitrova, V., Mizogucji, R., du Boulay, Graesser, A. (eds.) Artificial Intelligence in Education, pp. 425–432. IOS Press, Amsterdam (2009)

[4] Harris, A., Yuill, N., Luckin, R.: The influence of context-specific and dispositional achievement goals on children's paired collaborative interaction. British Journal of Educational Psychology 78(3), 355–374 (2008)

Learning with ALEKS: The Impact of Students' Attendance in a Mathematics After-School Program

Scotty D. Craig[1], Celia Anderson[1], Anna Bargagloitti[1], Arthur C. Graesser[1], Theresa Okwumabua[1], Allan Sterbinsky[2], and Xiangen Hu[1,*]

[1] University of Memphis, Memphis, TN, 38152, USA
[2] Jackson-Madison County School System, 310 North Parkway, Jackson, TN, 38305, USA
{scraig,croussea,brggltti,a-graesser,tokwumab,xhu}@memphis.edu,
adsterbinsky@jmcss.org

Abstract. We examined the effectiveness of using the Assessment and LEarning in Knowledge Spaces (ALEKS) system as a method of strategic intervention in after-school settings to improve the mathematical skills of struggling students. The study randomly assigned students into a classroom that either worked with the ALEKS system individually on computers or were taught by teachers in an interactive classroom. Results from year one revealed that students randomly assigned to the ALEKS condition significantly out performed students assigned to the teacher condition on a state assessment test (TCAP). However, this was only if the students received sufficient exposure to the program.

Keywords: After-school program, ALEKS, Mathematics education.

1 Introduction

Given the growing deficiency in mathematics education [1, 2], it is worthwhile to implement and test alternative computer technologies to help raise student performance in mathematics.

Technology is generally believed to have a positive impact on student learning in mathematics. Nevertheless, the research on using technology to improve performance in mathematics has provided some mixed results when evaluated in K–12. Some of the news is positive. In a review of research on the effects of technology on student's mathematics gains, Schacter [3] reviewed over 700 empirical research studies in which students had exposure to computer-assisted instruction. The students showed overall positive gains in achievement on tests that spanned researcher-conducted tests, standardized state tests, and national tests. However, Dynarski et al., [4] reviewed software products for first grade reading, fourth grade reading, sixth grade math, and algebra founding no significant test score differences between the groups of students. Similarly, the report of the National Mathematics Advisory Panel [2] points to mixed results in the research on computer-based tutorials. Therefore, our study was conducted to test ALEKS for 6th graders in an after-school setting at 4 schools.

[*] Corresponding author.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 435–437, 2011.

ALEKS uses Bayesian networks to adaptively select the next skill for a student to work on. The Bayesian networks of the knowledge space model attempts to fill learning deficits and correct misconceptions adaptively and dynamically using Knowledge space theory [5]. It tracks the knowledge states of learners in fine detail and adaptively responds with assignments that are sensitive to these knowledge states.

2 Methods

Participants (291 sixth grade students in a west Tennessee school district) who volunteered for our after-school program were randomly assigned to one of two conditions (ALEKS & Teacher). They attended the program two days a week for two hours a day over 25 weeks. The two hour sessions were divided into five 20-minutes segments with ten minute periods for start-up and dismissal. The students received three 20 minute instruction sessions. The instruction sessions were separated by two 20 minute *down-time* sessions during which students received snacks and played games. In the ALEKS condition, during each of the 20 minutes instructional periods, students interact with the program. The three learning phases in the teacher classrooms followed a Lecture, group application and practice schedule. The topics covered in both conditions are guided by the state performance indicators (SPIs).

For both the ALEKS and teacher conditions the outcome measure of performance was the Tennessee Comprehensive Assessment Program (TCAP), the states yearly student achievement measure. The scores of the 5th grade TCAP were used to assess students' pre program mathematics knowledge whereas the scores of the 6th grade TCAP were used as the posttest.

3 Results and Discussion

A series of t-tests were conducted on student's TCAP scores from the 5th (before the program, 2009 TCAP) and 6th grade (after the program, 2010 TCAP). These two tests are not equivalent pretest and posttest measures because they are testing different information and have a different range. However, these tests do provide information on the student's mathematics proficiency. The maximum score for each test was 900. However, the state of Tennessee modified the testing requirements between the 2009 and the 2010 TCAP. There were two primary changes. The first was proficiency levels. The 2009 TCAP had levels of 500 Below Basic, 657 Basic, 712 Proficient and 752 Advanced. The 2010 TCAP has level cutoffs of 600 Below Basic, 703 Basic, 755 Proficient and 791 Advanced. More importantly for the current project, the 2010 TCAP modification included more advanced topics requiring our 6th grade students to know math topics that were previously on 7th and 8th grade tests. These changes reflected attempts to aligned with national standards of NAEP.

One of our major problems observed in the first year was attrition. Of the 291 students starting the program, less than 30% completed our program. Of these only 24 showed consistent performance. Because of this, we analyzed our data at two "dosage" levels. If students signed up for the program and started attending they were in the "Any dosage" level (*n* = 291). Those 24 students with excellent attendance were included in the "full dosage" level. There were no significant differences between students in either condition for the two dosage levels on 5th grade TCAP performance. However, less accomplished students persisted (See Table 1).

No significant differences were observed on student performance between groups at the *any dosage* ($t(289) = .79$, $p = .22$, $d = .09$) level. However, a significant difference was observed on student's TCAP mathematics ability at the full dosage level ($t(22) = 1.41$, $p = 0.08$; $d = .47$).

Table 1. Student's means and standard deviations for TCAP by year and condition

	5th grade TCAP Mathematics subscore				6th grade TCAP Mathematics subscore			
	Teacher		*ALEKS*		*Teacher*		*ALEKS*	
	Mean	*ST Dev*	*Mean*	*ST Dev*	*Mean*	*ST Dev*	*Mean*	*ST Dev*
Any dosage	487.38	25.57	488.39	27.15	702.90	95.58	711.75	93.58
Full dosage	469.33	38.08	483.28	18.93	667.67	167.50	723.59	17.74

Several conclusions can be drawn from these findings. First, from looking at the 5th grade TCAP means, our subject population was below the scale rankings, not even reaching the lowest cutoff score of 500. After our program the students increased two categories on average to the basic level. So, it appears that both of our after-school programs (ALEKS and Teacher conditions) were helpful to our students.

Another conclusion is that dosage matters. While the small sample size of the full dosage level weakens this finding, the significant difference and medium effect size indicate that the ALEKS after-school program could be significantly better than certified mathematics teachers. However, replication is needed in future years.

Acknowledgments. This research was supported by the Institute for Education Sciences (IES) Grant R305A090528 to Dr. Xiangen Hu, PI. 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. National Center for Education Statistics: National assessment of educational progress: The nation's report card (2008)
2. National Mathematics Advisory Panel: Foundations for success: The final report of the national mathematics advisory panel. Washington, DC: US Department of Education (2008)
3. Schacter, J.: The impact of educational technology on student achievement: What the most current research has to say (ERIC Document Reproduction Service No. ED 430 537). Milken Exchange on Educational Technology, Santa Monica, CA (1999)
4. Dynarski, M., Agodini, R., Heaviside, S., Novak, T., Carey, N., Campuzano, L., et al.: Effectiveness of reading and mathematics software products: Findings from the first student cohort. U.S. Department of Education, IES, Washington, DC (2007)
5. Doignon, J.P., Falmagne, J.: Knowledge spaces. Springer, Berlin (1999)

Predicting Human Scores of Essay Quality Using Computational Indices of Linguistic and Textual Features

Scott A. Crossley[1], Rod Roscoe[2], and Danielle S. McNamara[2]

[1] Department of Applied Linguistics, Georgia State University, 34 Peachtree St. Suite 1200,
One Park Tower Building, Atlanta, GA 30303, USA
scrossley@gsu.edu

[2] Institute for Intelligent Systems, The University of Memphis, FedEx Institute of Technology,
Memphis, TN 38152
rdroscoe@memphis.edu, dsmcnamra1@gmail.com

Abstract. This study assesses the potential for computational indices to predict human ratings of essay quality. The results demonstrate that linguistic indices related to type counts, given/new information, personal pronouns, word frequency, conclusion n-grams, and verb forms predict 43% of the variance in human scores of essay quality.

1 Introduction

In educational settings, trained, professional readers (e.g., teachers) typically assess writing quality. These evaluations have important consequences for the writer because these judgments provide a source of feedback and determine passing or failing grades. The goal of this study is to investigate the linguistic and textual features in argumentative essays that influence human judgments of writing quality. This approach is in contrast to writing research that primarily investigates cognitive and behavioral processes that occur during writing (i.e., planning, translating, reviewing, and revising) but not the products of writing [1], such as the linguistic features of a text [2]. However, linguistic features at the word, syntactic, and discourse levels have been found to significantly influence essay quality, and can be important indicators of writing development [3].

A better understanding of the relationships between linguistic features and writing quality has several benefits. This knowledge may help writers to make more informed decisions about effective writing and composition. Such knowledge would also help readers and teachers make more accurate or specific evaluations of writing quality, which would enable them to provide more precise or targeted feedback.

In this study, we use computational linguistic indices to assess human ratings of essay quality. Because these linguistic and textual analyses are automated, they can be implemented within computer systems that automate the process of assessing writing and providing student feedback. Thus, this research informs both writing pedagogy and instructional technology (e.g., intelligent tutoring systems).

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 438–440, 2011.

2 Methodology

We collected 314 timed (25-minute) essays written by 314 college freshmen at a large university in the United States. All essays were written in response to two Scholastic Achievement Test (SAT) writing prompts. We separated the corpus into a training ($n = 209$) and test set ($n = 105$) based on a 67/33 split. The training set was used to select the computational indices for the initial statistical analyses (correlations and regression analyses). The test set was used to calculate the predictive ability of the selected variables in an independent corpus.

Expert raters rated the quality of the 314 essays in the corpus using a standardized SAT rubric for holistic quality. The final interrater reliability for all essays in the corpus was $r > .75$. We used the mean score between the raters as the final value for the quality of each essay unless the differences between the 2 raters was >= 2, in which case a third expert rater adjudicated the score.

The linguistic features of the essays were analyzed using Coh-Metrix indices [4]. We selected indices from Coh-Metrix with theoretical and empirical links to essay quality and writing proficiency. These indices were organized into broad measures that reflected general linguistic constructs: cohesion, lexical sophistication, syntactic complexity, rhetorical strategies, and text structure. Cohesion measures included causality, incidence of connectives, incidence of logical operators, lexical overlap, semantic co-referentiality, anaphoric reference, prompt overlap, and paragraph overlap. Lexical sophistication measures included word hypernymy, word polysemy, academic words, lexical diversity, word frequency and word information indices (e.g., word concreteness, familiarity, meaningfulness, and imagability). Syntactic complexity measures included syntactic similarity and phrase structure complexity. Rhetorical strategies measures included indirect pronouns, amplifiers, downtoners, exemplification and n-gram indices for rhetorical phrases common in high quality introductory, body, and concluding paragraphs.

3 Results

We selected the computational indices that demonstrated the highest Pearson correlation when compared to the human essay scores, and that did not demonstrate multicollinearity. This led to the selection of 26 variables.

A linear regression analysis was conducted with the 26 variables. These 26 variables were regressed onto the raters' score for the 209 essays in the training set, and were checked for outliers and multicollinearity. The linear regression yielded a significant model, $F(6, 200) = 23.202$, $p < .001$, $r = .641$, $r^2 = .410$. Six variables were significant predictors: total types, LSA given/new, incidence of personal pronouns, word frequency, all n-grams (conclusion paragraphs), and incidence of verb base form. The model for the test set using these variables yielded $r = .655$, $r^2 = .429$.

4 Discussion

This study demonstrated that a combination of computational indices related to type counts, given/new information, incidence of personal pronouns, word frequency, incidence of n-grams related to conclusion quality, and incidence of verb base form explained 43% of the variance in human judgments of essay quality. This is a two-fold increase in predictive power over previous findings [3] and provides further evidence that computational indices can be used to assess essay quality.

These linguistic indices allow us to better understand how textual features influence human judgments of writing quality. As in past studies, longer essays (i.e., greater number of word types) that use more sophisticated vocabulary (i.e., less frequent words), and more complex grammar (i.e., fewer base verb forms) were judged higher in quality. In contrast to past studies, higher quality essays in this analysis also displayed more cohesion such that essays judged higher in quality maintained stronger links to previously given information. Our model also reported a positive relationship between essay quality and the incidence of conclusion n-grams (e.g., concluding phrases, conditionals, and modals) indicating that the presence of rhetorical elements is important in judgments of essay quality. Additionally, lower quality essays used more personal pronouns suggesting that weaker writers relied more on writer-based prose than reader-based prose.

Advancing research on automated linguistic analysis enhances our ability to detect and understand the textual features that contribute to effective writing. In turn, this empowers us to teach developing writers how to harness such knowledge to further their academic and professional goals, both via traditional feedback given by teachers, and by automated feedback and strategies taught by intelligent tutoring systems.

Acknowledgments

This research was supported in part by the Institute for Education Sciences (IES R305A080589 and IES R305G20018-02). Ideas expressed in this material are those of the authors and do not necessarily reflect the views of the IES.

References

1. Abbott, R., Berninger, V., Fayol, M.: Longitudinal relationships of levels of language in writing and between writing and reading in grades 1 to 7. Journal of Educational Psychology 102, 281–298 (2002)
2. Berninger, V., Mizoka, D., Bragg, R.: Theory-based diagnosis and remediation of writing disabilities. Journal of School Psychology 29, 57–79 (1991)
3. McNamara, D.S., Crossley, S.A., McCarthy, P.M.: Linguistic features of writing quality. Written Communication 27, 57–86 (2010)
4. Graesser, A.C., McNamara, D.S., Louwerse, M.M., Cai, Z.: Coh-Metrix: Analysis of text on cohesion and language. Behavioral Research Methods, Instruments, and Computers 36, 193–202 (2004)

ProTutor: Historic Open Learner Models for Pronunciation Tutoring

Carrie Demmans Epp[1] and Gordon McCalla[2]

[1] TAGlab, Department of Computer Science, University of Toronto, Bahen Centre for Information Technology, 40 St. George Street, Room BA4242, Toronto, Ontario, Canada
[2] ARIES Laboratory, Department of Computer Science, University of Saskatchewan, 178 Thorvaldson Building, 110 Science Place, Saskatoon, Saskatchewan, Canada
cdemmans@cs.toronto.edu, mccalla@cs.usask.ca

Abstract. Acquiring proper pronunciation is difficult for second language learners. We built a Russian pronunciation tutor, called ProTutor, that uses open learner models (OLMs). Of particular interest is ProTutor's "historic OLM" that incorporates historic information about learner performance to encourage reflection and maintain learner motivation. In a formative evaluation participants indicated that ProTutor was helpful and fun to use.

Keywords: Open Learner Models, Computer Assisted Language Learning.

1 Introduction

Learners face many challenges. One of these is motivation, especially in areas where progress can be difficult to detect, like pronunciation [1]. Learners need safe practice environments [2]. So, we created a second language (L2) Russian pronunciation tutor (ProTutor) to meet this need. It has an Open Learner Model (OLM) and incorporates historic information about learner performance to maintain their motivation. A formative evaluation was performed to see if ProTutor maintained learner motivation or could be improved. It indicated that ProTutor and its learner models are motivating and that including historic information in the OLM is useful. It also confirmed that an OLM that includes audio information about learners' performance is desirable.

2 Related Literature

Many Computer Assisted Language Learning (CALL) systems use varied modes of interaction and feedback. ProTutor uses historic information in its OLM and several modes of interaction. Many older CALL systems (e.g., L2tutor and Word Munchers) rely on text-based interactions, but this is changing as more systems (e.g., INTELL and RosettaStone) incorporate audio materials as input or output. Simulation-based systems, such as the TLTS [3], use all input and output modes to replicate an immersive environment for intermediate or advanced learners. Many other CALL systems rely on combinations of instruction, tests, or games to help introductory learners; they

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 441–443, 2011.

focus on written L2 skills and vocabulary acquisition. See [4] for a complete CALL survey. As in other CALL systems, ProTutor uses games, instruction, and test-like activities to help novice learners develop pronunciation skills. ProTutor surpasses such CALL systems by incorporating OLMs to encourage learner reflection and self-awareness [5]. OLMs are common in intelligent tutoring systems (ITS) [6], but such ITSs rarely provide learners with information about how their performance changes over time. In contrast, ProTutor adds historic information to an OLM to provide learners with rich feedback about their progress.

3 System Overview

ProTutor's activities are sequenced to match the order of material in an accompanying course, and learners can progress through the activities at their own pace by following a prescribed learning path or by delving into personalized activity recommendations that come with instructional material. ProTutor tracks the performed activities and analyzes all utterances recorded by the learner. It determines the accuracy of the learner's pronunciation for each character of the alphabet and logs it in a pronunciation model. The diagnosis results are shown in the OLM by listing the three characters that the learner most often pronounces correctly (best) and incorrectly (worst). These lists provide positive and corrective information together, which helps maintain motivation by preventing the dismay that can accompany being told only your errors [7]. Learners are also shown an L2 sentence that highlights their pronunciation strengths and weaknesses. How an expert would pronounce the sentence is shown below how the learner would pronounce the sentence. This allows learners to compare their pronunciation to that of an expert so that they can see the ideal and work towards it.

Once learners have used the OLM for at least three weeks an open learner model with historic information (the HOLM) is presented to them; it adds information about the learner's previous performance and facilitates learner reflection about performance changes and the causes of these changes. The HOLM adds a previous pronunciation mapping immediately below the learner's current pronunciation mapping for the selected sentence. It also adds the previous best and worst pronounced characters beside the learner's current best and worst characters. The final feature of the HOLM is a chart that shows how the learner's pronunciation accuracy has changed over time.

4 System Evaluation and Results

A formative evaluation of ProTutor was performed in a university L2 Russian course. Participation required continued system use over nine weeks. All system use was tracked and feedback was collected through surveys following 3-week long stages of system use: no OLM or HOLM, OLM only, and HOLM only.

Five students participated in the evaluation; they attempted over 120 activities and recorded over 800 utterances. Four participants completed personalized activities based on information in their learner models. Three worked on improving characters in their worst list and one worked on characters from her best list because of the emphasis that the instructor had placed on them. On average participants viewed their

OLM 1.4 times (s.d. 0.55) and their HOLM 3.6 times (s.d. 1.82), which indicates that the HOLM was at least as useful to them as their OLM. Four participants continued to use ProTutor after the study's completion. One of them used ProTutor until mid-way through the next term when the course textbook changed, indicating that ProTutor was useful as long as it complemented the course material. Multiple participants said that ProTutor was helpful, easy to use, and fun. Responses to Likert-scale statements (1 – agree, 7 – disagree) revealed that ProTutor "reinforced what [they] were learning in class" (mean 1.8, s.d. 0.4) and facilitated "practising to speak in Russian" (mean 1.8, s.d. 0.4). Participants also liked many aspects of the learner model, including seeing the sounds that they were good at (mean 1.6, s.d. 0.5) and those that needed improvement (mean 1.8, s.d. 0.8). Participants also "felt that [their] pronunciation of Russian words improved" (mean 2.4, s.d. 1.1), and some requested that features be added to ProTutor. They wanted the ability to hear the pronunciation models.

5 Conclusions and Future Work

ProTutor uses a "snapshot" OLM of a learner's abilities and knowledge, but extends this by incorporating historic information into the model in order to maintain learner motivation and encourage reflection over time. This approach was well received by learners. Historic open modeling should be further investigated for its effectiveness in maintaining motivation and improving learner outcomes. Another future direction is to incorporate audio representations of pronunciation accuracy into the OLM.

Acknowledgements

The National Science and Engineering Research Council of Canada funded this work.

References

1. Johnson, W., Wu, S., Nouhi, Y.: Socially intelligent pronunciation feedback for second language learning. In: International Conference on Intelligent User Interfaces (IUI) Workshop in Modeling Human Teaching Tactics and Strategies, Island of Madeira (2004)
2. Archibald, J., O'Grady, W.: Contemporary Linguistic Analysis. Pearson, London (2008)
3. Johnson, W., Marsella, S., Vilhjalmsson, H.: The DARWARS Tactical Language Training System. In: Interservice/Industry Training, Simulation and Education Conference, Orlando (2004)
4. Demmans Epp, C.: ProTutor: A Pronunciation Tutor That Uses Historic Open Learner Models. University of Saskatchewan, Saskatoon, Canada (2010)
5. Lu, X., Di Eugenio, B., Kershaw, T., Ohlsson, S., Corrigan-Halpern, A.: Tutorial Dialogue Patterns: Expert vs. Non-expert Tutors. In: 3rd Midwest Computational Linguistics Colloquium, Urbana (2006)
6. Bull, S., Kay, J.: Student Models that Invite the Learner in: the SMILI:) Open Learner Modelling Framework. Int. J. Artif. Intell. Ed. 17(2), 89–120 (2007)
7. Barrow, D., Mitrovic, A., Ohlsson, S., Grimley, M.: Assessing the Impact of Positive Feedback in Constraint-Based Tutors. In: ITS, Montreal, pp. 250–259 (2008)

Does Self-Efficacy Matter When Generating Feedback?

Matt Dennis[1], Judith Masthoff[1], Helen Pain[2], and Chris Mellish[1]

[1] Department of Computing Science, University of Aberdeen, Aberdeen, AB24 3UE
[2] School of Informatics, University of Edinburgh, 10 Crichton Street, Edinburgh, EH8 9AB
{m.dennis,j.masthoff,c.mellish}@abdn.ac.uk
helen@inf.ed.ac.uk

Abstract. This study aims to establish how tutors adapt to Generalised Self-Efficacy when providing feedback on progress to a learner. Tutors seem to adapt to learners with low self-efficacy, providing a positive slant to topics on which the learner performed very badly. Results can be used by a conversational agent to adapt feedback to learners' self-efficacy.

1 Introduction

An important goal for any pedagogical agent is to keep the learner motivated throughout the learning process. Learners respond differently to feedback [1]. Human tutors attempt to mitigate this by varying learner feedback based on many factors, including their current emotional state and their abilities (e.g. [2]). We are interested in establishing which personality traits of a learner are considered by human tutors when providing feedback. In this study, we aim to discover the extent to which tutors consider a learner's General Self-Efficacy (GSE), defined as 'the extent to which a person believes they are capable of completing a task' [3]. More discussion of the literature and the goals of the wider research project can be found in [4].

2 Design of Study

This study investigates whether and how tutors adapt performance feedback to a learner's self-efficacy. We investigate whether tutors differ in their use of particular kinds of slanting for learners with high GSE compared to those with low GSE. While tutors were not explicitly told to keep the learner motivated, we assumed that they would consider learner motivation when deciding on feedback.

We used a between-subject design, where each participant was presented with a fictional student with either high or low-self efficacy, as well as a set of percentage marks representing the student's performance on a mock test. Participants then provided feedback to the student on their performance. There were 19 participants: 16 were trainee teachers and 3 were university lecturers (32% male; 74% were aged under 25; 16% 26 to 40 and 10% 41 to 65).

We used short stories to convey the student's level of GSE. The stories were based on the validated questionnaire for GSE [5] and polarized as validated in a pre-study.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 444–446, 2011.

The independent variable was the level of self-efficacy of the student: high and low, conveyed by the story. The dependent variable was the *slant* or bias that the participant employs in their feedback. For each topic, participants could say above, below or meeting expectations, and could modify this by using slightly or substantially (where applicable). To determine slants, we ran a pilot study: 3 judges were asked to state if they thought 33 responses were biased per topic, and if the overall response displayed a slant. From this we established the rules in table 1.

Table 1. Topics, Descriptions, Modifiers and slants

Topic (marks)	Description	Modifier (where applicable)	Slant
Aromathy (91%)	above	substantially, none	neutral
		slightly	negative
	meeting, behind	all	negative
Bartology (69%)	above	substantially	positive
		none, slightly	neutral
	meeting, behind	all	negative
Cleropathy (52%)	above	substantially	positive
		slightly, none	neutral
	meeting	n/a	neutral
	behind	all	negative
Deuronics (33%)	above, meeting	all	positive
	behind	substantially	negative
	behind	slightly, none	neutral
Epomathy (12%)	above, meeting	all	positive
	behind	all	neutral

A valid response contains at least one topic, and discusses each topic only once. We can calculate the overall response slant by summing the slants of the individual topics (see table 1), using 0 for neutral slants, +1 for positive and -1 for negative slants. Participants were not required to mention all topics. Compare "You are behind on Deuronomy and Eponomy", and "You are ahead on Aromathy and behind on Deuronomy and Eponomy". According to table 1, the slant will be neutral. Judges leaned towards believing that omitting passing grades provides a negative slant and omitting failing grades a positive slant, but mentioned that topics may be omitted for varying reasons.

We hypothesized that: Participants will produce more negatively slanted feedback in their responses to students with high GSE than students with low GSE (H1) and that participants will produce more positively slanted feedback in their responses to students with low GSE than students with high GSE (H2).

Procedure. Participants were shown the story about a student and this student's performance in a mock test (scores and topics shown in table 1). The scores were chosen to have roughly equal distance between them. We avoided round numbers to make them seem more realistic. Participants were told that the teacher expectation was 50% on each topic. Participants then provided feedback on the student's performance using a tool which allowed the combinations shown in table 1. If participants had chosen to omit a topic, they were to say if this was because they wanted to make the response more positive or negative (amongst other reasons, treated as neutral). This was then factored into the calculation, by using + or -1 for the omitted topic.

3 Results and Conclusion

We received 18 valid responses, 10 for High GSE and 8 for low GSE. Figure 1 shows the slants of the feedbacks produced. In correspondence with hypothesis H2, many participants in the low GSE condition provided a positive slant, however this number is not statistically significantly higher than the high GSE condition. This is likely due to the small number of participants. Investigating slants on individual topics, there is a significant difference (p<.01, t-test, Bonferroni corrected) between groups on topic E, with significantly more participants utilizing a positive slant. So, there seems reason to adapt to low GSE. There was no evidence to support hypothesis H1. The results will be used to make an algorithm for producing positive slants in feedback for low GSE learners and to study the impact of this on learner motivation.

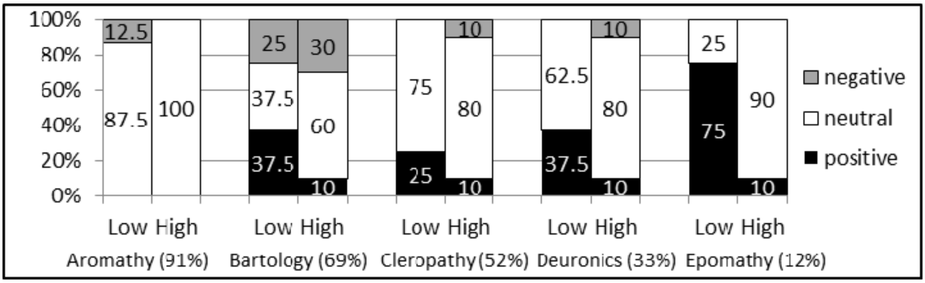


Fig. 1. Percentages of negative, neutral and positive slants applied to feedback per topic

References

[1] Robison, J.L., McQuiggan, S.W., Lester, J.C.: Modeling task-based vs. affect-based feedback behavior in pedagogical agents: An inductive approach. In: AIED 2009, pp. 25–32 (2009)

[2] Alexander, S., Sarrafzadeh, A., Hill, S.: Easy with eve: A functional affective tutoring system. In: Workshop on Motivational and Affective Issues in ITS., at ITS 2006, pp. 5–12 (2006)

[3] Mcquiggan, S.W., Mott, B.W., Lester, J.C.: Modeling self-efficacy in intelligent tutoring systems: An inductive approach. UMUAI 18(1), 81–123 (2008)

[4] Dennis, M.: Encouraging users to study more: Adapting feedback to personality and affective state. Young Researcher's Track. In: AIED 2011 (2011)

[5] Schwarzer, R., Jerusalem, M.: Generalized self-efficacy scale. In: Weinman, J., Wright, S., Johnston, M. (eds.) Measures in Health Psychology: A User's Portfolio. Causal and Control Beliefs, 1st edn., pp. 35–36, 37. NFER-NELSON, Windsor (1995)

Physiological Evaluation of Attention Getting Strategies during Serious Game Play

Lotfi Derbali and Claude Frasson

Département d'informatique et de recherche opérationnelle
Université de Montréal, 2920 Chemin de la Tour, Montréal, Canada

Abstract. This study investigated *Attention* getting strategies and their evaluation during serious game play. We proposed, therefore, the use of physical sensors, namely heart rate, skin conductance, and electroencephalogram (EEG), as well as a theoretical model of motivation (Keller's ARCS model) to evaluate two *Attention* getting strategies in a serious game environment. Results showed that some specific EEG ratios were more appropriate than others to physiologically evaluate learners' reactions. Finally, physiological evaluation of *Attention* getting strategies can relevantly provide an appropriate tool to discriminate between attentive and inattentive learners.

Keywords: *Attention* getting strategies, Keller's ARCS model of *Motivation*, physical sensors, EEG ratios.

1 Introduction

It is widely acknowledged that psychological and cognitive learners' states can affect their wills and skills in acquiring new knowledge [1]. Intelligent systems cannot, therefore, ignore the learners' states and should take them into account during learning process. One important learners' state is *Motivation* which plays a crucial role in both learners' performance and use of intelligent systems over time [2]. *Motivation* is considered a natural part of any learning process and learners need to believe that the activity will bring about some gains or sense of satisfaction [1]. Several studies have been undertaken to measure motivational learners' states. Self-report questionnaires have been the most frequently used method to assess *Motivation*. In addition, recent studies have involved a variety of physical sensors, such as camera, skin conductance (SC) or electroencephalogram (EEG), to measure *Motivation* and response to emotional and cognitive stimuli [3]. However, there are a handful of studies that have considered motivational strategies to overcome motivation problems in computer-based education context.

The present paper aims to examine the implication of different physiological sensors to evaluate *Attention* getting strategies, the first dimension of motivational strategies as defined by Keller's ARCS model of *Motivation*, and to highlight the corresponding learners' patterns. Since the serious games have been supposed an engaged environment that includes several *Attention* getting strategies [4], we use a serious game to carry out our empirical study. We also use three physiological recordings, namely heat rate (HR), skin conductance (SC) and brainwaves (EEG).

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 447–449, 2011.

2 Procedure

The ARCS model of *Motivation* [5] has been chosen to theoretically assess motivational strategies used in a serious game. Keller used existing research on motivational psychology to identify four categories to constitute the ARCS model of motivation: Attention, Relevance, Confidence, and Statisfaction. Each of the four categories also has subcategories that are useful in diagnosting learners' motivational profiles and in creating motivational tactics (or strategies) that are appropriate for the specific problems that are identified. Our approach attempts to determine the close relationship between *Attention* getting strategies and their physiological effects on learners. It starts by identifying some significant *Attention* getting strategies that are drawn from the ARCS theoretical model and evaluating their impact on learners using the physiological measures. Besides the SC and HR sensors which are typically used to study human affective states [6], we have considered relevant to use the EEG sensor in our proposed approach. EEG ratios of the different frequency bands were computed for delta/theta (δ/θ), theta/alpha (θ/α), theta/low-beta (θ/β) and alpha/low-beta (α/β).

Twenty-nine subjects (11 female), with mean age of 26.7 ± 4.1 years, were invited to play a serious game called FoodForce. Virtual tutors also accompany the player throughout the missions by offering various tips and lessons. A pre-test and post-test were also administered to compare learners' knowledge aquisition regarding the content of the serious game. The motivational measurement instrument called IMMS was used following each mission to assess *Motivation*. The SC and HR sensors were attached to the fingers of subjects' non-dominant hands, leaving the other free for the experimental task. EEG was recorded by using a cap with a linked-mastoid reference. Three selected areas (F3, C3, and Pz) were placed according to the international 10-20 system; the reference and the ground sensors were located at Cz and Fpz respectively. The EEG was sampled at a rate of 256 Hz. A Power Spectral Density (PSD) was computed to divide the EEG raw signal into the EEG ratios (δ/θ, θ/α, θ/β, and α/β).

3 Results

Subjects have been separated into two groups according to the ARCS scores after each mission: those with scores below the overall average (group "Below") and those with scores above the overall average (group "Above"). According to Keller's model, *Problem Solving* and *Alarm Trigger* are two *Attention* getting strategies. To evaluate *Problem Solving* strategy used for example by the virtual companion in the fifth mission of the serious game, we have considered the *Attention* scores related to this mission in order to determinate the "Below" and "Above" groups of subjects and compare their physiological reactions. The same procedure has been applied for the Alarm Trigger strategy. The physiological reactions for *Alarm Trigger* and *Problem Solving* strategies between the two groups are presented in Figure 1.

General results showed that SC and HR reach their limits in some cases for evaluating learners' reactions. In fact, no clear trends were found in SC and HR for evaluating Attention getting strategies (*Problem Solving* and *Alarm Trigger*). Conversely, EEG ratios, especially θ/β and α/β, have showed different trends for these strategies. They tended to decrease for subjects of "Above" group whereas they have shown opposite trends for those of "Below" group. They can provide an objective evaluation of motivational strategies for distinguishing between learners' reactions.

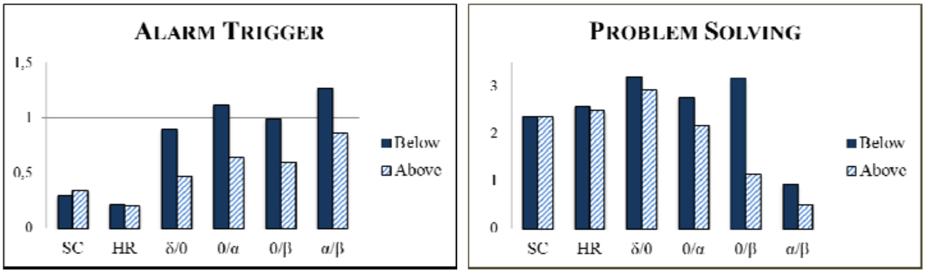


Fig. 1. Example of physiological reactions (two Attention getting strategies)

An explanation of these different trends between the two groups can be given by neuroscience. For example, a negative correlation exists between the θ/β ratio and the attention level of adults [7]. Furthermore, the ratio of α/β waves has been used as an indication of relaxation and better concentration and relaxed learners' states are indicated by increased β; decreased α; so decreased α/β ratio.

4 Conclusion

In this paper, we have assessed the effects of the first dimension of motivational strategies, namely *Attention* getting strategies, in a serious game using the ARCS theoretical model as well as three physiological sensors: HR, SC and EEG. We have successfully identified physiological patterns, especially EEG θ/β and α/β ratios, to evaluate these strategies and to possibly distinguish between attentive and inattentive learners. The integration of these results into an intelligent tutoring system, for example, can enrich the learner model and adapt the motivational interventions of the tutor model.

Acknowledgments

We thank the CRSNG, the FQRSC and the Tunisian Government for their support.

References

1. Bandura, A.: Social foundations of thought and action: A social cognitive theory. Prentice-Hall, Englewood Cliffs (1986)
2. de Vicente, A., Pain, H.: Informing the detection of the students' motivational state: An empirical study. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 933–943. Springer, Heidelberg (2002)
3. Derbali, L., Frasson, C.: Prediction of Players Motivational States Using Electrophysiological Measures during Serious Game Play. In: ICALT 2010, pp. 498–502 (2010)
4. Prensky, M.: Digital Game-Based Learning. McGraw Hill, New York (2001)
5. Keller, J.M.: Motivational design for learning and performance: The ARCS model approach. Springer, New York (2010)
6. Lin, T., Imamiya, A., Hu, W., Omata, M.: Display Characteristics Affect Users' Emotional Arousal in 3D Games. Universal Access in Ambient Intelligence Environments (2007)
7. Putman, P., van Peer, J., Maimari, I., van der Werff, S.: EEG theta/beta ratio in relation to fear-modulated response-inhibition, attentional control, and affective traits. Biological Psychology 83, 73–78 (2010)

Does Topic Matter? Topic Influences on Linguistic and Rubric-Based Evaluation of Writing

Nia Dowell, Sidney K. D'Mello, Caitlin Mills, and Art Graesser

Department of Psychology, Institute for Intelligent Systems, The University of Memphis,
Memphis TN 38152 USA
{ndowell,sdmello,cmills2,graesser}@memphis.edu

Abstract. Although writing is an integral part of education, there is limited knowledge on how assigned topics influence writing quality both in terms of micro-level linguistic features and macro-level subjective evaluations by human judges. We addressed this question by conducting a study in which 44 students wrote short essays on three different topics: traditional *academic-based* topics such as the ones used in standardized tests, *personal emotional experiences*, and *socially charged* topics. The essays were automatically scored on five linguistic dimensions (*narrativity*, *situation model cohesion*, *referential cohesion*, *syntactic complexity*, and *word abstractness*). They were also manually scored by human judges based on a rubric focusing on macro-level dimensions (i.e., introduction, thesis, and conclusion). The results indicated that topic-related differences were observed on both the rubric-based and linguistic assessments, although there were weak relationships between these two measures.

Keywords: Writing quality, Linguistics, Coherence, Coh-Metrix, Cohesion.

1 Introduction

Considering the high stakes placed on writing competency in the 21st century, it is not surprising that computational systems utilizing natural language processing techniques have been developed to automatically score written essays and provide interventions to promote writing proficiency (e.g., Intelligent Essay Grader, E-Rater, Summary Street, and Writing Pal). However, little is known about what factors influence the quality of writing; an area that could potentially benefit the advancement of such systems.

Some research has demonstrated that writing quality may be influenced by the topic the individual is writing about [e.g., 1]. A satisfactory understanding of topic influences on writing quality is necessary to ensure that automated writing interventions are optimally beneficial to students. The present research addressed this issue by examining the degree to which both linguistic features and rubric-based assessment scores vary as a function of essay topic. We collected a corpus of essays on three topics and scored the essays using a holistic rubric and Coh-Metrix, an automated text analysis tool that evaluates texts on a number of dimensions [2].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 450–452, 2011.

2 Methods

The participants were 44 undergraduates who participated for course credit. The study had a within-subjects design in which the participants were asked to write essays on three topics, namely *socially charged issues* (e.g., abortion, death penalty), *personal emotional experiences* (e.g., write about a happy experience), and *traditional academic prompts* (e.g., debates about extending high school) similar to ones a student might encounter on standardized tests. Within each topic, participants were presented with a number of subtopics and were asked to write for 10 minutes on a subtopic of their choice. A computer interface was used to facilitate typing of the essays. Texts from the 132 essays were saved for offline analyses.

Computational Evaluation. The following is a description of the five primary Coh-Metrix 2.0 [2] dimensions that were used to automatically score the essays. *Narrativity breakdowns* refer to deviations from a sequence of episodes with actions and events that convey a story. S*ituation model cohesion* and *referential cohesion breakdowns* occur when there are problems associated with text that are not cohesively connected at a deeper conceptual level or have little overlap in words and ideas, respectively. *Syntactic complexity* refers to structurally dense and embedded sentences that are difficult to process. Finally, *word abstractness* pertains to the extent to which the text contains abstract words (e.g., democracy) compared to words that are more concrete (e.g., table). It should be noted that the Coh-Metrix measures refer to textual problems, so higher numbers indicate either breakdowns in particular dimensions, more complexity, or greater abstractness. It is hypothesized that an essay that is clear should score lower on all these dimensions.

Human Evaluation. Two trained raters (interrater reliability r = 0.9) evaluated the essays using a holistic rubric [3], which is similar to the standardized rubric used in assessing essays on the SAT. The overall score was on a 6-point scale with a score of 1 indicating little or no mastery and a 6 indicating clear and consistent mastery. Note that the scores were standardized among each judge to remove any potential bias.

3 Results and Discussion

A repeated-measures MANOVA was performed to investigate the effect of topic on the five Coh-Metrix dimensions. The analysis revealed there was a significant main effect for essay topic, $F(2, 86) = 11.08$, $p < .001$. Posthoc tests with Bonferroni correction were conducted to identify significant ($p < .05$ for all analyses unless specified otherwise) differences across topics.

The results indicated that students' *academic* essays (M = -.68, SD = .70) had the highest frequency of *narrativity breakdowns*, when compared to *socially charged* (M = -.98, SD = .85) and *personal emotional experience* essays (M = -1.8, SD = .88). However, *academic* essays contained less referential cohesion breakdowns (M = -.69, SD = .89) when compared to the *socially charged* essays (M = -.27, SD = .75).

Students' *personal emotional experience* essays were characterized by story-like features (less *narrativity breakdowns*). However, these essays were also accompanied by more complex syntax (*M* = 1.0, *SD* = .70) than the *socially charged* (*M* = .62, *SD* = .74) and *academic* essays (*M* = .67, *SD* = .83). Students also used significantly more concrete words when writing about a *personal emotional experience* (*M* = .17, *SD* = 1.0) compared to a *socially charged* topic (*M* = 1.1, *SD* = .85).

Essays on *socially charged* topics (*M* = -.98, *SD* = .85) had less narrative-like features than essays on *personal emotional experiences* (*M* = -1.8, *SD* = .88). *Socially charged* essays were also characterized by more abstract words (*M* = 1.1, *SD* = .85) than essays on both *personal emotional experiences* (*M* = .17, *SD* = 1.0) and *academic* topics (*M* = .47, *SD* = .69).

An ANOVA on the rater-provided essay scores indicated that overall scores varied as a function of topic $F(2, 84) = 8.23$, $MSE = .398$, $p < .001$. Posthoc tests indicated that the *socially charged* essays (*M* = -.32, *SD* = .88) were rated lower than the *academic* essays (*M* = .16, *SD* = .95) and the *personal emotional experience* essays (*M* = .15, *SD* = .94), which were on par with each other.

We examined the relationship between the two different measures of essay quality by computing a 5 × 3 (Coh-Metrix measure × topic) matrix with each cell representing the Pearson's correlation between a linguistic feature and a rubric-based score for a particular topic. The mean absolute correlation was .14, which signifies a small relationship between linguistic and rubric-based evaluations.

4 Conclusions

The results presented here indicate that essay topic can have an impact on writing quality, in terms of both the micro-level linguistic features as well as the more macro-level rubric-based assessments. In line with this, computational systems aiming to advance students writing proficiency can undoubtedly benefit from taking into account such topic-related writing influences.

Acknowledgments. This research was supported by the National Science Foundation (ITR 0325428, HCC 0834847, BCS 0904909, DRK-12-0918409), the Institute of Education Sciences (R305G020018, R305A080589), The Gates Foundation, and START (DHSZ934002/UTAA08-063). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] Beers, S.F., Nagy, W.: Syntactic Complexity as a Predictor of Adolescent Writing Quality: Which Measures? Which Genre? Reading and Writing: An Interdisciplinary Journal 22, 185–200 (2009)

[2] Graesser, A.C., McNamara, D.S., Louwerse, M.M., Cai, Z.: Coh-Metrix: Analysis of Text on Cohesion and Language. Behavior Research Methods, Instruments, and Computers 36, 193–202 (2004)

[3] McNamara, D.S., Crossley, S.A., McCarthy, P.M.: Linguistic Features of Writing Quality. Written Communication 27, 57–86 (2010)

Thinking with Your Hands: Interactive Graphical Representations in a Tutor for Fractions Learning

Laurens Feenstra[1,2], Vincent Aleven[1], Nikol Rummel[3], Martina Rau[1], and Niels Taatgen[2]

[1] Carnegie Mellon University, HCI Institute, Pittsburgh PA, USA
[2] University of Groningen, Artificial Intelligence, Groningen, The Netherlands
[3] Ruhr-Universität Bochum, Educational Psychology, Bochum, Germany

Abstract. Learning with multiple graphical representations is effective in many instructional activities, including fractions. However, students need to be supported in understanding the individual representations and in how the representations relate to one another. We investigated (1) whether interactive manipulations of graphical representation support a deeper understanding of the representations compared to static graphics and (2) whether connection-making activities help students better understand the relations between representations. In a study with 312 4th and 5th grade students we found that interactive representations were indeed more effective in improving student fraction learning compared to static fraction graphics, especially for students yet unfamiliar with the topics being taught. We found no effect for connection-making activities. The results suggest that domains with (multiple) representations are best taught with tutor-guided student manipulation of these graphics rather than with static pictures.

Keywords: Interactive representations, connection making activities, virtual manipulatives, situational feedback.

1 Introduction and Method

The educational psychology literature indicates that multiple representations are useful for learning, provided they are task specific and provided students make connections between them [1]. Fractions are a challenging topic [5] with multiple task-appropriate graphical representations, such as circles, rectangles and number lines being in widespread use. However, very little experimental work has investigated how multiple representations can best be supported effectively in fractions learning [5], and in particular with technology.

The current study addresses the following hypotheses: (1) interactive representations support robust learning better than static representations and (2) support for connection making leads to more robust learning than providing multiple representations without such support.

A total of 312 4th and 5th grade students in three US elementary schools participated in the study during their regular mathematics instruction. Students worked with

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 453–455, 2011.

different versions of an example-tracing tutor designed and implemented specifically for this study using the Cognitive Tutor Authoring Tools (CTAT) [3]. The tutor is available online on the *MathTutor* website (https://mathtutor.web.cmu.edu) [2]. Students were randomly assigned to one of four conditions according to a 2x2 design with the following two factors: interactive versus static representations, and connection-making activities versus no connection-making activities.

We assessed students' knowledge on fractions using equivalent pretest, immediate and delayed posttest versions, each of which took 30 minutes to complete. For analysis, we used a hierarchical mixed-effects model [4] with five nested factors to explain part of the variance in the data.

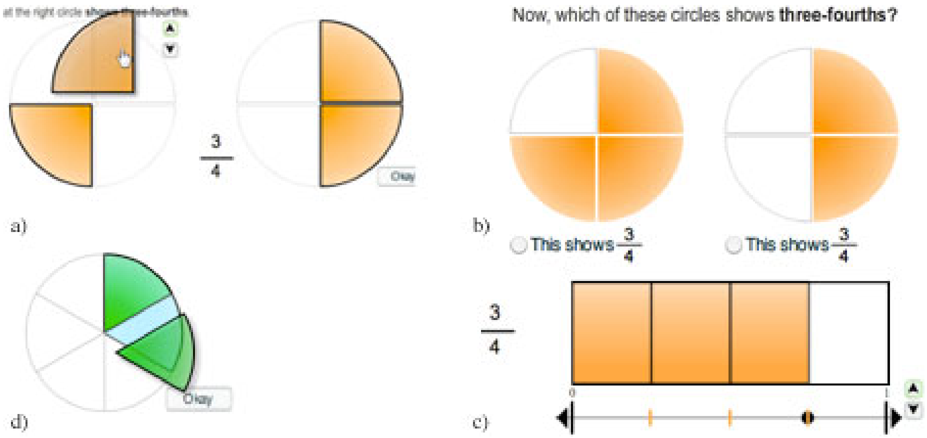

at the right circle shows three-fourths
3
4
Okay
a)
Now, which of these circles shows three-fourths?
b)
This shows 3/4
This shows 3/4
Okay
d)
3
4
0
1
c)

Fig. 1. *From top-left clockwise: a)* Interactive (circle) representations. Student partition the graphical representation using the button controls, drag pieces to the right circle to show the numerator and then press the "Okay" button to be graded. *b)* The same problem with static pictures and multiple-choice. *c)* Connection-making activities: student attention is directed to shared representational features, in this case the rectangle and numberline have the same relative distance from the left on the x-axis. *d)* Situational feedback: students are asked to find an equivalent fraction to one third. By dropping the darker pieces (sixths) *on top* of the one third they find 2/6 to be an equivalent fraction to one third.

2 Results and Conclusion

Across conditions, students scored an average of 24.2% better on the immediate posttest ($t = 9.920$, $p < .001$, $d = 1.42$) than on the pretest, an increase in performance they retained on the delayed test ($t = 12.338$, $p < .001$, $d = 1.30$). Students working with interactive representations scored 10.1% better than the group learning with static graphics ($t = 2.471$, $p < .02$, $d = 0.46$). The interactivity main effect was not significant for the delayed posttest ($t = 1.430$, $p = .10$). There was no significant main effect for connection-making activities on either of the posttests, although differences between conditions at the delayed posttest were bordering the level of significance ($t = 1.376$, $p = .09$).

We analyzed the data separately for 4th grade students and 5th grade students, in order to investigate whether the amount of prior fractions instruction influenced the effectiveness of the tutor, or that of the experimental factors. Both the 4th grade students' and 5th grade students' performance increased on the posttests ($t = 8.629$, $p < .001$), compared to the pretest. The effect of interactive representations on performance holds only for 4th-graders. The 4th-grade interactive group performed 20.4% better than the static group ($t = 4.235$, $p < .001$, $d = 1.06$) on the immediate posttest, and on the delayed test, the interactive group outperformed the static group by 16.1% ($t = 2.044$, $p < .04$, $d = 0.60$). Interactive representations have no additional effect on the learning of 5th grade students.

As the present study revealed, our example-tracing tutor for fractions learning with multiple graphical representations led to significant learning gains across conditions for all fractions topics addressed in the tutor curriculum. Interactive manipulating of fraction representations leads to an additional performance increase, especially for 4th-grade students and students with low prior fraction knowledge.

Besides fractions, many other topics use representations to visualize and clarify the main concepts. Allowing the students to interactively manipulate the relevant features, supported by both situational and tutored feedback may very well achieve the same learning effect as it does for fractions.

Acknowledgements. This work was funded by the National Science Foundation, award number SBE-0354420 and NSF (REESE) Award No. DRL-0910010. We would like to thank Ken Koedinger, Zelha Tunc Pekkan, Mitchell Nathan, Katherine Cramer, Margaret Smith, Eric Paulos, Jay Raspat the Datashop and CTAT teams, Cassandra Studer, and the students, teachers, and school principals.

References

1. 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)
2. Aleven, V., McLaren, B.M., Sewall, J.: Scaling up programming by demonstration for intelligent tutoring systems development: An open-access website for middle-school mathematics learning. IEEE Transactions on Learning Technologies 2(2), 64–78 (2009), `http://www.computer.org/portal/web/csdl/doi/10.1109/TLT.2009.22`
3. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A new paradigm for intelligent tutoring systems: Example-tracing tutors. Journal for Research in Mathematics Education 5(24), 428–441 (2009); 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)
4. Gelman, A., Hill, J.: Data Analysis using Regression and Multilevel/Hierarchical Models. Cambridge University Press, New York (2007)
5. Rau, M.A., Aleven, V., Rummel, N.: Blocked versus Interleaved Practice with Multiple Representations in an Intelligent Tutoring System for Fractions. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 413–422. Springer, Heidelberg (2010)

Classification Techniques for Assessing Student Collaboration in Shared Wiki Spaces

Chitrabharathi Ganapathy, Jeon-Hyung Kang, Erin Shaw, and Jihie Kim

Southern California, Information Sciences Institute
4676 Admiralty Way, Marina del Rey, CA, 90292 USA
{cganapat,jeonhyuk}@usc.edu, {shaw,jihie}@isi.edu

Abstract. This paper presents the case study of collaboration analysis in the context of an undergraduate student engineering project. Shared Wiki spaces used by students in collaborative project teams were analyzed and the paper presents new techniques, based on descriptive statistics and the Labeled Latent Dirichlet Allocation (LLDA) model for multi-label document classification, to assess quality of student work in shared wiki spaces. A link is shown between processes of collaboration, performance and work pace.

Keywords: Collaborative learning assessment, Wiki Assessment, Topic Modeling, Labeled Latent Dirichlet Allocation, Descriptive Statistics.

1 Introduction

Wikis are collaborative knowledge building environments that have been shown to promote collaborative learning [1][2][3], however, the results of Wiki use in academia have been mixed [1][4],and patterns of student Wiki use in engineering courses and their effect on learning have been challenging to assess. The goal of the work presented here is to make progress towards closing the 'assessment' gap, that is, to develop techniques to assist instructors and educational researchers in evaluating student performance in the context of an on-line collaborative learning environment, the shared Wiki space.

2 Wiki Document Classification and Assessment Using Labeled LDA Topic Model and Descriptive Statistics

Latent Dirichlet Allocation [6] based classification is a powerful tool for analyzing latent topics in documents, but it has all the disadvantages inherent to any unsupervised model. In this experiment, wiki pages were classified by page title and topic modeling tags generated using the Labeled Latent Dirichlet Allocation (LLDA) [7] topic model. In this experiment, a single topic hierarchy and label set was generated by manually analyzing the course curriculum and content of the Wiki pages across all the project groups. LLDA results were used to classify the documents according to the topic hierarchy and descriptive statistics measures like number of pages, amount

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 456–458, 2011.

of content for different topics were used in assessment of student work. Figure1 shows the topic hierarchy that represents the major types of the documents generated by students over the course. The two main topic categories are team management and software engineering principles.

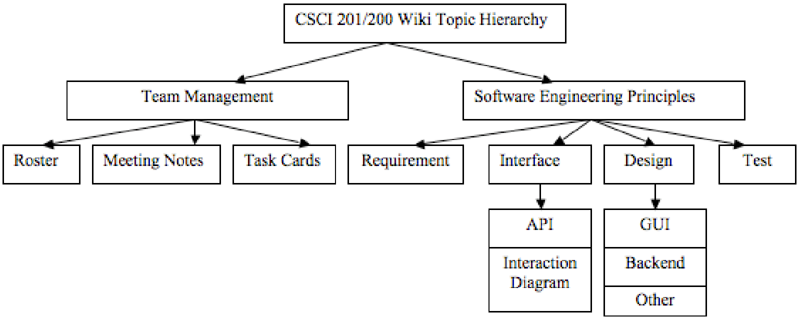
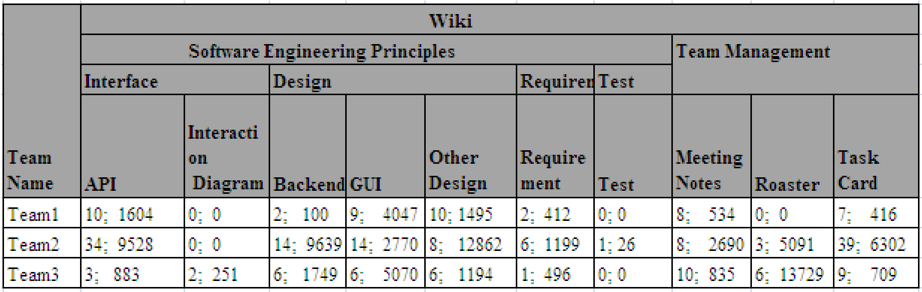


Fig. 1. Topic hierarchy used for the wiki document classification

2.1 Comparison of LLDA and Descriptive Statistics Results

Three randomly drawn teams of students from undergraduate courses working on collaborative programming projects were used in the case study. Table 1 shows the number of documents and content (number of words) in each page classified according to the primary topic. The number of words for each topic category can be used along with the number of pages under that topic category to understand the quality of the group Wiki. Overall, Team1 had the smallest number of documents in Wiki; furthermore they had incomplete backend design and some program topics such as Integration diagram and Test were not found in the Team1 wiki.

Table 1. Number of pages and amount of content (in words) generated by the three teams

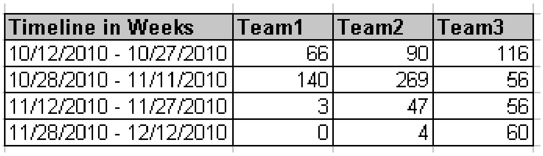


	Wiki									
	Software Engineering Principles							Team Management		
	Interface		Design			Requiren	Test			
Team Name	API	Interaction Diagram	Backend	GUI	Other Design	Requirement	Test	Meeting Notes	Roaster	Task Card
Team1	10; 1604	0; 0	2; 100	9; 4047	10; 1495	2; 412	0; 0	8; 534	0; 0	7; 416
Team2	34; 9528	0; 0	14; 9639	14; 2770	8; 12862	6; 1199	1; 26	8; 2690	3; 5091	39; 6302
Team3	3; 883	2; 251	6; 1749	6; 5070	6; 1194	1; 496	0; 0	10; 835	6; 13729	9; 709

2.2 Comparison of Wiki Activity Timeline

The number of updates done by students to each topic category can be benchmarked and low level of activity can be detected to provide immediate feedback to the instructor and the project group during the course of the project regarding their need to increase level of activity to be at par with other groups. Overall, Team1 got the lowest grade (out of the three teams analyzed); they started early but did not have much activity during the last 4 weeks of the project. Team2 had considerable activity levels for the first 6 weeks. Team3 got the highest score; they started early and worked almost uniformly throughout the course of the project.

Table 2. Timeline of edits to software engineering topic pages for bi-weekly intervals



Timeline in Weeks	Team1	Team2	Team3
10/12/2010 - 10/27/2010	66	90	116
10/28/2010 - 11/11/2010	140	269	56
11/12/2010 - 11/27/2010	3	47	56
11/28/2010 - 12/12/2010	0	4	60

3 Conclusion

The LLDA model based classification with descriptive measures like number of pages, amount of content and timeline of activity can be used to understand productive work patterns and to generate feedback that can help students to stay on track during the course of the project. Future directions would involve improving the accuracy of the labeled LDA model and developing techniques to remove spam or irrelevant content from the wiki.

Acknowledgments. The authors wish to thank the course instructors, USC Senior Lecturers David Wilczynski, Ph.D. and Michael Crowley, Ph.D. whose data and patience were invaluable. This work was supported by a National Science Foundation, CCLI-1 grant (#0941950).

References

1. Rick, J., Guzdial, M.: Situating CoWeb: a scholarship of application. Computer-Supported Collaborative Learning 1, 89–115 (2006)
2. Ben-Zvi, D.: Using Wiki to Promote Collaborative Learning in Statistics Education. Technology Innovations in Statistics Education 1(1), article 4 (2007)
3. Chen, H.L., Cannon, D.M., Gabrio, J., Leifer, L.: Using Wikis and Weblogs to Support Reflective Learning in an Introductory Engineering Design Course. In: Paper presented at the 2005 American Society for Engineering Education Annual Conference (June 2005)
4. Wang, H.-C., Lu, C.H., Yang, J.-Y., Hu, H.-W., Chious, G.-F., Chiang, Y.-T., Hsu, W.L.: An Empirical Exploration of Using Wiki in an English as a Second Language Course. In: Proceedings of the Fifth IEEE Int'l Conf. on Advanced Learning Technologies (2005)
5. Blei, D.M., Andrew, Y.N., Jordan, M.I.: Latent Dirichlet Allocation. Journal of Machine Learning Research 3, 993–1022 (2003)
6. Ramage, D., Hall, D., Nallapati, R., Manning, C.: Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the Empirical Methods in Natural Language Processing Conference (2009)

A Common Model of Didactic and Collaborative Learning for Theory-Aware Authoring Support

Yusuke Hayashi[1], Seiji Isotani[2], Jacqueline Bourdeau[3], and Riichiro Mizoguchi[4]

[1] Information Technology Center, Nagoya University, Japan
[2] The Institute of Mathematics and Computational Sciences, University of Sao Paulo, Brazil
[3] LICEF research center, TÉLUQ-UQAM, Canada
[4] The Institute of Scientific and Industrial Research (ISIR), Osaka University, Japan
hay@icts.nagoya-u.ac.jp

Abstract. This paper proposes an ontological model that is a flexible framework to create learning scenarios blending didactic and collaborative learning. This model enables us to describe the design rationale of such learning scenarios and to organize theoretical knowledge for designing such scenarios in the same manner.

1 Introduction

In practice, some lessons have well-thought-out linkage of different forms of learning such as didactic, inquiry and collaborative learning, and effectively achieve multiple learning goals such as cultivating an attitude toward learning, acquiring domain knowledge, developing communication skill and so on. However, it is difficult for teachers to design such lessons rationally because few studies have explored the potential to connect different forms of learning effectively.

The goal of this study is to provide teachers with authoring systems for functionally-relevant blending of various forms of learning in a learning scenario. Especially, this study currently focuses on didactic and collaborative learning. Here, by didactic learning, we mean learning following the “traditional” model of a teacher-student relationship. This paper proposes an ontological model of didactic and collaborative learning based on two ontologies, OMNIBUS ontology [1] for authoring of didactic learning and Collaborative Learning (CL) ontology [2] for authoring of collaborative learning. This framework provides following functionalities; (1) linking didactic and collaborative learning process in a learning scenario in a manner consistent with the goals of the scenario, and (2) accumulating and utilizing design knowledge of learning scenarios from theory and practice.

The structure of this paper is as follows. The next section proposes an ontological model to describe didactic and collaborative learning scenarios and design knowledge for them. Section 3 illustrates functionalities of theory-aware authoring system based on ontological models of learning/instructional theories for didactic and collaborative theories. Finally, the last section concludes and discusses future direction of this study.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 459–461, 2011.

2 A Common Model of Didactic and Collaborative Learning

The major differences between didactic and collaborative learning are existence or nonexistence of teacher, which comes from the difference between the employed principles. In order to build a common model, this study equate "instruction" by teacher in didactic learning with "collaboration" by learners in collaborative learning. Both can be considered as actions facilitating stakeholders' learning as discussed in [3].

Fig. 1(a) illustrates a typical model of didactic learning where the action by the teacher, which is "instruction", facilitates learning of learners. Fig. 1(b) illustrates a typical model of collaborative learning in which each learner's action in collaboration facilitates others' learning Fig. 1(c) illustrates another model of collaborative learning in which a leaner's action facilitates not only the other's learning but also the learner's learning. Like this, both types of learning can be modeled in a common framework.

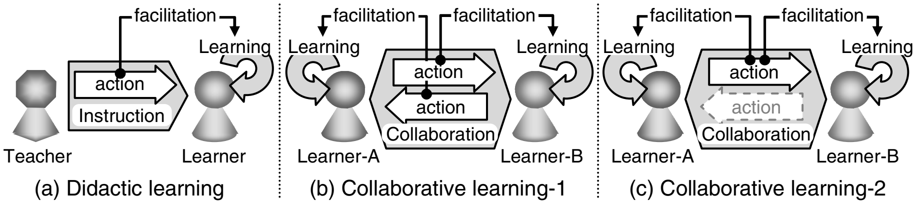


Fig. 1. Models of didactic and collaborative learning

We can implement this modeling by the concept of "I_L event" that is the common concept defined in OMNIBUS and CL ontology mentioned above. I_L event enables us to make clear description of the relation between actions and learning based on the consideration above. For further details on the definition of I_L event, see [1].

3 A Theory-Ware Authoring Support for Blended Learning

This study develops a theory-aware authoring system that functionally links didactic and collaborative learning in a learning scenario based on findings of development of authoring systems, CHOCOLATO for collaborative learning and SMARTIES for didactic learning, and has developed the usefulness of each system in practice.

Fig 2 shows a user interface of the prototype system. Here, a user is making a didactic learning part followed by collaborative learning part in a learning scenario. Fig. 2(A) shows a scene that the user has made the didactic learning part. Each node represents an event in the scenario, and the sequence of them from the left to the right represents the flow of the scenario. The hierarchical structure of events represents the design rationale of the scenario. In this system, authoring a scenario is to decompose events from one representing the goal of the entire scenario to expected actions done by participants in the scenario.

When the user adds an event following the didactic learning part, the authoring system can suggest how to decompose it. Fig. 2(B) and (C) show examples of suggestion. One comes from didactic learning theory and the other from collaborative learning one. The user can choose one out of these suggestions or describe his/her

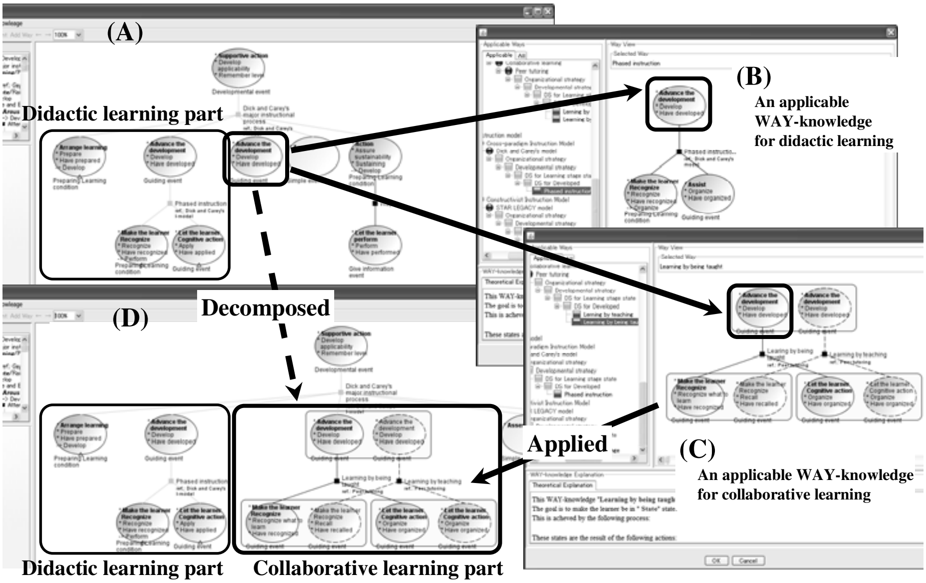


Fig. 2. User interface of the prototype system

own idea. In Fig. 2(D), the user chooses collaborative learning, and then the system adopts it to the scenario. In this way, didactic and collaborative learning can be combined in a learning scenario.

4 Conclusion

This paper proposes a common modeling framework for didactic and collaborative learning. This framework enables us to describe learning scenarios combining didactic and collaborative learning in a manner consistent with the goals of the scenario. However, some open issues remain. The most significant is the necessity of consistency of combination of the two types of learning. There is no theory concerning the combination of different types of learning in a scenario. This study would make a contribution to accumulating design knowledge about the combination of different types of learning.

References

1. Hayashi, Y., Bourdeau, J., Mizoguchi, R.: Using Ontological Engineering to Organize Learning/Instructional Theories and Build a Theory-Aware Authoring System. International Journal of Artificial Intelligence in Education 19(2), 211–252 (2009)
2. 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)
3. Reigeluth, C.M., Carr-Chellman, A.A.: Understanding Instructional Theory, Instructional-design theories and models: Building a Common Knowledge Base, pp. 3–26. Routledge, New York (2009)

Carelessness and Goal Orientation in a Science Microworld

Arnon Hershkovitz*, Michael Wixon, Ryan S.J.d. Baker,
Janice Gobert, and Michael Sao Pedro

Department of Social Science and Policy Studies, Worcester Polytechnic Institute**
arnonh@wpi.edu

Abstract. In this paper, we study the relationship between goal orientation within a science inquiry learning environment for middle school students and carelessness, i.e., not demonstrating an inquiry skill despite knowing it. Carelessness is measured based on a machine-learned model. We find, surprisingly, that carelessness is higher for students with strong mastery or learning goals, compared to students who lack strong goal orientation.

Keywords: carelessness, goal orientation, educational data mining, science inquiry.

Introduction

In recent years, there is increasing evidence that the goals students have during learning play a key role in their learning outcomes. These goals might impact learning by creating different forms of disengagement, but it is yet unclear which forms of disengagement are influenced by students' goals. One such a disengagement behavior is carelessness, i.e., when a student fails in answering a question despite knowing the answer [1]. Both mastery goals (the goal of learning), and performance-approach goals (the goal of demonstrating competence) are positively correlated with persistence and effort and correlated with self-regulated learning (SRL) strategies, hence it seems reasonable to hypothesize that carelessness will be less frequent when students have mastery or performance-approach goals. Within this paper, we operationalize carelessness using an automated detector of contextual slip, i.e., the probability that the student performed incorrectly at a specific time despite knowing the needed skill [2]. The notion of contextual slip matches previous carelessness definitions [e.g., 1], but is easier to apply than previous operational definitions. Our detector uses a log-based machine-learned model, hence can be scaled without being overly time-consuming.

1 Methodology

The learning environment. We study carelessness in demonstrating science inquiry skills (e.g., control for variable strategy). Our phase change activity enables students

* Corresponding author.

** Research was supported by NSF grant DRL-1008649 (to Janice Gobert and Ryan Baker). The authors thank Juelaila Raziuddin, Andy Montalvo, and Adam Nakama, for their support in collecting the data.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 462–465, 2011.

to use inquiry support tools while engaging in authentic inquiry using "microworlds", computer simulated worlds in which a student can conduct scientific inquiry. This learning environment detects whether students demonstrate inquiry skills using validated machine-learned models of these behaviors [3].

Participants and Data Set. 148 eighth grade students, aged 12-14 years old, from a public middle school in Central Massachusetts. All students' fine-grained actions were logged and then analyzed at the "clip" level; a clip is a consecutive set of a student's actions describing activity in its context.

The data set includes 2114 phase change clips in which the student failed to correctly demonstrate one or more of three inquiry skills: designing controlled experiments using the control for variable strategy (CVS), testing articulated hypotheses, and planning using the table tool. Each clip had a set of 73 features extracted for the machine-learning process, including the numbers of different types of actions that occurred during the clip, the timing of each action, and the probability that the student knew the skill to solve the relevant problem set before their first attempt on action N, $P(L_{n-1})$ (calculated using a Bayesian Knowledge Tracing model of student inquiry skill). In addition, students completed standard questionnaires for the Patterns of Adaptive Learning Scales (PALS) survey [4].

Carelessness Detector. We developed the carelessness detector in RapidMiner 5.0 using REPTree, a regression tree classifier. Carelessness, first predicted at the clip-level, was computed at student-level by taking average values over all of the student clips. The resulting regression tree (a 6-fold cross-validation correlation of r=0.63) includes 13 variables, has a size of 35 and a total depth of 13.

Cluster Analysis. Exploratory cluster analysis was conducted to group the students by their PALS measures in order to examine whether certain sub-groups of students which manifest specific characteristic patterns on the PALS survey also differ on carelessness. We used Two-step Cluster Analysis (in SPSS 17.0) with the PALS measures (Z-standardized) and a log-likelihood distance measure. We chose k=3 as it led to more interesting separations between aspects of the PALS.

2 Results

Overall, mean carelessness across clips (N=2114) was 0.05 (SD=0.16). The predicted carelessness across students (N=130) had a mean of 0.06 (SD= 0.05).

Carelessness and PALS Measures. Three of the 8 sub-scales of the PALS survey were significantly correlated with carelessness: a) Carelessness was positively correlated with *academic efficacy* with r=0.24, F(1,121)=7.10, p<0.01; b) Carelessness was negatively correlated with *disruptive behavior* with r=-0.22, F(1,121)=5.96, p<0.01; and c) Carelessness was negatively correlated with *self-presentation of low achievement* with r=-0.23, F(1,121)=6.49, p<0.05.

Carelessness and PALS-based Clusters. In general, cluster analysis suggested that certain patterns of response on the PALS survey might predict carelessness measures. Mean values of the clustering variables are given in Table 1, according to which we named the clusters: 1) *mastery goal orientation,* 2) *performance goal orientation*, and 3) *lack of goal orientation*.

Table 1. Centers of the clusters formed by Two-step Cluster Analysis with k=3 (N=121)

Variable	Mean (std)		
	Cluster 1	Cluster 2	Cluster 3
Mastery goal orientation	4.66 (0.40)	4.38 (0.64)	2.07 (0.87)
Performance-approach goal orientation	1.69 (0.57)	3.20 (1.04)	2.40 (0.82)
Performance-avoid goal orientation	1.86 (0.72)	3.78 (0.67)	3.62 (0.68)
Academic efficacy	4.41 (0.49)	4.22 (0.55)	3.65 (1.06)
Avoiding novelty	1.96 (0.60)	2.58 (1.00)	3.02 (1.21)
Disruptive behavior	1.54 (0.68)	1.61 (0.68)	2.07 (1.01)
Self-presentation of low achievement	1.33 (0.31)	1.59 (0.60)	3.43 (1.00)
Skepticism about the relevant of school for future success	1.57 (0.49)	1.92 (0.82)	2.07 (0.87)
N	**35**	**66**	**20**
Mean Carelessness (SD)	**0.06 (0.06)**	**0.06 (0.05)**	**0.03 (0.02)**

Mean carelessness in cluster 3 was significantly lower from its mean in both cluster 1, with t(45.94)=2.78, p<0.0, and cluster 2, with t(76.17)=3.86, p<0.01. For both analyses, the F of Levene's Test of Equality of Variances was significant at p<0.05, hence equal variances were not assumed. No significant differences were found between clusters 1 and 2, t(99)=0.12, p=0.90.

Our results surprisingly suggest that students with strong mastery/performance goal orientation were on average twice as careless as those with no goal orientation. We compared inquiry skills between clusters, as measured by $P(L_{n-1})$ (averaged over time for each student, then over each cluster). There were no significant differences in mean inquiry skills between clusters 1 and 3, t(53)=0.06, p=0.95; nor between clusters 2 and 3, t(84)=1.08, p=0.29. Hence, differences in carelessness between clusters are not likely to be due to differences in student inquiry skills.

3 Summary

In summary, the research presented here shows that students characterized by mastery or performance goal orientation have (on average) double the probability of carelessness as compared to students characterized by low scores for these goal orientations. One possible interpretation of the results is that students with higher amounts of mastery or performance goals succeed in learning and correspondingly become more confident (as suggested in [1]), and that this confidence leads to carelessness despite their goal orientation. Further research regarding the ways that goal orientation relates to student behaviors within educational software may have the potential to better elucidate the mechanisms by which goal orientation impacts learning and, in turn, long-term learning outcomes.

References

[1] Clements, M.A.: Careless errors made by sixth-grade children on written mathematical tasks. Journal for Research in Mathematics Education 13, 136–144 (1982)

[2] Baker, R.S.J.d., Corbett, A.T., Aleven, V.: More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian Knowledge Tracing. In: Proceedings of the 9th International Conference on Intelligent Tutoring Systems, p. 2008.

[3] Sao Pedro, M.A., Baker, R.S.J.d., Gobert, J.D., Montalvo, O., Nakama, A.: Using machine-learned detectors of systematic inquiry behavior to predict gains in inquiry skills. In: User Modeling and User-Adapted Interaction (Revise and Resubmit)

[4] Midgley, C., Maehr, M.L., Hruda, L.Z., Anderman, E., Anderman, L., Freeman, K.E., Gheen, M., Kaplan, A., Kumar, R., Middleton, M.J., Nelson, J., Roeser, R., Urdan, T.: Manual for the Patterns of Adaptive Learning Scale. University of Michigan, Ann Arbor (2000)

Kit-Build Concept Map for Automatic Diagnosis

Tsukasa Hirashima, Kazuya Yamasaki, Hiroyuki Fukuda, and Hideo Funaoi

Learning Engineering Group, Information Engineering, Graduate School of Engineering,
Hiroshima University,
1-4-1, Kagamiyama, Higashi-Hiroshima, Hiroshima, Japan
tsukasa@isl.hiroshima-u.ac.jp

Abstract. In this paper, we describe a framework of Kit-Build Concept Map (we call it as KB map) that can diagnose. The task to make a concept map is divided into two sub-tasks: 1) "segmentation task" where parts of the concept map are extracted and 2) "structuring task" where the extracted parts (kit) are connected into a map. In the framework of KB map, an ideal concept map (goal map) is prepared by a teacher or an expert at first, and parts are generated by decomposing the goal map. The parts are provided to learners, and then the learners build concept maps (learner maps) by connecting the parts. Since the same parts are used both in the goal map and learner maps, it is possible to diagnose the maps by comparing them. This paper mainly explains a practical flow of KB map building.

Keywords: Kit-Build, Concept Map, Automatic Diagnosis, Goal map, Learner Map, Group Map, Difference Map, Segmentation and Construction Tasks.

1 Introduction

Automatic diagnosis of concept map is one of the most important issues in using concept map in technology-enhanced learning [1, 2]. In this paper, "Kit-Build Concept Map" as an approach to realize automatic diagnosis of concept maps is proposed [3]. We have divided the task to build a concept map into two sub-tasks: 1) "segmentation task" where parts (called "kits") of a concept map are extracted and 2) "structuring task" where the extracted parts are connected. In the framework of KB map, an ideal concept map (goal map) is prepared by an expert or a teacher at first, and parts are generated by decomposing the goal map. The parts, then, are provided to the learner, and then the learner builds a concept map (learner map) by connecting the parts. Therefore, in the framework of KB map, the segmentation task is carried out by teacher or domain expert, and learner carries out recognition task instead of the segmentation task. Then, the construction task remains as it is. The same approach where segmentation task is replaced to recognition task and construction task is kept, has been often adopted in the context of “note-taking” [4].

In the KB map, because the learner builds a learner map with the same parts with the goal map, it is possible to realize automatic diagnose learner maps by comparing with the goal map. This diagnosis makes the following matters possible for a teacher

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 466–468, 2011.

and learners: (i) getting the differences between a goal map and a learner map, (ii) getting the differences between each of learner maps, and (iii) getting a group concept map which is generated by overlaying several learner maps including the group. The group map can also be compared with the goal map or the learner map. The results of the diagnosis enable the environment to indicate inadequate portions in the concept map of individuals or group. Additionally, since it is possible to evaluate the similarity of the concept maps of each learner's, the results are useful to formulate collaborative learning group. In terms of learning effect, replacement of the segmentation task with the recognition task is expected to be useful as scaffolding by deleting the load of segmentation task and by focusing learner's task on connection between concepts. In usual situation of concept map building, however, segmentation task and structuring task are executed with mutual interaction. Since the segmentation task is replaced to the recognition task, the adequately applicable range of the KB map should be more restricted than the general concept map. In the remaining part of this paper, a practical flow of KB map building is explained.

2 Practical Flow of KB Map Building

In order to realize KB Map adequately, we have designed a practical flow to build the KB map shown in Figure 1. In the flow, the teacher and learners are able to interact with each other through concept maps. There are four main phases: 1) goal map building and kits generation, 2) learner map building, 3) map diagnosis and goal map modification, and 4) Learner map modification.

In the learner map building phase, kit generated by decomposing a goal map is provided a learner, like shown in Figure 2. Then, the learner is required to make a learner map by using the kit based on his/her understanding. Figure 3 shows an example of a learner map that is not completed. Because the map is composed by

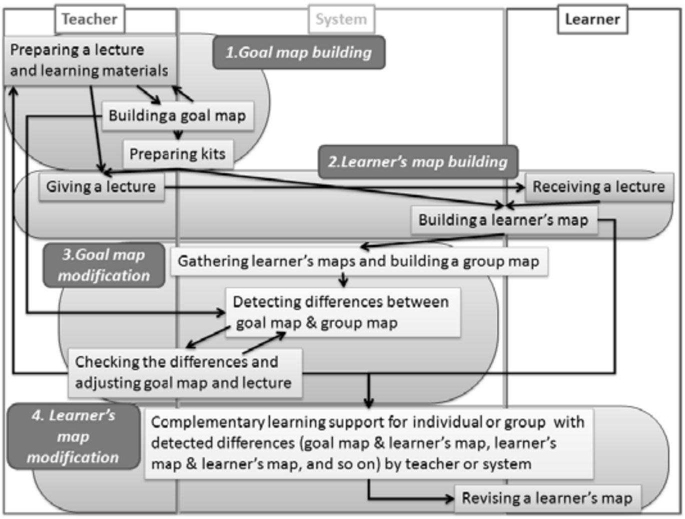

Teacher
System
Learner
Preparing a lecture and learning materials
1.Goal map building
Buildinga goal map
Preparing kits
2.Learner's map building
Giving a lecture
Receiving a lecture
Buildinga learner's map
3.Goal map modification
Gathering learner's maps and buildinga group map
Detecting differences between goal map & group map
Checking the differences and adjusting goal map and lecture
4. Learner's map modification
Complementary learning support for individual or group with detected differences (goal map & learner's map, learner's map & learner's map, and so on) by teacher or system
Revising a learner's map

Fig. 1. Practical Flow of Kit-Building Concept Map Building

connecting links between nodes, all errors are detected as mistakes in link connection. For example, in Figure 3, "Sublimation" is not used in the learner map. Since "Deposition" link from "Solid" to "Gas" in the learner map does not exist in the goal map, this error is also detected.

Just after the learner's map building phase, the learner maps are gathered on-line and a group map is generated by overlaying them. Figure 4 shows an example of the group map. A link included in more learner maps is drawn by bold link, and a link included in less learner maps is drawn by thin link. For example, since "Melting" link from "Solid" to "Liquid" is included in almost all learner maps, it is drawn as a bold link. Since "Deposition" link from "Solid" to "Gas" is included only in few learner maps, it is drawn as a thin link. Somewhat bold "Condensing" link disconnected to the group map means that the link is not used in several learner maps.

By comparing the group map with the goal map, the difference between ideal understanding and learners' current understanding is extracted. The difference is represented as "difference map". Based on the difference map, it is possible to generate feedback for learners in order to support them to correct their maps or for teachers in order to promote them to improve their goal maps.

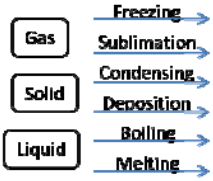


Fig. 2. Kit of Map

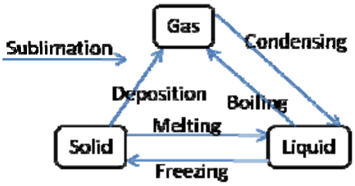


Fig. 3. A Learner Map

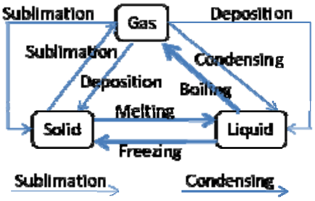


Fig. 4. A Group Map

3 Conclusion

In this paper, the Kit-Build method is proposed as a promising approach to realize automatic assessment of a concept map. Thorough several preliminary experimental uses of the system, we have judged that the Kit-Build method is a promising approach to realize automatic assessment of concept map. We are planning a large-size, long-term and more practical use of the environment as our important next step of this research.

References

1. Kornilakis, H., Grigoriadou, M., Papanikolaou, K.A., Gouli, E.: Using WordNet to Support Interactive Concept Map Construction. In: Proc. of ICALT 2004, pp. 600–604 (2004)
2. Gouli, E., Gogoulou, A.: Evaluating learner's knowledge level on concept mapping tasks. In: Proc. of ICALT 2005, pp. 424–428 (2005)
3. Yamasaki, K., Fukuda, H., Hirashima, H., Funaoi, H.: Kit-Build Concept Map and Its Preliminary Evaluation. In: Proc. of ICCE 2010, pp. 290–294 (2010)
4. Armbruster, B.B.: Taking Notes from Lectures. In: Flippo, R.F., Caverly, D.C. (eds.) Handbook of College Reading and Study Strategy Research, pp. 175–199. Lawrence Erlbaum Associates, NJ (2000)

The Effects of Domain and Collaboration Feedback on Learning in a Collaborative Intelligent Tutoring System

Jay Holland[1], Nilufar Baghaei[2], Moffat Mathews[1], and Antonija Mitrovic[1]

[1] Intelligent Computer Tutoring Group, University of Canterbury, Private Bag 4800,
Christchurch 8140, New Zealand
[2] Dept. of Computing, Unitec Institute of Technology, Auckland, New Zealand
tanja.mitrovic@canterbury.ac.nz

Abstract. We present initial results from a study comparing the effects of domain and collaboration feedback on learning within COLLECT-*UML*, a collaborative problem-solving ITS. Using COLLECT-*UML*, two students in separate physical locations (a collaborative pair) construct UML class diagrams to solve problems together. In the default version, COLLECT-*UML* provides both domain and collaboration feedback. In this study however, collaborative pairs were randomly assigned to one of four modes (treatment conditions) which varied the feedback presented by the system: no feedback (NF), domain feedback only (DF), collaborative feedback only (CF), and both domain and collaborative feedback (DCF). All conditions improved significantly between pre- and post-test, showing that practicing within COLLECT-*UML* helps learning. At a surface level, collaborative pairs in all modes had similar amounts of collaboration. The DCF mode had significantly higher learning gains than the other modes, indicating the value of receiving both domain and collaborative feedback. Surprisingly, the CF mode had the lowest learning gains (lower than NF), suggesting that, in this case, good collaboration without domain feedback could have simply reinforced erroneous domain knowledge.

Keywords: Collaboration, domain feedback, collaborative feedback.

1 Introduction

Researchers in Computer Supported Collaborative Learning have shown the benefits of adaptive collaboration support in Intelligent Tutoring Systems (ITSs) [1-2]. We previously extended COLLECT-*UML* with a collaboration model which provided students with automatic feedback on their collaboration in addition to on-demand domain feedback [3-4]. Here, we present the initial results of a study in which we attempt to separate the effects of domain and collaborative feedback and find their effect on learning.

COLLECT-*UML* is a constraint-based collaborative ITS which provides students with opportunities to practice their Unified Modeling Language (UML) skills by collaborating with a partner [3]. The system automatically creates collaborative pairs by connecting two students who have logged in and are still unpaired. The web

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 469–471, 2011.

interface provides each student with two solution spaces (individual and group). The intention is that each student first thinks about the problem individually (while creating their individual diagram) before contributing to the shared group diagram. Each student is encouraged to communicate (e.g. discuss their knowledge, provide explanations, seek justifications) with their partner via a chat interface. COLLECT-*UML* stores both domain and collaboration student models and, in the default version, provides students with two types of feedback: domain and collaboration feedback.

2 Evaluation

COLLECT-*UML* has been used in a second-year Software Engineering course (COSC224) at the University of Canterbury for the last few years. We used the lab sessions during the week of 20 September 2010 (week six of the course) to conduct the evaluation study. The intention was to have a setting that was as close to the normal learning environment experienced by students. Seventy-two COSC224 students participated in this study for no reward. None of these students had prior experience with COLLECT-*UML*. Written pre- and post-tests were administered during which students were given ten minutes to answer questions relating to UML diagrams. Both tests were comparable in difficulty. Following the pre-test, students were asked to read a one-page document which contained basic instructions for the study and guidelines for good collaboration [4].

Each collaborative pair was randomly placed into one of four treatment conditions (modes). Each mode altered the type of feedback students received: 1) no feedback (NF), 2) domain feedback only (DF), 3) collaboration feedback only (CF), and 4) domain and collaboration feedback (DCF). Students who received domain feedback (DF and DCF) could submit their solutions at any time to get feedback. Students who did not receive any domain feedback (NF and CF) were instructed to work on their problems till the pair jointly agreed that the solution was correct before moving on to another problem. All modes could request to view the full solution. However, as the full solution is a form of domain feedback, all students were advised that viewing the full solution would lock their problem (i.e. they would not be able to continue working on the problem after viewing the full solution). The system logged all actions performed, including their chats. The system regularly updated all student collaboration models; however, only modes CF and DCF received feedback on their collaboration. All other aspects of the system were identical between modes.

Sixty-one students completed both tests (Table 1). There were no significant differences on the pre-test. However, all modes improved significantly between pre- and post-test (all with $p<0.01$). DCF had significantly higher gain than the other modes ($F=4.46$, $p<0.01$), even when the normalized gain is used ($F=3.48$, $p=0.02$); conversely, CF had the lowest gain.

The number of times a student held the pen (to modify the group solution), the chat file size, and the number of changes made to the solution are shown in Table 1. These give us an idea of the amount of collaboration at a surface level. There were no significant differences between the modes indicating that the amount of collaboration was relatively similar between groups. However, further analyses have to be conducted to examine the quality of these collaborative actions.

Table 1. Statistics for all treatment groups

Mode (# students)	**NF (20)**	**DF (18)**	**CF (18)**	**DCF (16)**
Test completed	19	12	15	15
Pretest	2.4 (0.9)	2.3 (0.8)	2.4 (0.8)	1.7 (1.1)
Posttest	3.9 (0.9)	4.3 (0.8)	3.7 (1.1)	4.5 (0.9)
Gain	1.5 (1.2)	2.0 (1.9)	1.2 (1.4)	2.8 (1.1)
Pen held	13.4 (7.9)	11.0 (7.7)	13.7 (7.5)	15.6 (7.6)
Chat size	2604.7 (2359.0)	2726.4 (2470.9)	2818.3 (1473.6)	2494.8 (1935.2)
Changes	148.8 (69.2)	197.7 (105.2)	179.4 (110.9)	193.5 (75.5)

3 Conclusion

We presented a study comparing the effects of domain and collaboration feedback on learning within COLLECT-*UML*. All four treatment conditions improved significantly between pre- and post-test, showing that practicing within COLLECT-*UML* helps learning. At a surface level there was no difference in collaboration between modes. The DCF mode learnt significantly more than other modes, indicating the value of receiving both domain and collaborative feedback. Surprisingly, the CF mode had the lowest learning gains (lower than NF). One possible interpretation of this could be without domain advice students simply shared and possibly even promoted their misconceptions. We plan to perform deeper analyses of collaboration quality and problem-solving progress.

References

[1] Tchounikine, P., Rummel, N., McLaren, B.: Computer Supported Collaborative Learning and Intelligent Tutoring Systems. In: Nkambou, R., et al. (eds.) Advances in Intelligent Tutoring Systems, vol. 308, pp. 447–463. Springer, Heidelberg (2010)

[2] Hausmann, R.G.M., van de Sande, B., VanLehn, K.: Shall we explain? Augmenting learning from intelligent tutoring systems and peer collaboration. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S., et al. (eds.) ITS 2008. LNCS, vol. 5091, pp. 636–645. Springer, Heidelberg (2008)

[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., et al. (eds.) UM 2007. LNCS (LNAI), vol. 4511, pp. 217–227. Springer, Heidelberg (2007)

[4] Baghaei, N., Mitrovic, A., Irwin, W.: Supporting collaborative learning and problem-solving in a constraint-based CSCL environment for UML class diagrams. Computer-Supported Collaborative Learning 2, 159–190 (2007)

Multimodal Affect Detection from Physiological and Facial Features during ITS Interaction

M.S. Hussain[1,2] and Rafael A. Calvo[2]

[1] National ICT Australia (NICTA), Australian Technology Park, Eveleigh 1430, Australia
[2] School of Electrical and Information Engineering, University of Sydney, Australia
Sazzad.Hussain@nicta.com.au, Rafael.Calvo@sydney.edu.au

Abstract. Multimodal approaches are increasingly used for affect detection. This paper proposes a model for the fusion of physiological signal that measure learners' heart activity and their facial expressions to detect learners' affective states while students interact with an Intelligent Tutoring System (ITS). It studies machine learning and fusion techniques that classify the system's automated feedback from the individual channels and their feature level fusion. It also evaluates the classification performance of fusion models in multimodal systems, identifying the effects of fusion over the individual modalities.

Keywords: Affective computing, multimodality, AutoTutor, feedback, learning interaction, fusion.

1 Introduction

Our affective states (e.g. emotions) influence what we learn and how we do it, both face to face and online. We can detect them using modalities such as facial expressions, gesture, vocalization and a variety of physiological signal [1, 2]. Recent affective computing research, within AIED and elsewhere, has focused on integrating multiple modalities. Most of this work has involved features from audio-visual, speech-text, dialog-posture, face-body-speech, or physiological signals [c.f.p. 2, 3].

Despite progress there are many open questions on how to integrate signals from different physiological components and other modalities. Feature level and decision level fusion are commonly used in affective computing [2], yet how this should be done is still a challenge. In some cases the fusion models may be less accurate than the best individual channel. In other cases when the fusion model is more accurate, the effect could be linear or nonlinear. To quantify the nonlinearity in multisensory response, neuroscientists have introduced the concept of superadditivity where, the combination of more than one sensor is higher than either one alone [4]. Using the theories of multisensory integration, multimodal classification performance is superadditive (nonlinear) when it is greater than the sum of individual channels. Alternatively, redundancy among the channels may also be responsible where the channel integration may not yield higher performance [1]. Inhibitory effects will occur when the fusion model exhibits significantly lower accuracy scores than the individual channels.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 472–474, 2011.

The feedback provided to a learner, for example by an Intelligent Tutoring System (ITS) will have an impact on his affective state. Aghaei Pour et al. [5] investigated the impact of ITS feedback on learners' affective states and physiological states. This paper follows on that work evaluating the effects of physiological and facial feature fusion using classification approaches for the same system feedback dataset. Affective states are significantly dependent on AutoTutor feedback therefore; the feedback data is used as stimuli that trigger affective states which are reflected in students' physiology and facial expressions [5].

2 Computational Model and Results

In this study, an electrocardiogram (ECG) sensor measured heart activity and a video camera recorded facial expression from 16 learners' while they interacted for 45 min with AutoTutor. AutoTutor is an intelligent tutoring system (ITS) that provides customized instructions and feedbacks related to learning by interacting with them in natural language [6].

The Augsburg Matlab toolbox (AuBT)[1] for physiological signal processing was used for extracting 87 statistical features from the ECG channel. The eMotion[2] software by Visual Recognition was used on the face videos for extracting vector of motion 12 features from certain regions of the face. Chi-square (X^2) feature selection algorithm was used for selecting equal features from each channel. All selected features were then merged to achieve the feature level fusion. For classification, the Waikato Environment for Knowledge Analysis (Weka)[3] was used. For this study three machine learning algorithms; k-nearest neighbor (KNN), linear support vector machine (SVM), and decision tree were selected for classification. Finally, the average probability of the three classification outcomes was achieved using a vote classifier and the training and testing was performed with a 10-fold cross validation. The overall classification performance was measured using a kappa value. The target classification results are the types of AutoTutor feedback (positive-negative). The individual channels performed better than the fusion model that had an inhibitory effect for eight learners (ECG for 5, face for 3). Superadditivity was observed for two learners and redundancy for one learner. The accuracy of detecting feedback for the remaining 5 learners were below random (kappa=<0) for both channels and their fusion model. As an example of the three types of effects, results are presented for the learners exhibiting superadditive (two learners), redundant (one learner) and inhibitory effects (two learners selected randomly out of eight). Overall performance (kappa) of the individual channels and the fusion model are shown in Figure 1 for five learners. Further analysis on the performance (accuracy) of detecting the individual feedback types can be done to evaluate how the fusion of multichannel features can increases precision for some feedback types but reduces for others.

[1] AuBT: http://mm-werkstatt.informatik.uni-augsburg.de/project_details.php?id=%2033

[2] eMotion: http://www.visual-recognition.nl/

[3] WEKA: http://www.cs.waikato.ac.nz/ml/weka/

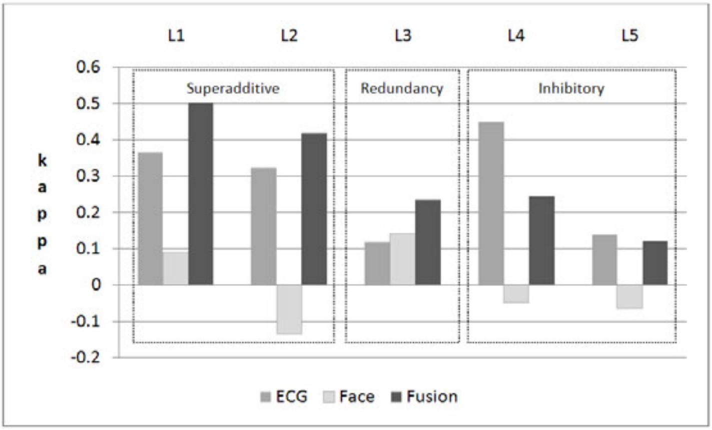


Fig. 1. Kappa scores for the individual channels and fusion model of five sample learners

3 Conclusion

Tutor feedback (human or automated) has an affective impact on learners. The aim of this study was to evaluate approaches for the fusion of physiological and facial features during learning interaction. The results for the fusion model were evaluated against the single channels to understand the effects of multimodality. Results show that the fusion of ECG and facial expression improved the mean accuracy (kappa) over the face channel but not the ECG channel. Results further show that the fusion model can perform very well with superadditive effects for some learners and redundancy/inhibitory effects with others.

Acknowledgments. M. S. Hussain was supported by Endeavour Awards and National ICT Australia (NICTA). We thank the Institute for Intelligent Systems, University of Memphis for AutoTutor. We also thank Edmundo Pablo Leiva-Lobos for extracting facial features using the eMotion software.

References

1. D'Mello, S., Graesser, A.: Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. User Modeling and User-Adapted Interaction 20, 147–187 (2010)
2. Sebe, N., Cohen, I., Gevers, T., Huang, T.S.: Multimodal approaches for emotion recognition: a survey. In: Proc. SPIE, vol. 5670, pp. 56–67 (2005)
3. Calvo, R.A., D'Mello, S.: Affect Detection: An Interdisciplinary Review of Models, Methods, and their Applications. IEEE Transactions on Affective Computing 1, 18–37 (2010)
4. Holmes, N.P., Spence, C.: Multisensory integration: space, time and superadditivity. Current Biology 15, 762–764 (2005)
5. Aghaei Pour, P., Hussain, M., AlZoubi, O., D'Mello, S., Calvo, R.: The Impact of System Feedback on Learners' Affective and Physiological States. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6094, pp. 264–273. Springer, Heidelberg (2010)
6. Graesser, A.C., Chipman, P., Haynes, B.C., Olney, A.: AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. IEEE Transactions on Education 48, 612–618 (2005)

Students' Enjoyment of a Game-Based Tutoring System

G. Tanner Jackson, Natalie L. Davis, and Danielle S. McNamara

University of Memphis, Memphis, Tennessee
{gtjacksn,nldavis,dsmcnamr}@memphis.edu

Abstract. iSTART-ME is a new game-based learning environment developed on top of an existing ITS (iSTART). The current study deviates from previous work focusing on individual ITS components, and utilizes a smaller number of students (n=9) who engaged with the entire system over a period of several weeks. The participants indicated that they enjoyed the game-based aspects of the system significantly more than the non-game aspects. These results support the use of iSTART-ME as a system that promotes long-term enjoyment.

Keywords: Serious Games, Intelligent Tutoring Systems, game-based tutoring.

1 iSTART-ME

The Interactive Strategy Training for Active Reading and Thinking (iSTART) tutor is a web-based system for young adolescent to college-aged students designed to improve reading strategies [1]. iSTART training consists of three main modules: Introduction, Demonstration, and Practice. The Introduction module contains three animated agents that engage in a vicarious dialogue to introduce the concept of self-explanation and the iSTART reading strategies. The Demonstration module includes two agents who generate and discuss the quality of example self-explanations. The Practice module requires learners to generate their own self-explanations and an animated agent provides qualitative feedback on how to improve the self-explanation quality. An Extended Practice environment continues this generative practice over a longer time period and allows teachers to assign specific texts. This long-term practice is necessary for skill mastery [2], but it can often lead to disengagement. Thus, iSTART-ME (motivationally enhanced) has been developed on top of an existing ITS and incorporates many game-based elements [3].

Within iSTART-ME there are three methods of generative practice (Coached Practice, Showdown, and Map Conquest) as well as three isomorphic identification minigames (Strategy Match, Bridge Builder, and Balloon Bust). Coached Practice is the updated version of the original iSTART practice, were learners generate their own self-explanations (SEs), are awarded points, receive feedback on SE quality, and an agent provides verbal feedback on how to improve the SE. In Showdown, a learner's generated SE is compared to a computerized opponent SE, and the player with the higher score wins the round (player with the most rounds wins the game). In Map Conquest the quality of a student's SE determines the number of dice that the student can use to conquer territories controlled by two virtual opponents. iSTART-ME also contains three isomorphic identification games that contain the same cognitive task

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 475–477, 2011.

within different combinations of game features. Strategy Match, consists of a drag and drop interface where the students can earn points and move up levels. Bridge Builder uses a similar interface with points and levels, but also includes a virtual scene where the users construct a bridge. Balloon Bust adds in a perceptual element to the virtual scene, where users must follow and click on the correct balloons.

1.1 Evaluation of iSTART-ME

All participants (n=9) completed the full iSTART-ME training (Introduction, Demonstration, and Practice), spent five one-hour sessions freely using the Selection Menu, and filled out a posttest survey. Analyses of the posttest survey compared iSTART-ME modules as well as the various mini-games. Within-subjects ANOVAs found significant differences between modules for the items “I had fun using this module,”, and “I would recommend this module to a friend,”, but did not find differences for the item, “This module was easy to use,” (see Table 1).

Table 1. Means (SD) for module ratings (1-6, higher numbers = stronger agreement)

	Intro	Demo	Practice	Menu	F(1,8)
I had fun using this module	1.22_a (0.44)	3.00_b (1.58)	2.78_b (1.20)	4.33_c (1.73)	28.89
This module was easy to use	5.00_a (1.32)	4.78_a (1.39)	4.89_a (1.05)	4.78_a (1.39)	0.231
I would recommend this module to a friend	1.44_a (0.53)	2.78_b (1.64)	2.89_b (1.27)	4.11_c (1.90)	20.17

*Subscripts indicate significantly different subgroups within a row, $p < .05$.

Table 2. Means (SD) for mini-game ratings (1-6, higher numbers = stronger agreement)

	Generation Games			Identification Games		
	Prac	Show	Map	Match	Bridge	Balloon
I liked the graphics in this game	3.44_a (1.24)	3.44_a (1.13)	3.89_a (1.45)	3.50_x (1.20)	3.50_x (1.69)	4.12_x (1.73)
I liked the sound effects in this game	2.33_a (1.23)	4.33_b (1.22)	4.22_b (1.56)	3.13_x (1.13)	3.62_x (1.60)	3.75_x (1.50)
I liked the music in this game	2.78_a (1.64)	4.22_b (1.09)	4.00_b (1.73)	3.50_x (1.51)	3.75_x (1.75)	3.75_x (1.49)
This game was fun to play	2.56_a (1.13)	3.33_a (1.41)	3.44_a (1.88)	3.38_x (0.92)	3.50_x (0.93)	4.62_x (1.30)
I would play this game again	2.56_a (1.42)	3.22_a (1.86)	3.22_a (1.79)	2.50_x (1.20)	3.62_y (0.74)	4.62_z (1.30)
This game was frustrating	2.44_a (1.33)	2.33_a (1.22)	4.00_b (2.06)	3.13_x (2.03)	2.50_x (1.31)	2.62_x (1.06)
I enjoyed playing this game	2.67_a (1.32)	3.44_{ab} (1.74)	3.67_b (1.73)	3.00_x (1.41)	3.62_x (1.06)	4.38_y (1.19)

*Subscripts indicate significantly different subgroups within a row, $p < .05$.

Two separate comparisons were made to investigate the group of generation games and the set of isomorphic identification games. Within-subjects ANOVAs on the three generation games yielded significant differences for several posttest survey questions (see Table 2). Students rated Map Conquest as the most frustrating, F(1,8)=7.84, *p*=.02, but also the most enjoyable generation game, F(1,8)=7.20, *p*=.03. Within-subjects ANOVAs comparing the identification games found that Balloon Bust was significantly more enjoyable than the other games, F(1,8)=6.67, *p*=.04, and was the most likely to be played again, F(1,8)=12.11, *p*=.01.

2 Conclusions

The current results support the design of iSTART-ME and indicate that students enjoyed interactions with the new game-based aspects of the system over an extended period of time. Specifically the students provided higher ratings for those modules and mini-games that contained more game-like aspects.

One limitation of the current study is the small sample size, and how that limits generalization to a broader population of users. However, despite this limitation, the data indicate interesting trends that are supported by previous research [4],[5], and suggest that iSTART-ME can successfully sustain enjoyment over an extended period. This finding provides a foundation for future work focusing on the timelines of effects for specific game elements (e.g., competition, challenge, variety, etc.).

References

1. McNamara, D.S., Levinstein, I.B., Boonthum, C.: iSTART: Interactive Strategy Trainer for Active Reading and Thinking. Behavioral Research Methods, Instruments and Computers 36, 222–233 (2004)
2. Jackson, G.T., Boonthum, C., McNamara, D.S.: The Efficacy of iSTART Extended Practice: Low Ability Students Catch Up. In: Kay, J., Aleven, V. (eds.) Proceedings of the 10th International Conference on Intelligent Tutoring Systems, pp. 349–351. Springer, Heidelberg (2010)
3. Jackson, G.T., Dempsey, K.B., McNamara, D.S.: The evolution of an Automated Reading Strategy Tutor: From Classroom to a Game-enhanced Automated System. In: Khine, M.S., Saleh, I.M. (eds.) New Science of Learning: Cognition, Computers and Collaboration in Education, vol. (2010), pp. 283–306. Springer, New York (2010)
4. Cordova, D.I., Lepper, M.R.: Intrinsic Motivation and the Process of Learning Beneficial Effects of Contextualization, Personalization and Choice. Journal of Educational Psychology 88, 715–730 (1996)
5. Papastergiou, M.: Digital Game-based Learning in High School Computer Science Education: Impact on Educational Effectiveness and Student Motivation. Computers and Education 52, 1–12 (2009)

Optional Finer Granularity in an Open Learner Model

Matthew D. Johnson and Susan Bull

Electronic, Electrical and Computer Engineering, University of Birmingham, UK
{mdj384,s.bull}@bham.ac.uk

Abstract. Open learner models (OLMs) available independently from specific tutoring or guidance, such as an intelligent tutoring system may provide, can encourage learners to take greater responsibility for learning., Our results suggest that finer grained OLM information, in this context, can support learners in identifying strengths/weaknesses, planning and focussing learning, when different OLM granularities exist. Learners drew regular comparison between OLM and domain information, showing the flexibility of interaction to be important.

Keywords: Open Learner Model. Domain Information.

1 Introduction

A learner model (LM) is a representation of student knowledge in an educational environment, such as an intelligent tutoring system (ITS). It allows personalisation and adaptation towards the student and their current needs, and may be opened to the learner for consultation during personalised interaction. Open learner models (OLMs) may encourage learner awareness, responsibility and independence in learning, and may promote such metacognitive activities as reflection, planning and self assessment [1]. An OLM *independent* from specific tutoring or guidance, such as an ITS can provide, may further increase learner control and responsibility. These independent OLMs require learners to think in greater depth about activity choices and planning, as the responsibility for all learning based decisions rests with the learner [1].

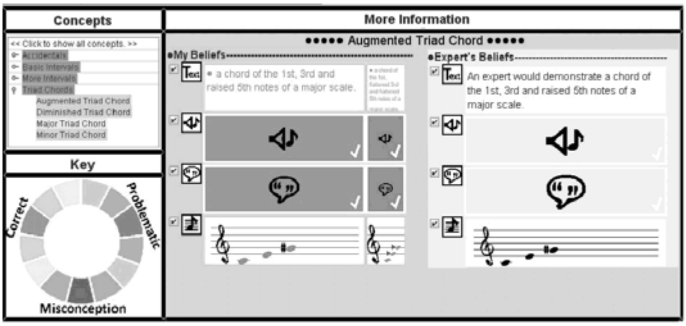
OLMs may display simple *knowledge level* information (e.g. coloured nodes [2], skill meters [3]) or learner *beliefs/ misconceptions* of finer granularity (e.g. text, structural relationships, animation [1]). The more fine grained information can provide evidence for *why* something is (mis)understood, elaborating beyond knowledge level information which states *if* something is understood. Information of finer granularity may commonly be accessed through a (simpler) knowledge level representation [2].

We consider whether optionally available finer granularity in an independent OLM can support learners in terms of knowledge, focus and planning, whether learners choose to inspect equivalent domain information and whether comparisons are drawn.

2 MusicaLM

MusicaLM is an IOLM in the domain of basic harmony, extended from [4]. Learners configure concepts on which to receive randomly selected questions. Combinations of notes are entered on a virtual keyboard/music stave to demonstrate understanding. Semitone intervals between entered notes are modelled by comparing patterns to the

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 478–480, 2011.



• a chord of the 1st, 3rd and raised 5th notes of a major scale.

• a chord of the 1st, flattened 3rd and flattened 5th notes of a major scale.

An expert would demonstrate a chord of the 1st, 3rd and raised 5th notes of a major scale.

Fig. 1. Open Learner Model and Equivalent Domain Information

domain model (e.g. [*n*=note; #=no. of semitones] pattern *n4n3n* = *major triad chord*.) Information is weighted in favour of the most recent, and understanding that is consistently incorrect in the same way is categorised a misconception. OLM/domain information may be inspected at any point, in different levels of detail. Background colour is used throughout to identify information as *correct*, *problematic* or a *misconception*. *Knowledge level* information (coarse granularity) is shown in the tree structure ("Concepts", top left, Fig. 1). Selecting a leaf from the tree presents *inferred beliefs* (patterns) in the "More Information" window (finer granularity). Alternative presentations cater for learners' individual preferences. The LM is shown centre (Fig. 1) and equivalent domain information on the right. Each comprises (top down): a *textual description*, *audio* (played on the piano), *spoken word* and *music notation*.

3 Evaluation

Participants were 15 adult volunteers (aged 18-30), learning music theory through personal interest and not receiving music tuition. All were familiar with music notation/keyboard. Each: (i) received a demonstration of MusicaLM; (ii) after familiarisation, attempted questions in MusicaLM for 30 minutes and were reminded they could inspect OLM/domain information at any point; and (iii) completed a questionnaire.

Results indicate learners made regular use of the finer granularity and found it useful (Table 1, a&b). Inspection of finer granularity (beliefs) was more frequent than

Table 1. Questionnaire responses: inspection of more finely grained beliefs

Question	Agree←→ Disagree		
a) It was useful to view my beliefs *(OLM information)*	13	1	1
b) It was useful to view expert beliefs *(domain information)*	13	1	1
c) I compared expert understanding with my own	12	2	1
d) It was useful to compare my understanding with that of an expert	13	1	1

Question	My Beliefs			Expert Beliefs		
Inspecting beliefs was useful …	Agree←→Disagree			Agree←→Disagree		
e) when identifying what I understood	12	2	1	9	5	1
f) when identifying problems	14	1	-	12	3	-
g) when identifying misconceptions	10	3	2	8	5	2
h) to help me focus my learning	11	2	1	12	2	1
i) to decide what to do next	8	5	2	9	5	1

Table 2. a) Depth of Inspection, b) Belief Inspection, c) Belief Inspection Episode

	Tree: Node Only	Tree: Leaf Only	Beliefs	Learner Beliefs	Domain Beliefs	Domain Only	Both	Learner Only
Total	56	105	169	323	279	35	73	17
Mean	3.7	7	11.3	21.5	18.6	2.3	4.9	1.1
Median	2	3	5	23	12	1	4	0
Range	2-18	3-20	5-38	0-58	0-79	0-13	0-18	0-17

coarse granularity in isolation (tree information) (Table 2a). Overall, slightly more learner beliefs were inspected than domain information (Table 2b). Beliefs were inspected on 125 occasions (Table 2c): 35 occasions were domain alone; 17 learner beliefs in isolation; on the majority of occasions (73) both were used together. Learners indicated they regularly drew comparison between their own beliefs and the domain, and found it useful (Table 1, c&d). Learners often inspected their own beliefs to help identify problems and misconceptions (Table 1: f, g) in addition to confirming things already understood (e). Expert beliefs were used less frequently for these purposes, but were indicated as most use when identifying problems (f). Inspecting beliefs was considered helpful to focus learning (h) and just over half of learners agreed it helped them choose what to do next (i).

4 Discussion and Summary

MusicaLM is *independent* from specific tutoring and guidance. It is thus encouraging that learners found information useful for planning, confirming understanding and providing focus in learning; behaviour consistent with non-independent OLMs [1], perhaps influencing learning-based decisions. More finely grained OLM information was used particularly with problems; learners inspected detail with purpose. Providing domain information saw regular comparisons being drawn; perhaps additional granularity can support learners in sense making, and encourage independence. Sometimes knowledge level information was sufficient, potentially satisfying learners' short-term informational/planning goals. Further work should establish goals and their effect on students' learning experience. Results show flexible access to OLM information important when varying granularity exists. This is a promising area for future research.

References

1. Bull, S., Kay, J.: Open Learner Models. In: Advances in Intelligent Tutoring Systems, pp. 318–338. Springer, Heidelberg (2010)
2. Zapata-Rivera, D., Hansen, E., Shute, V.J., Underwood, J.S., Bauer, M.: Evidence-based Approach to Interacting with Open Student Models. International Journal of Artificial Intelligence in Education 17(3), 273–303 (2007)
3. Mitrovic, A., Martin, B.: Evaluating the Effect of Open Student Models on Self Assessment. International Journal of Artificial Intelligence in Education 17, 121–144 (2007)
4. Johnson, M., Bull, S.: Belief Exploration in a Multiple-Media Open Learner Model for Basic Harmony. In: Artificial Intelligence in Education 2009, pp. 299–306. IOS Press, Amsterdam (2009)

Contextualized Reflective Support in Designing Instruction Based on Both Theory and Practice

Toshinobu Kasai[1], Kazuo Nagano[2], and Riichiro Mizoguchi[3]

[1] Graduate School of Education Master's Program, Okayama University, Japan
[2] Faculty of Liberal Arts, University of the Sacred Heart, Japan
[3] The Institute of Scientific and Industrial Research, Osaka University, Japan
kasai@cc.okayama-u.ac.jp, nagano@kayoo.org,
miz@ei.sanken.osaka-u.ac.jp

Abstract. In this study, we developed a system called FIMA (Flexible Instructional Design Support Multi-Agent System) that dynamically supports teachers in designing instruction by facilitating their thinking ways according to the characteristics of those of expert teachers. In the present study, we focused on a support to facilitate teachers' contextualized thinking included in the processes of expert teachers based on instructional/learning theories. In order to provide such support, we make use of the OMNIBUS ontology, which describes knowledge extracted from instructional/learning theories and best practices.

Keywords: Ontology, Instructional Design, Multi-Agent, Instructional/Learning Theories.

1 Introduction

The educational gaps caused by differences in teachers' professional abilities are a perennial problem, especially for complex tasks like instructional design. Among the several approaches to resolving this problem, providing teachers with an efficient and usable support system is promising, since most teachers want to participate in the process of designing high-quality instruction. In order to investigate strategies to support less-skilled teachers in designing instruction, it is effective to analyze skilled teacher's thinking processes in approaching this task. Sato et al. have investigated differences in thinking processes between expert and novice teachers when they analyze existing instructional plans [1]. This investigation led us to the conclusion that the thinking of expert teachers is characterized by the following three features: 1) multiple viewpoints thinking, 2) contextualized thinking, and 3) problem framing and reframing strategies. Because it is also important for teachers to analyze instruction objectively when they themselves design the instruction, this study aims to support teachers in designing high-quality instruction by directly facilitating these three types of thinking. In order to provide such support, we have proposed a Flexible Instructional design support Multi-Agent system, called FIMA [2]. In this paper, we focus on one of FIMA's supports to facilitate teachers' contextualized thinking based on instructional/learning theories and best practices.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 481–483, 2011.

Teachers explicitly create plans of their lessons that show rough flows of instructional and learning scenes designed to attain educational goals before they deliver them, in a format called a "lesson plan." We think that this format is one of the reasons why novice teachers cannot consider their lessons sufficiently from a contextual viewpoint. In this format, it is possible for teachers to describe every scene even if they do not consider relationships between scenes and their intentions in every scene in their lessons. Others' opinions and points of view are important to making teachers consider their lessons more deeply and effectively by teacher's contextualized thinking. Therefore, providing a computer support that can automatically provide teachers with others' reliable opinions and interpretations of their designed lesson plans can be highly valuable.

Based on these considerations, the intent of the present study was to support teachers in designing instruction based on instructional/learning theories and empirical knowledge extracted from best practices that can be regarded as others' opinions and interpretations. We made use of the OMNIBUS ontology, which describes knowledge that is extracted from instructional/learning theories and practices from the perspective of learners' state changes in a common form using shared concepts [7].

2 The I_L Event Decomposition Tree and Reflective Support

The OMNIBUS ontology is built to organize a variety of instructional/learning theories and empirical knowledge extracted from best practices independently of the learning paradigms [3]. Fig. 1 shows the basic construction of the OMNIBUS ontology. The core concepts of the OMNIBUS ontology are an *I_L event* and its decomposition structure. An I_L event is a basic unit of learning and instruction and is composed of the state change of a learner and instructional action and learning action. Such an I_L event shows what state a learner reaches. A method for how to achieve the state change (macro-I_L event) is expressed by a decomposition relation with micro-I_L events, called a *WAY*. By this decomposition, various methods that can achieve an educational goal can be described as WAYs. A macro-I_L event is decomposed into a couple of micro-I_L events by applying a WAY. A decomposition tree is developed by applying such decomposition recursively to other micro-I_L events as shown in Fig. 1.

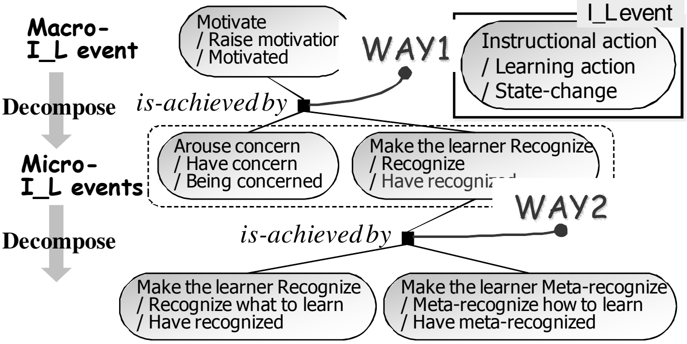

Macro-I_L event
Motivate / Raise motivatior / Motivated
WAY1
I_L event
Instructional action / Learning action / State-change
Decompose
is-achieved by
Micro-I_L events
Arouse concern / Have concern / Being concerned
Make the learner Recognize / Recognize / Have recogni...
WAY2
Decompose
is-achieved by
Make the learner Recognize / Recognize what to learn / Have recognized
Make the learner Meta-recognize / Meta-recognize how to learn / Have meta-recognized

Fig. 1. The Modeling Framework in the OMNIBUS ontology

With this modeling framework, the flow of a lesson is modeled as a tree structure of I_L events that is called an *I_L event decomposition tree*. In addition, teachers' strategies to achieve the goal of a lesson are expressed as a hierarchical structure by decomposing a macro-I_L event into smaller grained I_L events using WAYs. In this study, FIMA interprets lesson plans designed by teachers based on 110 WAYs, and automatically makes related I_L event decomposition trees in a bottom-up manner. Then, FIMA facilitates teachers' deep contextualized thinking by letting them compare decomposition trees with their lesson plans. FIMA expects that they will deeply reflect on their intentions regarding instructional design from a contextual viewpoint through confirming content expressed in the two kinds of nodes.

In this study, we have evaluated the function of FIMA through the practical uses by three teachers. Although we cannot describe in detail the results of the evaluation due to space limitation, we found out that teachers found 2.5 improvement points in each lesson plan on average by their contextualized thinking using FIMA.

3 Related Work and Conclusions

Here, we would like to introduce a related work, SMARTIES [3] to contrast FIMA with it. SMARTIES is an authoring system that aims to support designing learning/instructional scenarios based on the OMNIBUS ontology. By using SMARTIES, teachers can make I_L event decomposition trees compliant with learning/instructional theories through deeply reflecting their design intention of their lessons. For such instructional design, it is necessary for teachers to think from contextual and multiple viewpoints. So, though this approach is effective for expert teachers who can think from these viewpoints, it is very difficult for novice teachers.

On the other hand, our approach employs a bottom-up way and can automatically make I_L event decomposition trees through interpreting lesson plans that teachers usually design. By providing teachers with I_L event decomposition trees, this study expects that they will be conscious of their deep-level intention that they were not explicitly conscious of. In our approach, even novice teachers can participate in easily. This is one of the characteristics of our approach. To the best of our knowledge, there is no system which can automatically interpret teachers' deep-level intentions from their designed lesson plans, and can support them based on results of the interpretation.

References

1. Sato, M., Iwakawa, N., Akita, K.: Pratical Thinking Styles of Teachers: Comparing Experts' Monitoring Processes with Novices. Bulletin of the Faculty of Education 30, 177–198 (1991)
2. Kasai, T., Nagano, K., Mizoguchi, R.: An Ontological Approach to Support Teachers in Designing Instruction Using ICT. In: Proceedings of ICCE 2009, pp. 11–18 (2009)
3. Hayashi, Y., Bourdeau, J., Mizoguchi, R.: Using Ontological Engineering to Organize Learning/Instructional Theories and Build a Theory-Aware Authoring System. International Journal of Artificial Intelligence in Education 19(2), 211–252 (2009)

Problem-Solution Process by Means of a Hierarchical Metacognitive Model

Michiko Kayashima[1], Alejandro Peña-Ayala[2,3,4], and Riichiro Mizoguchi[4]

[1] College of Humanities, Tamagawa University, Japan
[2] WOLNM, [3] ESIME-Z & [3] CIC [3] National Polytechnic Institute
[4] Institute of Scientific and Industrial Research, Osaka University
kayasima@lit.tamagawa.ac.jp, apenaa@ipn.mx,
miz@ei.sanken.osaka-u.ac.jp

Abstract. We propose a Metacognitive Model devoted to problem-solving. It stimulates abstraction, modification, and instantiation metacognitive activities. Our model holds a hierarchical structure, a learning paradigm, and a workflow to skills acquisition. Such a model is a reference for problem-solving processes.

Keywords: Metacognitive model, abstraction, instantiation, class modification.

1 Introduction

Our metacognitive model enhances learner's cognitive skills. It aims individuals to become better learners and problem solvers. This paper is organized as follows: In section 2 we overview the underlying items of our model; whereas, a description of our Metacognitive Model is set in section 3. We summarize the contributions of our model and the future work to be achieved in the conclusions section.

2 Metacognitive Model's Baseline

Our model accounts: Flavell's Metacognitive Monitoring Model [1], the Meta-level/object-level Model set by Nelson and Narens [2], the workflow for skill acquisition designed by Anderson [3], and the Metacognitive Activity Model [4].

2.1 Metacognitive Phenomena

The Metacognitive Monitoring Model holds four classes of phenomena: *knowledge*, *experience*, *goals-tasks*, and *strategies* [1]. The knowledge holds a set of beliefs about person, task, and strategic factors that bias cognitive activities. The experiences represent subjective internal responses about preconditions for achieving a task and expectations of progress or completion of a task. Goals-tasks depict what the task is and the desired outcome to be fulfilled. Strategies are ordered processes devoted to control one's own cognition and to ensure the achievement of a goal.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 484–486, 2011.

2.2 Two Abstraction Levels Architecture

The Meta-level/object-level Model organizes cognitive processes into a meta-level and an object-level [2]. The former pursues to control internal cognitive processes and the later controls the mental activity achieved by individual in the external world. A monitoring flow is performed when the meta-level is informed by the object-level about the cognitive activity. A control flow is triggered when information goes from the meta-level to the object-level for changing the behavior at the object-level.

2.3 Skills Acquisition Workflow

The workflow for skill acquisition tailored by Anderson embraces three stages: *cognitive*, *associative*, and *autonomous* [3]. The Cognitive stage enables learner to get knowledge by objectivism practice. The outcome is *declarative knowledge* of the skill. The associative stage privileges the constructivism practice by problem-solving exercises. As a result, it adds *procedural knowledge*. The autonomous stage aims the learner to develop more domain problems, whose cases are diverse and represent increasing degree of complexity. This stage produces *refined knowledge* of the skill.

3 A Profile of the Metacognitive Model

Our Metacognitive Model is organized as a multi-tiers architecture [4]. The structure allocates cognitive activities according to their target of control and interaction. At the top, a metacognitive learning paradigm is set to represent the manipulation of classes. At the middle tier, a cognitive model for problem-solving is outlined. It encompasses a sequence of cognitive activities to represent the process of problem-solving. At the bottom level, a double-loop cognitive model is tailored. It accounts the skills acquisition workflow to acquire, evolve, and refine knowledge.

3.1 Metacognitive Learning Paradigm

The paradigm encompasses three cognitive operations to manipulate classes: 1) *abstraction operation*: monitors a problem-solving process at the “object-level” and yields a class to generalize its attributes at the “meta-level”; 2) *modification operation*: revises and updates class attributes at the appropriate grey-level. It holds three class operators: *addition*, *modification*, and *deletion*; 3) *instantiation operation*: occurs when a suitable class, an abstraction at “meta-level” of a problem-solving process, is successfully chosen to “control” cognitive activities at the “object-level”.

3.2 Cognitive Model for Problem-Solving

Our model achieves eight activities: 1) *observation*: creates cognitive products in working memory (WM); 2) *abstraction*: sets a class at meta-level; 3) *rehearsal*: maintains contents in WM; 4) *evaluation*: qualifies class attributes; 5) *modification*: tunes the class attributes; 6) *virtual execution*: applies operators to cognitive objects to test the class; 7) *selection*: chooses the class for being instantiated, 8) *instantiation*: deploys a representation of the class at the object-level to guide the problem-solving process [4].

3.3 Double-Loop Cognitive Model

The model follows three stages to acquire knowledge skill. In each stage, cognitive activity is performed as a double-cycle. A cycle contains three items: *input*, *process*, and *output*. At instance-level, input reveals the cognition of external objects, whilst at the meta-level it corresponds to monitoring. Process is the cognitive model for problem-solving at meta-level; whilst at object-level it reveals the cognitive activities to problem-solving. Output depicts the control flow from the meta-level to the object-level and the actions to be fulfilled at the instance-level.

4 Conclusions

Our model extends the Flavell's Metacognitive Monitoring Model by adding a structure of three tiers. The model also enhances the Meta-level/object-level Model by means of class operators and class activities. As a future work, we plan to develop a computer-based prototype to implement our Metacognitive Model.

Acknowledgments

The second author gives testimony of the strength given by his Father, Brother Jesus and Helper as one WOLNM project. The research holds a support from grants: CONACYT 118962, CONACYT 118862, CONACYT-SNI-36453, SIP-20110398, SIP-EDI: DOPI /3189/08, COFAA-SIBE, and a Grant-in-Aid for Scientific Research (C) 22500894 in Japan.

References

1. Flavell, J.H.: Metacognition and Cognitive Monitoring: A New Area of Cognitive-Developmental Inquiry. American Psychologist 34, 906–911 (1979)
2. Nelson, T.O., Narens, L.: Why Investigate Metacognition. In: Metcalfe, J., Shimamura, A.P. (eds.) Metacognition: Knowing about Knowing. MIT Press, Cambridge (1994)
3. Anderson, J.: Acquisition of Cognitive Skill. Psychological Review 89(4), 369–406 (1982)
4. Kayashima, M., Inaba, A.: The Model of Metacognitive Skill and How to Facilitate Development of the Skill. In: Proc. Int. C. Computers in Education, Hong Kong, pp. 277–285 (2003)

Modeling Mentoring Dialogue within a Teacher Social Networking Site

Jihie Kim, Yu-Han Chang, Sen Cai, and Saurabh Dhupar

Information Sciences Institute, University of Southern California, CA USA
{jihie,ychang,scai,dhupar}@isi.edu

Abstract. Online social networking tools promise to enable mentorship, professional development, and resource sharing between teachers across the Internet. This paper describes a first attempt to model teacher dialogue, an effort that will eventual lead to overlay tools that promote these kinds of beneficial community behaviors.

Keywords: Teacher social network, teacher mentoring.

1 Introduction

With the advent of the interactive web and social networking, the average school teacher now has unprecedented capability to reach out and connect with other educators from around the country, discover curriculum materials, share best practices, and create connections that enrich the education of our nation's youth (Brown, 2008). Several education-related social networks have arisen, such as Classroom 2.0 and MSP2, the Middle School Portal 2: Math and Science Pathways (MSP2, http://msteacher2.org). In our work, we initially focus on MSP2, which is a site where teachers use discussion forums and blogs for communicating their problems and providing help to others. In particular, we seek to identify mentoring strategies and characteristics of exchanges that seem particularly helpful for the teachers seeking help. By doing so, we can then build tools that specifically encourage these types of behaviors.

Our work builds on the existing research in mining and modeling online discussion forums (Kim and Shaw 2009, McLaren et al., 2007). Whereas prior attempts focused on student forums and subject comprehension, here we focus on 'teacher-to-teacher' dialogue and professional exchanges. We also expand the modalities considered, analyzing both forums and blogs. Dialogue annotations and quantitative content measures are used to analyze the site data for mentoring or help-providing activities. We plan to develop automatic classification approaches.

2 Annotation and Analysis of Teacher Online Discussions and Blogs

Table 1 shows a selection of the tags that we use in analyzing information exchange in forums and blogs. The full list includes categories derived from (Danielson et al.,

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 487–490, 2011.

2009; Ravi and Kim 2007; Klein, 2006; Skyes, 1983). To determine the kappa score, two annotators reviewed 10 forums threads containing 77 total messages and 20 blog postings. Agreement is generally reasonable; we expect this to improve as we refine the annotation manual. We are particularly interested in the Knowledge Sharing Strategies that teachers use to communicate with each other, often in the form of answering each other's questions about particular Teacher Tasks.

Table 1. A selection of tags used in annotating teacher discussions and blog posts

Tag Type	Tags	Description	Example cues	Kappa
Speech Act	que	Questions or requests for help.	"how do you","I wonder"	0.69
	ans	Provides answers or suggestions to a previous question or request for help.	"my suggestion is", "you probably want to"	0.59
Knowledge Sharing Strategies	link	Specifies a link to a resource, a video clip, or a general website	"<a href=…", "here is a website", "here is a link"	0.73
	personal_exp	Answers a question using personal experience	"I have been doing this for", " I was"	0.62
	other_exp	Answers a question by citing others' experiences	"my collegue who is history teacher has been doing this for years"	0.79
	book	Answers by recommending a related book as reference	" I have been reading <title> about …"	0.74
Teacher Tasks	it	Integrating Technology	twitter, moddle, podcast	0.68
	instr	Instructional Strategies	differentiated instruction,	0.65
	comm	Communication with students	"twitter with student"	0.64
	math_anx	Students' math anxiety	"never good at math", "hate math","fear math"	0.87

Our initial analysis focuses on determining the characteristics of messages and authors that lead to popular discussions, which for the moment we define as long discussions with many participants. Table 2 shows summary statistics across long vs. short discussions and blogs. Long discussions are defined as the top quartile of threads (in terms of number of responses) that have at least one response; the other categories are defined similarly. Long discussions, for example, have at least seven responses, and short ones have at most two responses.

Immediately there are a few interesting observations that can be made. First, the sharing of personal experiences appears to engender longer discussions. Second, IT and Instructional Strategies tend to be the most popular discussion topics, which is reasonable given the nature of MSP2. Third, and perhaps somewhat surprisingly, we note that lengthy initial forums posts appear to be a turn-off, leading to short discussions. The length of the first answer to the initial post appears to be a much better predictor of the eventual popularity of the thread.

Table 2. Comparison of mentoring activities in long versus short threads

Discussions and Blogs	Avg # resp.	Avg # words initial post	Avg # words in 1st ans.	Avg # participants	Knowledge sharing strategy of the first answer/comment	Related topics and tasks
Long Disc.	13.1	166.0	109.5	8.09	link(11),pers_exp (8), other_exp (1)	it(9),instr(5),comm (4),math_anx(1)
Short Disc.	1.5	423.3	65.79	1.36	link(16), personal_exp(1), other_exp(1)	it(7), instr(4), math_anx(1), comm(1)
Long Blogs	11.0	211.7	135.3	6.11	link(16), book(2), pers_exp(2),other_exp(1)	it(18), instr(9)
Short Blogs	1.2	137.7	68.1	1.24	link(4), other_exp(1)	instr(4)

Table 3. Characteristics of the help provided by some of the frequent participants

	#threads started	#ans.	Avg #words in ans.	# replies to answers	Avg # replies to answers	#messages annot.	Knowledge Sharing Strategy
P1	11	52	90.75	51	2.00	17	personal_exp(6) link(4)
P2	22	45	100.64	31	2.83	7	link(4), pers_exp(2), other_exp(1)
P3	6	18	142.16	23	0.73	7	pers_exp(4), link(2)
P4	4	19	148.63	23	3.71	9	book(3), link(2), pers_exp(1),
P5	6	24	207.88	17	2.36	7	personal_exp(1)

We also examined the top participants within the MSP2 site and analyzed their collaboration behavior. In particular, we are interested to see how they attempted to provide help to others, and how their contributions impacted the participation of others in the discussions. Table 3 highlights a few statistics that were collected from individual teachers. These statistics were calculated across 27 randomly sampled discussion threads, comprising a total of 153 messages.

Many of the frequent participants tend to draw on personal experience and point to web links as knowledge sharing strategies, and they tend to write average length responses to others' questions. Further investigation is needed to determine the variables that best predict which answers receive the most feedback. While not shown in the tables due to space constraints, the popularity of individual authors appeared consistent between the blogs and forums, i.e. people who get more responses in discussions also get more comments in blogs and vice versa. We plan to analyze potential partitions between help-seekers and help-providers.

3 Summary

Clearly this is only a first step towards modeling the dialogue and behavior of teacher online mentoring and collaboration. By first understanding the data, we will be able to automatic classifiers, and eventually, add-on tools that enable the current generation of professional networking tools to be much more effective.

Acknowledgement

This work is supported by NSF (DUE/NSDL award #1044427).

References

Brown, Seely, J., Adler, R.P.: Minds on Fire: Open Education, the Long tail, and Learning 2.0. EDUCAUSE Review 43(1) (2008)

Fredericks, A.D.: The Teacher's Handbook: Strategies for Success. Rowman & Litlefield Education (2010)

Kim, J., Shaw, E.: Pedagogical Discourse: Connecting Students to Past Discussions and Peer Mentors within Online Discussion Board. In: Innovative Applications of Artificial Intelligence Conference (2009)

Klein, M.B.: New Teaching and Teacher Issues. Nova Science Publishers, New York (2006)

Sentiment-Oriented Summarisation of Peer Reviews

Sunghwan Mac Kim and Rafael A. Calvo

School of Electrical and Information Engineering, University of Sydney
{sunghwan.kim,rafael.calvo}@sydney.edu.au

Abstract. It is common that students peer-review other students' writing, and these reviews are useful information to instructors, both on the particulars of the essay being reviewed, the feedback provided and the overall progress of the class. This paper describes a novel approach to summarising feedback in academic essay writing. We present a summarisation method for identifying and extracting representative opinion sentences from each feedback. Sentiment score-based techniques are employed and SentiWordNet is used as a linguistic lexical resource for sentiment summarisation. We evaluate our approach with the reviews written by a group of 50 engineering students.

Keywords: sentiment summarisation, peer review, student feedback.

1 Introduction

Online reviews are becoming increasingly significant on the Internet. Users create and rate significant amounts of content with useful information for academic and commercial purposes. Within the fields trying to automatically make sense of this content, sentiment (or opinion) summarisation has become a growing research topic.

Most existing research has been based on movie or product reviews. However, sentiment summarisation would be useful in other situations. For example, sentiment summaries can be used in students' feedback to interpret the rationale behind an evaluation. This valuable information can, for example, help a university lecturer obtain a more precise understanding of the[1] feedback for his or her lecture. Peer feedback of writing tasks is helpful to discover positive and negative features linked to the quality of the document [1].

In this study, we concentrate on a specific domain, which is peer review on student essay. Providing feedback to students is crucial to the learning process during the process of writing. Automatically computer generated summary on peer review can effectively enhance student's learning in academic writing. Furthermore, the highlighted feedback encourages the students to engage in the next writing and to make more meaningful changes on their work. To our best knowledge, there is not any research on summarising sentiments or opinions under the context of peer review. We suggest summarisation in terms of sentiment in order to provide more meaningful feedback to students.

[1] This project has been partially supported by a Google Research Award.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 491–493, 2011.

2 Sentiment-Oriented Summarisation (SOS)

We define three methods to extract a set of candidate 'sentiment' sentences. Each sentence is ranked based on the scores below, and the highest ranked sentences are used to produce the summary. In our experiment we selected the top six sentences often considered enough to summarize a document [2]. If there are less than six sentences in a document, the entire content of document is included as a summary.

Three sentiment scores are produced for each sentence. The first, Intensity Sentiment Score (ISS), attempts to extract opinion sentences that contain as much sentiment information as possible. The sentiment information of each word in a sentence is calculated with the linear combination of TF-IDF (Term Frequency-Inverse Document Frequency) weight, Part-Of-Speech (POS) and Polarity. POS and Polarity values come from SentiWordNet 3.0 [3]. We define the Representativeness Sentiment Score (RSS) function, which computes the representativeness of a sentence by finding keywords that capture the main content in a document. RSS is a measurement of how well a summary captures the principal information content of the original document. We adopt and improve Luhn's approach [4]. RSS is a linear combination of occurrence frequency and TF-IDF weight. The Coverage Sentiment Score (CSS) function calculates the relative significance degree of a sentence with respect to the remaining sentences in a document by means of SentiWord Correlation Factor (SCF). The SCF is computed by iteratively measuring the relevance of each word in a sentence to words in the whole document.

3 Evaluation and Discussion

This study used peer-reviews collected from 50 undergraduate students in ELEC3610 (E-Business Analysis and Design). First students were required to write a first draft of a Project Proposal, then they acted as peer-reviewers to evaluate each a draft as a part of course assessment. Both the draft and the peer-review were part of the assessment and were supported through iWrite [5].

We use the Jensen-Shannon (JS) divergence, a symmetric measure of the similarity of two pairs of distributions as evaluation metric. The JS divergence without smoothing is considered a good measure for comparing input text and summary content [6]. The smaller the value of divergence, the better the summary. We compared the SOS measure based on the three scores and two baseline measures: the *First-6* and *Last-6* summarisers, which are constructed by extracting the first and the last six sentences in the peer review, respectively [7]. The baseline summaries are based on the assumption that students usually start writing their feedback with overall sentiment or conclude them with good summaries. Figure 1 shows the JS divergence for the different peer-reviews and their summaries. Overall, *SOS* outperforms *First-6* and *Last-6*. A notable aspect observed in the Figure 1 is that *First-6* and *Last-6* perform better than or equally to *SOS* in some reviews. In these cases, the reviews usually have the short length of a document, which is approximately equal to the size of a summary. Hence, the length of a review has a considerable influence on summarisation quality. In addition, the presented result shows an interesting fact that students tend to write overall sentiment sentences at the end rather than at the beginning of the feedback. It means that they conclude with opinionated sentences.

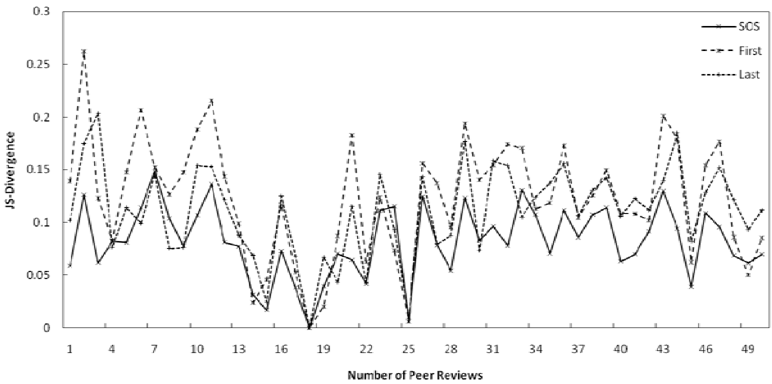


Fig. 1. Jensen-Shannon divergence between peer-review and its summary

4 Conclusions

Our study evaluated an approach for summarization of reviews using sentiment analysis. The method showed to be more accurate than the baseline using heuristics described in the literature. Such techniques can be useful to understand the impact of systems designed to teach academic writing [5] and other automated feedback systems.

References

1. Cho, K., Schunn, C.D., Charney, D.: Commenting on Writing: Typology and Perceived Helpfulness of Comments from Novice Peer Reviewers and Subject Matter Experts. Written Communication 23, 260–294 (2006)
2. Mihalcea, R., Hassan, S.: Using the Essence of Texts to Improve Document Classification. In: Proceedings of Recent Advances in Natural Language Processing (2005)
3. Baccianella, S., Esuli, A., Sebastiani, F.: SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: Proceedings of the 7th Conference on Language Resources and Evaluation LREC 2010, pp. 2200–2204 (2010)
4. Luhn, H.P.: The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development 2, 159–165 (1958)
5. Calvo, R.A., O'Rourke, S.T., Jones, J., Yacef, K., Reimann, P.: Collaborative Writing Support Tools on the Cloud. IEEE Transactions on Learning Technologies 4, 88–97 (2011)
6. Louis, A., Nenkova, A.: Automatic Summary Evaluation without Human Models. In: Proceedings of the Text Analysing Conference (2008)
7. Pang, B., Lee, L.: A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In: 42nd Annual Meeting of the Association for Computational Linguistics, pp. 271–278 (2004)

Design Dimensions of Intelligent Text Entry Tutors

Per Ola Kristensson

Computer Laboratory, University of Cambridge,
JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom
pok21@cam.ac.uk

Abstract. Intelligent text entry methods use techniques from artificial intelligence to improve entry rates. While these text entry methods are useful in situations when a full-sized keyboard is impractical or unavailable, they also require substantial training investment from users. We hypothesize that intelligent text entry tutors may reduce this time and effort. However, before we set out to design these tutors we need to consider their design space. This paper contributes to this understanding by proposing and analyzing five design dimensions: automaticity, error correction, coverage, feedback and engagement.

Keywords: Text entry, intelligent text entry, text entry tutor, typing tutor.

1 Introduction

Intelligent text entry methods use techniques from artificial intelligence (AI) to improve entry rates. Examples of such methods are handwriting and speech recognition, the touch-screen gesture keyboard SHARK^2 [3] (commercialized as ShapeWriter/Swype/T9 Trace/Flext9), and the gaze writing method Dasher [10].

What these and other such methods have in common is that design restrictions, such as the form factor of the device or the capabilities of the user, reduce the input rate compared to ten-finger touch-typing on a full-sized keyboard. To compensate for a lower input rate intelligent text entry methods infer or predict what the user intends to write [2]. A challenge for some of these methods is that users need to relearn how to write using them, either completely (e.g. [10]) or partially (e.g. [3]). While commercial typing tutors are available for full-sized keyboards (e.g. Sega's *Typing of the Dead*), intelligent text entry methods pose unique challenges for learners. We here propose and analyze five design dimensions when building text entry tutors for them.

2 Design Dimensions

The first dimension is *automaticity*. For users to write fast they need saturate motor learning. This may require users to push themselves beyond an initial performance boundary or comfort zone. It has been suggested [4] that one effective way of achieving this is to use the expanding rehearsal interval algorithm [5] (also known as spaced repetition). This algorithm asks the user to write a word according to a certain rehearsal interval. If the user writes the word correctly within a set threshold the rehearsal

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 494–496, 2011.

interval is extended according to a multiplier. Otherwise, it is assumed the user has not reached automaticity for this word and the interval is left unchanged. Thus the algorithm regulates the tradeoff between rehearsing a word so often it is a waste of the user's time and rehearsing a word so seldom that the user never progresses. If users still fail to progress past their comfort zone, we also suggest investigating whether a series of immediate repetitions can trigger a transition.

The second dimension is *error correction*. Error correction is an unavoidable aspect of text entry and users need to know how to most effectively use the error correction techniques that are provided. For some interfaces, such as multimodal mobile speech recognition, there may be more than five different ways of correcting errors [9], each with its own pros and cons. The performance benefits in understanding when to use a certain error correction strategy can be substantial. For example, words that are out-of-vocabulary may never be correctly identified by the recognizer. In such cases expert users immediately fall back to another modality.

The third dimension is *coverage*. There are hundreds of thousands of words in a language. Fortunately two phenomena dramatically reduce this space. First, words tend to follow a highly skewed power law distribution known as Zipf's law[1]. It tells us that the most frequent words in a language comprise a large fraction of the text mass. For instance, it has been observed that around 46% of the British National Corpus consists of the 100 most frequent words [4]. Hence, substantial gains can be obtained by letting users practice only the top 100–200 most frequent words initially. Second, users have both an active and passive vocabulary. Passive words are the words we understand while active words are the words we as individuals use when we write and speak. While the former is in the order of tens of thousands of words the latter is usually only in the order of thousands. Hence, once a user is well practiced on the most frequent words in the language we suggest identifying the individual user's active words (e.g. by mining sent emails) and thereafter using these for further practice. It has been argued that the distinction between active and passive words is sharp rather than gradual [6] so dividing up the words users practice into these two sets is not as arbitrary as it may initially appear.

The fourth dimension is *feedback*. Here at least three factors need to be considered. First, users need to be informed on how they are progressing and they need to be rewarded for their progress (see also the fifth dimension *engagement* below). Second, the complexities in the underlying AI algorithms can result in behavior that puzzles users [2]. Ideally, text entry tutors can explain why AI algorithms fail to recognize or predict an intended word. For instance, Kristensson [2] describes various techniques such as confidence visualization and morphing the user's input into the recognized output to help users understand how systems process their data. Third, users who are using suboptimal strategies may benefit from immediate guidance. If empirical data on common misunderstandings among users is collected into an error library [7] then systems may be able to diagnose users' errors and provide remedial instructions [7].

The fifth dimension is *engagement*. Text entry tutors have to some extent explored this before, such as in a balloon game that explicitly stated "fun" as a design goal [4] and in a writing tutor for Japanese characters which was inspired by an existing game

[1] Zipf's law estimates the probability P_r of occurrence of a word in a corpus to be $P_r \propto 1/r^{\alpha}$, where r is the statistical rank of the word in decreasing order and α is close to one.

[8]. However, these tutors were relatively simple and repetitive. Users are generally impatient and need to be quickly convinced that learning a new text entry method is worthwhile. Therefore it is important to not only hook users initially but to also keep them hooked until they are able to use the new text entry method effectively. Clanton [1] discusses how user interface design can be aided by game design in this regard.

3 Conclusions

Teaching intelligent text entry methods poses unique challenges due to several factors, such as the complexities of the underlying AI algorithms, the need for users to quickly reach automaticity for frequently used words, and the need to hook users until they are able to use the new text entry methods effectively. To guide the design of intelligent text entry tutors we here proposed and analyzed five dimensions. These reflect design issues when teaching a wide array of text entry methods. We currently use these to guide our own development and hope they will stimulate further research.

Acknowledgments. This work was supported by the Engineering and Physical Sciences Research Council (grant number EP/H027408/1).

References

1. Clanton, C.: An Interpreted Demonstration of Computer Game Design. In: 16th ACM Conference on Human Factors in Computing Systems, Conference Summary, pp. 1–2. ACM Press, New York (1998)
2. Kristensson, P.O.: Five Challenges for Intelligent Text Entry Methods. AI Mag. 30(4), 85–94 (2009)
3. Kristensson, P.O., Zhai, S.: SHARK^2: A Large Vocabulary Shorthand Writing System for Pen-Based Computers. In: 17th Annual ACM Symposium on User Interface Software and Technology, pp. 43–52. ACM Press, New York (2004)
4. Kristensson, P.O., Zhai, S.: Learning Shape Writing by Game Playing. In: 25th ACM Conference on Human Factors in Computing Systems, Extended Abstracts, pp. 1971–1976. ACM Press, New York (2007)
5. Landauer, T.K., Bjork, R.A.: Optimum Rehearsal Patterns and Name Learning. In: Gruneberg, M., Morris, P.E., Sykes, R.N. (eds.) Practical Aspects of Memory, pp. 625–632. Academic Press, London (1978)
6. Meara, P.: A Note on Passive Vocabulary. Second Lang. Res. 6(2), 150–154 (1990)
7. Ohlsson, S.: Some Principles of Intelligent Tutoring. Instr. Sci. 14, 293–326 (1986)
8. Stubbs, K.: Kana No Senshi (Kana Warrior): A New Interface for Learning Japanese Characters. In: 21st ACM Conference on Human Factors in Computing Systems, Extended Abstracts, pp. 894–895. ACM Press, New York (2003)
9. Vertanen, K., Kristensson, P.O.: Intelligently Aiding Human-Guided Correction of Speech Recognition. In: 24th AAAI Conference on Artificial Intelligence, pp. 1698–1701. AAAI Press, Menlo Park (2010)
10. Ward, D.J., MacKay, D.J.C.: Fast Hands-Free Writing by Gaze Direction. Nat. 418(6900), 838 (2002)

SMART: Speech-enabled Mobile Assisted Reading Technology for Word Comprehension

Anuj Kumar*, Pooja Reddy, and Matthew Kam

Carnegie Mellon University, Pittsburgh, PA, USA
anujk1@cs.cmu.edu

Abstract. In this study, we designed and developed two educational games on mobile phones with support for speech-recognition to examine and train the cognitive underpinnings of word reading in English as a Second Language (ESL) learners in rural India. Specifically, we tested the hypothesis that articulating a word aloud will be more advantageous for strengthening the sub-lexical components required for word reading – orthographic, phonologic, and semantic – than silently practicing it. 31 children from grades 4 and 5 learning ESL in rural India participated in the study. The results corroborated the hypothesis, suggesting that production is important for second language word reading development.

Keywords: Word Reading, Mobile Learning, Speech Recognition, Literacy.

1 Introduction

Reading and understanding written words is fundamental for literacy development. Yet it is one of the most challenging skills to acquire, especially in a second language, because it requires the integration of visual, sound, and meaning information [5], all in a new language. In order to address this need, we examined the possible benefits of practicing producing words aloud, rather than simply reading them receptively in one's mind, for ESL word reading development in rural India students of grades 4-5.

Word reading – the ability to decode (sound out) and understand written real words – is a multi-faceted construct that depends on the quality of representation of three linguistic and cognitive sub-systems: orthographic (visual script), phonologic (sound), and semantic (meaning), according to the Lexical Quality Hypothesis (LQH) [5]. When a written word is encountered, each orthographic unit must be connected to its appropriate phonological unit, allowing a learner to assign sound information to a word and thus *decode* it accurately. For instance, the letter "c" must be mapped with the sound /k/, "a" with /a/, and "t" with /t/ and so forth. A connection must also be made between this phonological representation (the sound /cat/) and its appropriate meaning (small, furry animal), and thus *semantic extraction* must also occur. If the quality of any of these sub-systems is compromised, then word reading will be hampered, and thus, reading comprehension will be impaired [4].

At the same time, it has been argued that oral production is critical for new learners of a language as an oral output provides specific input back to the mind, which in turn

[*] Corresponding author.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 497–499, 2011.

assists a learner to transition from declarative knowledge (ability to declare that you know a word) to productive knowledge (ability to fluently use the word) [1].

Integrating these strands of research, our conceptual framework is that there are three – orthographic, phonologic, and semantic – representations that are required for word reading, and the connections between them can be processed productively or receptively. Based on this framework, we expect that saying the name of a picture out loud (productive processing) will reinforce semantic extraction more strongly than if the link is receptively processed, i.e. matching a picture and its name [1, 6]. Thus, our experiment consisted of two training conditions: 1) Receptive (Re); and 2) Productive (Pr), and we hypothesized that: (H1) Productive training will be more beneficial for word reading than Receptive training.

2 System and Game Design

Due to prior success of using games for education [2, 3], our intervention consisted of two English literacy learning games: Market Game, and Farm Game. These were prototyped using ActionScript 3.0 for Nokia N810. Our game designs drew on our experiences from prior field studies of traditional Indian games that rural children enjoyed most, including physical tag-like games that have actions such as *catching* or *evading* a player [3]. The speech recognizer was fine-tuned for the accent, noise and speaking rate of the participants, and the final recognition accuracy was 91%.

In the Market Game (Figure 1A), the boy character had to travel from home to the market to buy items while avoiding monkeys en-route. At the market, depending on the experimental condition that the user had been assigned to, she/he purchased items by either *selecting* the correct item that corresponded to the said word (Figure 1C, Re condition), or by *saying the word aloud* that corresponded to the image displayed (Figure 1D, Pr condition). Similarly, in the Farm Game (Figure 1B), the objective was to save a farm by catching all the thieves and retrieve the items that they had stolen in one of the two ways described above for the Market Game.

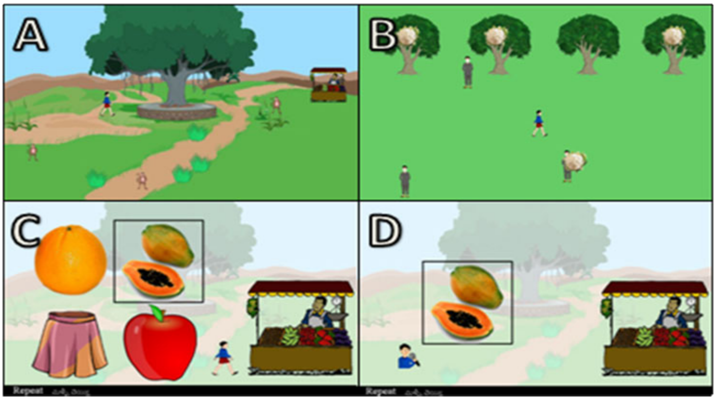


Fig. 1. (A) shows the boy moving towards the shop in the market game, while the monkeys attempt to catch him; (B) shows the boy catching the thief in the farm game; (C) Re training condition in market game; (D) Pr training condition in market game

3 Experiment and Early Results

31 participants (18 boys) participated in this study. They were 9-13 year olds (M=10.5) and were in grades 4-5. All participants were attending a public school in a rural part of South India. The experimental design was a pre-post test block design, and the intervention comprised of the above described games. The outcome variable, word reading, was tested before and after each game. 25 words were selected from grade 4-5 level government-issued textbooks for the intervention. Each child played both games, but was randomly assigned to one of the two experimental conditions. The findings from this study demonstrated that with even 30 minutes of targeted practice of words and their meanings, it is possible to increase word reading scores, regardless of whether it is productive or receptive; however, as predicted, we found that productive training is significantly more beneficial for word reading than receptive training, $t = -3.01$, $p < .05$. This is in line with SLA theories, which stress that output of linguistic forms consolidates knowledge [1, 6]. In this case, when a learner is forced to make a link between a word's meaning and its pronunciation productively, the bridge between a word's meaning and its name (sound) are more strongly reinforced, making it more deeply embedded in memory.

Acknowledgments. We thank Anuj Tewari, Venkatesh Keri, Rajat Agarwal and all others who helped in developing and deploying the software.

References

1. De Bot, K.: The psycholinguistics of the Output Hypothesis. Language Learning 46, 529–555 (1996)
2. Gee, J.P.: What Video Games Have to Teach Us About Learning and Literacy. P. Macmillan, Basingstoke (2004)
3. Kam, M., Mathur, A., Kumar, A., Canny, J.: Designing Digital Games for Rural Children: A Study of Traditional Village Games in India. In: Proc. of ACM Conference on Human Factors in Computing Systems (CHI 2009), Boston, Massachusetts, April 4-9 (2009)
4. Perfetti, C.A.: Reading ability: Lexical quality to comprehension. Scientific Studies of Reading 11, 357–383 (2007)
5. Perfetti, C.A., Hart, L.: The lexical quality hypothesis. In: Verhoeven, L., Elbro, C., Reitsma, P. (eds.) Precursors of Functional Literacy, vol. 11, pp. 67–86. John Benjamins, Amsterdam (2001)
6. Swain, M., Lapkin, S.: Problems in output and the cognitive processes they generate: A step towards second language learning. Applied Linguistics 16, 371–391 (1995)

Enhancing the Error Diagnosis Capability for Constraint-Based Tutoring Systems

Nguyen-Thinh Le and Niels Pinkwart

Clausthal University of Technology, Germany
{nguyen-thinh.le,niels.pinkwart}@tu-clausthal.de

Abstract. Constraint-based modelling techniques have been demonstrated a useful means to develop intelligent tutoring systems in several domains. However, when applying CBM to tasks which require students to explore a large solution space, this approach encounters its limitation: it is not well suited to hypothesize the solution variant intended by the student, and thus corrective feedback might be not in accordance with the student's intention. To solve this problem, we propose to adopt a probabilistic approach for solving constraint satisfaction problems.

Keywords: ITS, weighted constraint-based model, cognitive diagnosis.

1 Introduction

The constraint-based modelling (CBM) approach [5] has been successfully employed in several domains, such as diagnosing grammar errors in natural languages [3], building intelligent tutoring systems for SQL [4]. One of the strengths of this approach is that it does not require an enumeration of every correct solution for modelling, nor is it necessary to anticipate possible errors made by students. Instead, a number of domain principles and properties of correct solutions for a problem need to be specified. However, this approach encounters its limitation when applying it to tasks which have a large solution space. Corrective feedback derived from results of constraint-based error diagnosis, might be misleading, because the solution strategy the student intended to implement is not the same one the constraints are based on. This problem has been identified and discussed in [2] and [6]. This problem raises the need to hypothesize the student's intention in terms of the applied solution strategy during the process of diagnosing errors. Once the solution strategy of the student has been identified, it makes sense to evaluate constraints in the context of that specific solution strategy only. This paper introduces a weighted constraint-based model adopting a probabilistic approach for solving constraint satisfaction problems: each constraint is enriched with a weight value indicating the importance of the constraint. Applying this model, a tutoring system is able to decide on the most plausible hypothesis about the solution strategy intended by the student.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 500–503, 2011.

2 A Weighted Constraint-Based Model For ITS

In order to be able to identify shortcomings in a student solution and to provide appropriate corrective hints according to the solution strategy pursued by the student, a tutoring system needs to cover a space of possible solutions and the student solution needs to be analyzed thoroughly. In the approach proposed in this paper, the weighted constraint-based model serves these two purposes.

Semantic Table: Instead of using a single ideal solution to capture problem-specific requirements as in [2], the model introduced in this paper uses a so-called *semantic table* which comprises two ideas: 1) it models several solution strategies, and 2) it represents model solutions in a relational form. The first characteristic serves to hypothesize the most plausible strategy underlying a student solution. The second one has the advantage that solution variants (e.g., created by alternative orderings of solution components) can easily be covered. The following table illustrates a partial semantic table for the problem *"Calculate the return after investing an amount of money at a constant yearly interest rate"*, covering one possible solution strategy (tail recursive), where *CI* and *SI* are abbreviations for clause index and subgoal index, respectively.

Strategy	CI	Head	SI	Subgoal	Description
Tail recursive	1	p(S,_,P,Ret)	1	P=0	Recursion stops
	1	p(S,_,P,Ret)	2	Ret=S	Recursion stops
Tail recursive	2	p(S,R,P,Ret)	1	P>0	Check period
	2	p(S,R,P,Ret)	2	NS is S*R+S	Calculate new sum
	2	p(S,R,P,Ret)	3	NP is P-1	Update period
	2	p(S,R,P,Ret)	4	p(NS,R,NP,Ret)	Recur with new period

Constraints: We distinguish between *general constraints* and *semantic constraints*. Constraints of the former type are used to model domain-specific principles, which every solution variant of any problem must adhere to and are independent of problem-specific requirements. For instance, in programming, to evaluate an arithmetic expression, all variables must be instantiated. Such a domain principle can be modeled by means of general constraints which can be instantiated by the following constraint schema, where the problem situation X and the condition Y can be composed of elementary propositions using conjunction or disjunction operators.

IF problem situation X is relevant **THEN** condition Y must be satisfied

Semantic constraints are used to check the semantic correctness of a student solution. Constraints of this type require problem-specific information specified in the semantic table and have the following schema, where *STS* is an abbreviation for *student solution.*

IF in the semantic table, a component X exists and satisfies condition α
THEN in the STS, a corresponding component exists and satisfies α

Transformation Rules: To extend the solution space for a problem that involves mathematical expressions, transformation rules can be defined based on mathematical theorems, e.g., distributive and commutative laws.

Constraint Weights: As pointed in Section 1, constraint-based tutoring systems might provide misleading corrective feedback. We need a means to search the most plausible hypothesis about the student's solution variant. For this purpose, we exploit approaches to softening constraints in constraint satisfaction problems (CSP). The weighted constraint-based model proposed here adopts the probabilistic CSP approach [1] for error diagnosis. Following this approach, each constraint is attached a *constraint weight*, indicating the measure of importance. Weight values are taken from the interval $[0;1]$. The value close to 0 indicates the weight for constraints which model most important requirements. Constraints of the latter type can be considered *hard constraints.* The plausibility of each hypothesis is calculated using the formula: $Plausibility(H) = \prod_{i=1}^{N} W_i$, where W_i is the weight of a violated constraint.

Error Diagnosis: Given a student solution, the process of error diagnosis starts to match the solution against each of the solution strategies specified in the semantic table. This process initialises *global mappings* representing hypotheses about the strategy underlying the student solution and this level of matching is referred to as *strategy level.* Then, the process continues to generate hypotheses about the student's solution variant by matching the components of the student solution against the corresponding ones of the selected solution strategy. The matching process results in *local mappings* representing hypotheses about the student's solution variant. They are used to complete global mappings. This level of matching is called *solution variant level.* After hypotheses have been generated, the process of error diagnosis evaluates each hypothesis with respect to its plausibility. On the solution variant level, the most plausible solution variant of the student solution is determined by choosing the hypothesis with maximal plausibility score. On the strategy level, the hypothesis with the highest plausibility score is considered the solution strategy being implemented in the student solution. Diagnostic information is derived from constraint violations based on the best hypothesis.

3 Conclusion

In this paper, we have argued that the classical CBM approach is not well-suited to build intelligent tutoring systems for tasks which have a large solution space. Here, the process of error diagnosis needs to hypothesize the solution variant applied by the student. For this purpose, we introduced the weighted constraint-based model which adopts the idea of probabilistic techniques for solving constraint satisfaction problems.

References

1. Fargier, H., Lang, J.: Uncertainty in Constraint Satisfaction Problems: a Probabilistic Approach. In: Moral, S., Kruse, R., Clarke, E. (eds.) ECSQARU 1993. LNCS, vol. 747, pp. 97–104. Springer, Heidelberg (1993)
2. Martin, B.: Intelligent Tutoring Systems: The Practical Implementation Of Constraint-based Modelling. PhD thesis, University of Canterbury (2001)

3. Menzel, W.: Diagnosing Grammatical Faults - a Deep-modelled Approach. In: AIMSA, pp. 319–326 (1988)
4. Mitrovic, A., et al.: Constraint-based Tutors: A Success Story. In: 14th Int. Conf. on Industrial and Engineering Appl. of AI and Expert Systems, pp. 931–940 (2001)
5. Ohlsson, S.: Constraint-based Student Modeling. In: Greer, J.E., et al. (eds.) Student Modelling: The Key to Individualized Knowledge-based Instruction, pp. 167–189. Springer, Heidelberg (1994)
6. Woolf, B.P.: Building Intelligent Interactive Tutors. Morgan Kaufmann, San Francisco (2009)

Question Taxonomy and Implications for Automatic Question Generation

Ming Liu and Rafael A. Calvo

University of Sydney, Sydney, NSW 2006, Australia

Abstract. Many Automatic Question Generation (AQG) approaches have been proposed focusing on reading comprehension support; however, none of them addressed academic writing. We conducted a large-scale case study with 25 supervisors and 36 research students enroled in an Engineering Research Method course. We investigated trigger questions, as a form of feedback, produced by supervisors, and how they support these students' literature review writing. In this paper, we identified the most frequent question types according to Graesser and Person's Question Taxonomy and discussed how the human experts generate such questions from the source text. Finally, we proposed a more practical Automatic Question Generation Framework for supporting academic writing in engineering education.

Keywords: Academic Writing Support, Question Taxonomy, Question Generation.

1 Introduction

The purpose of academic writing is to generate new knowledge through a review of what is currently known on a given topic. Steward [1] defines a good review as "*Comprehensive, Relevant, A synthesis of key themes and ideas, Critical in its appraisal of the literature, and Analytical developing new ideas from the evidence.*" Simple generic questions are often provided to trigger student's reflection. For example, the following generic question is to ask the student to critically evaluate the literature, *Have you clearly identified the contributions of the literature reviewed?* However, such questions are too general and not likely to provide strong support in the process of writing on a specific topic. Our previous study [2] showed that specific questions generated from citation sentences significantly outperformed generic ones in supporting learning. Nevertheless, several open questions remain: what types of trigger questions are commonly used by human experts and how useful are they? how do human experts themselves generate trigger questions from the source text?

Question Taxonomies have been proposed according to different application domains. The best known question taxonomy was proposed by Graesser and Person [3] based on their study investigated the questions asked in tutoring sessions on college research methods and algebra. In our study, we adapted Graesser and Person's Question Taxonomy.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 504–506, 2011.

2 A Case Study

We used 13 frequent question types defined in Graesser and Pearson's Question Taxonomy [3] to ask two human to independently annotate 125 supervisors questions, generated from 36 PhD students' literature review writing. Cohen's Kappa coefficient is 0.57 (n=13; N=125; k=2). These students are asked to give score on each question from 1 to 5 based on quality measures (This question makes me reflect what I have written and is useful). Table 1 shows seven frequent question types in our dataset.

Table 1. Graesser and Pearson's question taxonomy with examples of questions from academic supervisors

Question Type	Examples
Verification: implied yes/no/ answers	Is it possible to reuse some of previous routing techniques, for example those used in cellular networks, in the NGMN?
Concept: Who,When ,What,Where?	Can you give more details about the Generalized Beam Theory?
Causal Antecedent: what event causally led to an event?	Why network coding in [13] can increase the system throughput?
Causal Consequence: What is the consequence of an event?	What is the likely consequence of the nonlinear stress-strain curve?
Procedural: What instrument or plan allows an agent to accomplish a goal?	How does the formation of mechanical twins provide corrosion resistance?
Judgmental: What do you think of X?	How do you see the Generalized Beam Theory being applied in your project?

It is found that Concept (29), Causal (29), and Procedural (23) questions are more frequent than Judgmental (13) and Verification (11) questions[1]. Moreover, the Verification (4.75), Concept (4.12) and Procedural questions (4.34) have higher scores in average than Causal (3.88) and Judgmental (3.92) questions. This result suggests that supervisors like to generate both simple and deep questions. Surprisingly, some simple questions outscore the deep questions indicating that the conceptual questions are as important as procedural or causal questions and they should be considered when designing the question templates.

In order to investigate how the questions generated from the source text can be used in AQG, we organized them in the following four abstract levels: Lexicon, Sentence, Discourse and Background Knowledge. We found that the number of questions generated from Lexicon and Sentence Level took up 56.8% (71 out of 125) while the discourse level took up 14.4% and the background knowledge level 28.8%. The result indicates that the opportunities for developing an AQG system. For the Concept type, questions are used to ask students to critically identify the key concept, such as Coating (*Material*),Generalized Beam Theory (*Theory*),the field of Fluid Mechanics (*Research Field*) and SVM Algorithm (*Algorithm*), thus most of the questions are generated at lexicon level. The Judgmental questions are often used to ask students to identify its relevancy.

[1] The number in this sentence indicates the frequency of each question type.

An example of Judgmental question, What impact would the proposed project have on the field of fluid mechanics? In this case, the field of fluid mechanics is a Research Field concept. For the Causal type, questions are often asked about why a *method* or *technique* is effective. The source sentences are usually causal or express a *result.* For Procedural questions, supervisors like to ask about how a *method/schema/material is applied.*

3 Future Work on Automatic Question Generation Framework

Based on the findings, we propose an AQG framework for academic writing support. The input to the system is academic text in natural language text and the output is the set of specific trigger questions generated. In stage 1, a sentence is extracted, simplified and parsed and the term-sentence vector space model is build for finding the key concept. In stage 2, the sentence type and key concept are identified by using the sentence classifier and key phrase extractor separately. As we discussed before, a sentence can be used to express *Application, Result, Comparative Test and Opinion* while a concept *research field, algorithm, theory*. In stage 3, the questions are generated by using a rule-based approach based on the semantic meaning of the sentence or the concept. The Verification, Concept and Procedural, Causal and Judgmental questions are the major question types used in the question template design. In stage 4, a statistical question ranker scores the question. In conclusion, we analyzed 125 trigger questions generated by engineering supervisors, to support students' literature review writing. Six frequent question types based on Graesser and Person's question taxonomy were identified as useful to design question templates. The results showed that 56.8% questions generated from Lexicon and Sentence level without complex inference processing, which indicates that there are many potential questions can be exploited by using current NLP techniques.

Acknowledgements

The work was supported by Australian Research Council Project DP0986873.

References

1. Steward, B.: Writing a Literature Review. The British Journal of Occupational Therapy 67, 495–500 (2004)
2. Liu, M., Calvo, R.A., Rus, V.: Automatic Question Generation for Literature Review Writing Support. In: Tenth International Conference on Intelligent Tutoring Systems, pp. 45–54. Springer, Pittsburgh (2010)
3. Graesser, A.C., Person, N.K.: Question asking during tutoring. American Educational Research Journal 31, 104–137 (1994)

Agent-Mediated Immersion in Virtual World: The Implications for Science Learning

Chee-Kit Looi[1], Longkai Wu[1], Beaumie Kim[1], and Chunyan Miao[2]

[1] Natioanl Institute of Education, Nanyang Technological University, Singapore
[2] School of Computer Engineering, Nanyang Technological University, Singapore
{cheekit.looi,longkai.wu,beaumie.kim}@nie.edu.sg,
ascymiao@ntu.edu.sg

Abstract. In a virtual world environment, mediated immersion can be important in enhancing students' engagement in learning. This paper describes the study of agent-mediated immersion in a virtual world called Chronicles of Virtual Singapura that seeks to promote students' science learning. The results of an empirical study provide us some evidences that the agent-mediated immersion could have positive effects on students' science learning, especially to those with less prior domain knowledge.

Keywords: Virtual World, Agent-Mediated Immersion, Science Learning.

1 Introduction

Recent research has emphasized creating virtual learning worlds that provide students with a sense of immersion into the content, with the ability to both manipulate the content and change the content to derive new understanding [1]. Dede and Barab [2] note that immersive designs in virtual world (e.g., immersive interfaces [3]) offer promising vistas for improving science education, whereas emerging technologies, such as agent technology [4] can be incorporated to address core issues of student's engagement, mastery of sophisticated knowledge and skills, transfer of learning, and attaining scale. However, Trindade et al. [5] find that not all students' sense of immersion can contribute to their conceptual understanding of science even they provide substance to abstract concepts. Coffman and Klinger [1] argue that students need scaffolding to solve problems in immersive environments. Meanwhile, other studies show that pedagogical agents, which are life-like personas, can execute behaviors that involve emotive responses, interactive communication, and effective pedagogy, to scaffold and optimize students' learning by exploiting their characteristics [6].

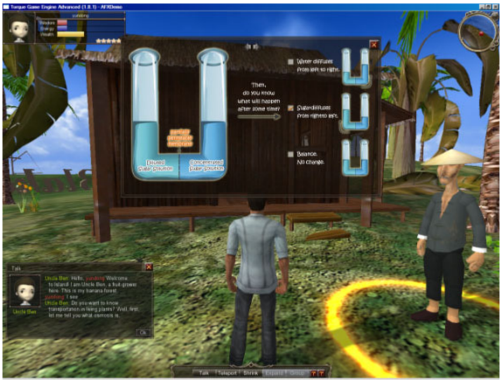
These prior studies motivate us in exploring possibilities by developing and deploying innovative pedagogical agents into a virtual world called Chronicles of Virtual Singapura (Fig. 1), as scaffolding tools, to personalize and augment students' immersive learning experiences, for the purpose of enhancing engagement and thus promoting science learning. In this paper, we describe the study of agent-mediated immersion that serves these purposes, by incorporating pedagogical agents (such as remembrance agents and teachable agents) into the environment of Chronicles of Virtual Singapura. We intend to use such an immersive environment to actively engage secondary school

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 507–509, 2011.

students in learning a topic in biology, namely, transport in living things and in practicing in-depth experiential learning through agent scaffolding.

2 Empirical Study

An empirical study was conducted in a local male high school with four 45-minutes classroom sessions. The 33 participants were from 14 to 15 years old, who are domain novices in the biology subject. They individually completed the exploration tasks in the virtual environment and the 22 multi-choice domain questions, which were examined and approved by the teacher, in their pre- and post- tests. They were also asked to draw their individual mind maps to explain plant growth after the treatment (32 students completed the entire task). Comparatively, a control class of 38 students, who had regular lectures in diffusion and osmosis, was also asked to draw individual mind maps when they finished the chapter. All the maps were collected and compared to see the difference of conceptual changes between the two classes.


Torque Game Engine Advanced (1.8.1) - AFXDemo
yunfeng
Wisdom
Energy
Wealth
partially permeable membrane
Diluted Sugar Solution
Concentrated Sugar Solution
Talk
Uncle Ben
Ok
Talk
Teleport
Shrink
Expand
Group

Then, do you know what will happen after some time?

□ Water diffuses from left to right.

☑ Sugardiffuses from rightto left.

□ Balance. No change.

Uncle Ben: Hello, yunfeng! Welcome to Island! I am Uncle Ben, a fruit-grower here. This is my banana forest.
yunfeng: I see.
Uncle Ben: Do you want to know transportation in living plants? Well, first, let me tell you what osmosis is.

Fig. 1. Overview of Chronicles of Virtual Singapura: Help Uncle Ben to Save the Tree

3 Results

As to the scoring of pre-/post- test, one point is assigned to one question when students marked the correct choice and zero points, otherwise. As shown in Table 1, the participants achieved mean scores in pre-test (Mean = 12.94, SD = 2.85) and post-test (Mean = 14.13, SD = 2.57) . We categorized the participants into *High* and *Low* group as to their prior biological knowledge, which is relevant to diffusion, osmosis and transport in plant, shown in the pre-test. Participants scored above the mean score in pre-test (Mean = 12.94) were included in the *High* group (18 participants) and the rest were included in the *Low* group (14 participants).

The results of a paired samples t-test show that the 32 participants as a whole has experienced a significant increase as to score mean (t = 2.48, p = .019). Specially, the 14 students in the *Low* group showed a more prominent increase on score mean (t = 3.88, p = .002). Meanwhile, the 18 students in the High group exhibited a slight increase on score mean (t = .20, p = .847).

Table 1. Pre-/Post- Tests Result

	Pre-test Score[1]	Post-test Score	t-test[2]	Effect Size (Cohen's *d*)[3]
Total	12.93 (2.85)	14.13 (2.57)	2.48*	0.44
Low Group	10.43 (1.70)	13.00 (2.75)	3.88*	1.12*
High Group	14.89 (1.84)	15.00 (2.11)	0.20	0.05

[1] Standard deviation are shown in parentheses.

[2] $p < .05$

[3] $d > 0.8$ is considered as "large" effect.

The analysis of students' mind maps sought to establish students' conceptual change on plant growth in terms of quantity of concepts, quantity of propositions, quantity of elaborations and quantity of pictures. An ANOVA test shows significant difference, between the control class and the experimental class, in the means across three categories of map analysis: quantity of concepts (F = 1.84, p < .05), quantity of elaborations (F = 3.32, p < .05) and quantity of pictures (F = 1.45, p < .05).

4 Conclusions and Future Work

Overall, the preliminary results indicate that agent-mediated immersion has the potential to bring positive effects in science learning. First, the statistical results of 32 students in classroom study support that all students could benefit from the virtual agent-mediated environment as to the knowledge gains, especially to those with less prior knowledge. Second, the comparison on mind maps between the experimental and control classes indicates that the experimental class could tend to gain a deeper conceptual change of major concepts and develop a more robust memory of scientific information. In future studies, we will further collaborate with school teachers in curriculum design and examine how such virtual environments with agent-mediated immersion can be incorporated in pedagogical practices in classroom learning and teaching.

References

1. Coffman, T., Klinger, M.B.: Utilizing virtual worlds in education: The implications for practice. International Journal of Social Sciences 2(1), 29–33 (2007)
2. Dede, C., Barab, S.: Emerging Technologies for Learning Science: A Time of Rapid Advances. Journal of Science Education and Technology 18(4), 301–304 (2009)
3. Dede, C.: Immersive Interfaces for Engagement and Learning. Science 323, 66–69 (2009)
4. Chase, C., et al.: Teachable agents and the protégé effect: Increasing the effort towards Learning. Journal of Science Education and Technology (2009)
5. Trindade, J., Fiolhais, C., Almeida, L.: Science learning in virtual environments: a descriptive study. British Journal of Educational Technology 33(4), 471–488 (2002)
6. Person, N.K., Graesser, A.C.: Pedagogical Agents and Tutors. In: Guthrie, J.W. (ed.) Encyclopedia of Education, pp. 586–589. Macmillan, New York (2003)

Virtual Manipulatives in a Computer-Based Learning Environment: How Experimental Data Informs the Design of Future Systems

Maria Mendiburo and Gautam Biswas

Dept. of EECS/ISIS, Vanderbilt University, Nashville, TN, USA
{maria.mendiburo,gautam.biswas}@vanderbilt.edu

Abstract. In the fall of 2009, we conducted an experiment in which we compared virtual and physical fractions manipulatives. We facilitated instruction in the virtual condition using a program called Virtually Fractions, which we designed specifically for the experiment. The instructional elements of Virtually Fractions align with a commercial curriculum that integrates physical manipulatives into instruction. As such, our experiment allowed us to test the non-instructional elements of the Virtually Fractions system within a proven instructional model. In this paper, we discuss the design of the system as well as what the data collected during our experiment tells us about the ways students interacted with the system.

Keywords: Computer-based learning environments, virtual manipulatives, mathematics education, fractions.

1 Introduction

Theoretical literature (e.g. [1]) suggests that many of the practical and pedagogical difficulties associated with manipulatives may be reduced or eliminated if teachers use virtual rather than physical manipulatives during instruction about fractions, but there are few methodologically rigorous studies that test this hypothesis. To address this gap in the mathematics literature, we conducted a teaching experiment that compared the efficiency and the effectiveness of physical and virtual manipulatives in 5th and 6th grade mathematics classrooms. The results of the study indicated that students learn basic fraction concepts equally well using virtual and physical manipulatives. In addition, when teachers give students the same amount of time to work on practice activities, students who use virtual manipulatives complete more practice activities than students who use physical manipulatives [2].

We facilitated instruction in the virtual manipulative treatment condition using a set of instructional scripts and a computer-based learning environment (CBLE) that we designed specifically for the experiment. We named our CBLE Virtually Fractions[1]. We intentionally designed the instructional scripts and learning activities in

[1] We would like to thank Laura Goin for leading the technical aspects of the development of Virtually Fractions.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 510–512, 2011.

Virtually Fractions to align very closely with the instructional scripts and learning activities from a popular, commercially-available curriculum that fully integrates physical manipulatives into classroom-based instruction about basic fractions concepts. In the rest of this paper, we describe Virtually Fractions in more detail, and we discuss what the data collected during our experiment tells us about the ways students interacted with the system

2 Description of Virtually Fractions

A virtual Fractions Kit with manipulative representations of a whole, halves, fourths, eighths, and sixteenths appears in the lower, left-hand quadrant of the Virtually Fractions interface. Students can drag pieces from the Fractions Kit into the Workspace that appears in the upper, left-hand quadrant of the interface. The practice exercises students complete using Virtually Fractions appear in the right half of the interface. The practice exercises in Virtually Fractions align with the student workbook pages from the commercial curriculum in both appearance and content. Students receive immediate feedback about the accuracy of their response to each practice exercise in that the system highlights correct responses in green and incorrect responses in red. The system tracks the total number of practice exercises completed by each student, the total number of practice exercises answered correctly, and the total number of practice exercises answered incorrectly on each day and week of instruction across both weeks of instruction during the intervention. Students complete assessments on the fifth and tenth days of instruction.

3 What the Data Tell Us about Different Learners

We collected data for a total of thirty-three students that used Virtually Fractions. When analyzing the data, we quickly identified two students as consistently high-achievers, five students as consistently low-achievers, and one student as a consistently average-achiever. All of these students received scores on the practice exercises and assessments that were within the same range throughout the intervention.

However, the data we collected on the practice exercises and assessments completed by the other twenty-seven students told a more complicated story. There was a group of students who scored much higher on the assessments than the practice exercises, and another group that had the opposite results. Several other students scored significantly higher on the practice exercises and assessments during one of the two weeks of instruction.

When examining the total number of practice exercises completed along with the percentage of practice exercises students answered correctly, we noted that the majority of students who correctly answered a high percentage of practice exercises completed less than the maximum number of practice exercises, while the students who completed the maximum or close to the maximum number of practice exercises tended to answer them incorrectly.

Finally, by examining the types of errors students made on the assessments, we were able to understand other differences between the learners who used Virtually

Fractions. In some cases, students made errors that we associate with common misunderstandings about basic fractions concepts. Other students made errors that are associated with misunderstandings about other content domains not explicitly taught during Virtually Fractions

Clearly, the data we collected about students using Virtually Fractions give us some understanding about the types of learners who used the system, but it falls far short of providing a complete set of information in many ways. The system can identify high and low-achieving students, but for the majority of students, the system cannot give us a clear understanding of their knowledge of basic fractions concepts. This is because the multiple forms of assessment included in the system provide conflicting or inconsistent information about what students understand, and they provide very little information about the types of misconceptions students might have about what they are learning. Perhaps most notably, since the system does not track students' actions with the manipulatives, it provides no information about how these tools impact students' learning.

4 Designing a New Instructional Model for a More Sophisticated CBLE

We collected valuable data about different learners using Virtually Fractions, but the system falls short of ideal in many ways. This is not surprising considering we designed the system to mirror an existing curriculum that does not utilize technology. Our research team recently began the process of designing a new instructional model for a more sophisticated CBLE that takes full advantage of all the benefits technology can provide. We intend to use a similar interface in the new system, and we intend to use similar manipulatives since our data indicated students who learned fractions using these tools achieved similar levels of understanding as students who learned fractions using a commercial curriculum and physical manipulatives. We also intend to design an instructional model intended for use in classroom settings. However, the new instructional model will expand upon the previous instructional model in many ways. By expanding upon the previous instructional model and taking full advantage of the benefits technology can provide, we expect to see a greater impact on student achievement with the new system than what we achieved in our previous experiment.

References

1. Clements, D.: 'Concrete' Manipulatives, Concrete Ideas. CIEC 1(1), 45–60 (1999)
2. Mendiburo, M.: Virtual Manipulatives and Physical Manipulatives: Technology's Impact on Fraction Learning. Unpublished doctoral dissertation, Peabody College of Vanderbilt University, Nashville, TN (2010)

Typed versus Spoken Conversations in a Multi-party Epistemic Game

Brent Morgan[1], Candice Burkett[1], Elizabeth Bagley[2], and Arthur Graesser[1]

[1] University of Memphis, Psychology, Institute for Intelligent Systems,
365 Innovation Drive, Memphis, TN 38152, USA
[2] University of Wisconsin-Madison, Educational Psychology, Educational Sciences Building,
Room 1078D, 1025 West Johnson Street, Madison, WI 53711

Abstract. Multi-party chat is a standard feature of popular online games and is increasingly available in collaborative learning environments. This paper addresses the differences between spoken and typed conversations as high school students interacted with the epistemic game Urban Science. Coh-Metrix analyses showed that speech was associated with narrativity and cohesion whereas typed input was associated with syntactic simplicity and word concreteness. These findings suggest that the modality in group communication should be considered.

Keywords: distance learning, epistemic games, natural language processing.

1 Introduction

There is a large body of research on differences between oral and written one-way communication [1, 2] and interactive dialogues [3], but little is known about these differences in a group setting. This study compared speech and chat in the context of the epistemic game, *Urban Science,* designed to simulate an urban planning practicum experience created by education researchers at the University of Wisconsin-Madison [4]. During the game, players communicate with each other and an adult mentor. These conversations were analyzed using Coh-Metrix, which is a computational linguistic tool that measures text cohesion and difficulty on a range of word, sentence, paragraph, and discourse dimensions. Recently, a principal components analysis (PCA) reduced 53 Coh-Metrix measures to five major dimensions of text: narrativity, referential cohesion, situation model cohesion, syntactic simplicity, and word concreteness [5]. We used these language-discourse components and more superficial aspects of the text (number of sentences, words and words per sentence) to better understand whether and how oral and typed communication differ. We expected that narrativity would favor the spoken condition because it is associated with oral language. The two cohesion measures should also favor the spoken condition because multiple conversational threads in chat may create cohesion breakdowns. Finally, students in the typed condition could edit contributions, so greater syntactic simplicity and more concrete words were expected.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 513–515, 2011.

2 Method

21 high school-aged participants played the epistemic game, Urban Science, for 10 hours over 3 days. Participants worked in teams and interacted with 2 trained mentors. Upon arrival, students were randomly assigned to one of two conditions: typed (interacted through an internal chat program) or spoken (communicated orally).

3 Results and Discussion

Our analyses focused on identifying the linguistic differences between typed and spoken language during student and mentor conversations. The dependent variables contained the 5 language-discourse dimensions and 3 superficial aspects (see Introduction). For each dependent variable, a mixed analysis of variance was conducted on the z-scores (deep-level) or numerical counts (superficial). The results are displayed in Table 1.

Table 1. Coh-Metrix Analysis of spoken and typed conversations in Urban Science

	Typed		Spoken				
	M	*SD*	*M*	*SD*	*F*	*p*	η^2
Narrativity	0.94	0.33	1.46	0.21	28.17	0.01	0.88*
Referential Cohesion	-0.62	0.78	-0.09	0.49	6.24	0.07	0.61
Situation Model Cohesion	-0.26	0.73	1.03	0.84	60.97	0.00	0.94*
Syntactic Simplicity	0.57	0.44	0.12	0.54	4.93	0.09	0.55
Word Concreteness	-1.76	0.82	-2.43	0.53	63.53	0.00	0.94*
Total Number of Words	342.31	173.20	1,001.33	659.70	13.80	0.02	0.78*
Total Number of Sentences	32.05	18.15	71.97	49.00	8.81	0.04	0.69*
Words per Sentence	11.38	1.81	14.11	4.08	10.93	0.03	0.73*

*p< .05

As predicted, conversations in the spoken conditions were significantly higher in narrativity, situation model cohesion, and marginally higher in referential cohesion. Also, as predicted, typed conversations were higher on word concreteness and marginally higher on syntactic ease. In addition, we predicted that the spoken condition would have higher values for the aggregate measures (number of word and number of sentences) and the results confirmed the hypotheses.

4 General Discussion

Our analyses confirmed that conversations in the spoken condition were more narrative and cohesive (global), whereas typed conversations contained more concrete words and simpler syntax (local). The superficial aspects of the texts indicated that the players were more verbose in the spoken condition. These results show that when using NLP to process student contributions, the implementation of the AI needs to take the medium of communication into consideration. In particular, if multiple conversational threads in group communication are creating breaks in discourse cohesion the interface and discourse management facilities must find ways to connect the content of conversational turns to the appropriate points in the conversation.

Acknowledgments. This work was funded in part by the Macarthur Foundation, National Science Foundation (REC-0347000, DUE-091934, DRL-0918409, DRL-0946372, BCS 0904909, DRK-12-0918409), the Gates Foundation, and U.S. Department of Homeland Security (Z934002/UTAA08-063). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of these funding agencies, cooperating institutions, or other individuals. We thank Zhiqiang Cai for the Coh-Metrix software development.

References

1. Biber, D.: Variation across speech and writing. Cambridge University Press, Cambridge (1988)
2. Louwerse, M.M., McCarthy, P.M., McNamara, D.S., Graesser, A.C.: Variation in Language and Cohesion Across Written and Spoken Registers. In: Forbus, K., Gentner, D., Regier, T. (eds.) Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society, pp. 843–848. Erlbaum, Mahweh (2004)
3. Graesser, A.C., Jeon, M., Yang, Y., Cai, Z.: Discourse Cohesion in Text and Tutorial Dialogue. Information Design Journal 15, 199–213 (2007)
4. Bagley, E.S., Shaffer, D.W.: When People Get in the Way: Promoting Civic Thinking Through Epistemic Game Play. International Journal of Gaming and Computer-Mediated Simulations 1, 36–52 (2009)
5. Graesser, A.C., McNamara, D.S.: Computational Analyses of Multilevel Discourse Comprehension. Topics in Cognitive Science (in press)

Statistical Relational Learning in Student Modeling for Intelligent Tutoring Systems

William R. Murray

Boeing Research and Technology
P.O. Box 3707, MC 7L-66
Seattle, Washington 98124-2207
william.r.murray@boeing.com

Abstract. Statistical Relational Learning (SRL) provides a common language to express diverse kinds of learner models for intelligent tutoring systems that are broadly applicable across different domains or applications. It provides new more expressive user modeling capabilities, such as the ability to express (1) probabilistic user models that model causal influence, *with* feedback loops allowed, (2) logical rules with exceptions, and (3) both hard and soft constraints in first-order logic. Practically, for example, SRL learner models can facilitate building team user models and user models for collaborative instruction by leveraging social network analysis. They can also facilitate building learner models for affective computing that simultaneously model inferences from affect to cognition and cognition to affect.

1 Introduction

Statistical Relational Learning (SRL) is a kind of machine learning that performs inference over multiple kinds of objects and multiple kinds of relationships expressed using first-order logic (FOL) formulas.[1] Weighted formulas represent soft constraints while unweighted ones represent hard constraints. SRL (Markov Logic Networks in particular) subsumes HMMs, Bayesian Networks (BNs), Dynamic Bayesian Networks, and (in the limit of infinite weights) FOL. It can be viewed as a common interface language for artificial intelligence [1].

2 A Common Language for User Modeling

Similarly, we can view SRL as a common language for user modeling in intelligent tutoring systems. Weights for formulas can either be provided subjectively, or, preferably, from data collected by educational data mining. The same user model could be ported across domains, plugging in new domain constants and domain-specific rules and then relearning all rule weights from data for the new domain. Existing approaches, such as belief-net backbones [2], may also be more concisely expressed using the FOL formula notation of SRL learner models.

[1] Markov Logic Networks (MLNs) will be used as a representative for all SRL approaches until differences can be discussed in more detail in the Alternatives section.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 516–518, 2011.

3 Lifted User Models

Currently many user models for ITS, e.g., overlay and BN models, are propositional, that is, they embed domain objects in rules or probabilistic graphs and their structure and / or weights are tailored for a specific domain. Thus, they cannot be easily reused in new domains. SRL user models, in contrast, allow expression of more general theories of user modeling in rules that apply across domains. These rules can then be incorporated with domain-specific predicates, relations, rules, and constants. Similarly, constraint-based [3] and logic-based user models are not inherently propositional. Still, they express domain-specific rules or constraints, are less expressive than FOL, and less tolerant or incapable of handling noise or probability.

4 Statistical Relational Learning for more Expressive User Models

FOL is used in SRL learner models, not just Horn clauses, and logical formulas can be softened to allow exceptions with the use of weights. For example, the formulas:

```
prereq(+sk1,+sk2) ^ knows(+sk1) => knows(+sk2)
prereq(+sk1,+sk2) ^ knows(+sk2) => knows(+sk1)
```

express logical rules that obviously have exceptions. The first rule asserts that knowing a prerequisite to a skill implies knowing the skill. The second rule asserts that knowing the skill itself implies knowing its prerequisites. SRL software such as Alchemy [4] learns appropriate weights for such rules from user data. The weights vary for each prerequisite and skill.

The two rules above create a feedback loop not allowed in a causal network (a kind of BN). Practically this has important ramifications. We can now express new kinds of learner models, such as models that reason from affect to cognition, and from cognition to affect, essentially in both directions at the same time.

5 User Models That Leverage Social Network Analysis

Social network analysis expresses social relationships such as friendship, peer influence, command structure, and proximity as links in a graph. SRL is well-suited for collective classification and link analysis using objects and links in such a graph [5]. SRL models appear promising for application in user modeling for collaborative teaching applications as well as for jointly modeling teams and team members in team training applications. E.g., consider this rule in a Markov Logic Network user model:

```
friends(x,y) ^ inteam(x,t) ^ inteam(y,t) =>
                              grade(x,t) = grade(y,t)
```

It says that two friends in the same collaborative team are likely to receive the same grade on individual tests. Such an assertion may or may not be typically true, but nevertheless, it is easily expressed in the SRL formalism.

6 Alternatives

SRL is a rapidly expanding field undergoing rapid change. Alternatives to MLNs include lifted approaches to Bayesian Networks, such as Stochastic Logic Programs [6]. Most SRL approaches represent joint probability distributions. An alternative is to optimize a function representing overall coherence of graph relationships and object types [7]; such an approach may be sufficient for user modeling.

7 Challenges

There are a plethora of SRL approaches and the software can be quite difficult to use. Alchemy [4], for example, has a variety of inference algorithms (Gibbs, MC-SAT, and belief propagation) for determining conditional probabilities and others for determining the most likely state (MAP, *maximum a posteriori*) given constraints and evidence (MaxWalkSAT). "Lazy" versions of algorithms are more memory efficient.

8 Summary

SRL models have the promise of a common language for user modeling and more expressive user models. Practically, they could provide better models for affective computing, team training, and collaborative instruction. Currently, they are challenging to use and there has been little application to user modeling. Further research is required to determine the advantages and disadvantages of SRL user models compared to existing approaches.

References

1. Domingos, P., Lowd, D.: Markov Logic: An Interface Layer for Artificial Intelligence. Morgan & Claypool, San Rafael (2009)
2. Reye, J.: Student Modeling based on Belief Networks. Int. Journal AI ED (2004)
3. Mitrovic, A.: An Intelligent SQL Tutor on the Web. Int. Journal AI ED (2003)
4. Alchemy - Open Source AI, `http://alchemy.cs.washington.edu/`
5. Getoor, L., Taskar, B.: Introduction to Statistical Relational Learning. The MIT Press, Cambridge (2007)
6. Muggleton, S., Pahlavi, N.: Stochastic Logic Programs: A Tutorial. In: [1]
7. Roth, D., Yih, W.: Global Inference for Entity and Relation Identification via a Linear Programming Formulation. In: [1]

Use of the DynaLearn Learning Environment by Naïve Student Modelers: Implications for Automated Support

Richard Noble[1] and Bert Bredeweg[2]

[1] University of Hull International Fisheries Institute, Hull, HU6 7RX, UK
R.A.Noble@hull.ac.uk
[2] University of Amsterdam, Informatics Institute, PO Box 94323,
1090 GH Amsterdam, The Netherlands
B.Bredeweg@uva.nl

Abstract. This paper shows that naïve students will require coaching to overcome the difficulties they face in identifying the important concepts to be modeled, and understanding the causal meta-vocabulary needed for conceptual models. The results of this study will be incorporated in the automated feedback components that are currently being implemented in the DynaLearn software.

Keywords: DynaLearn, Conceptual Models, Education, Coaching, Feedback.

1 Introduction

Conceptual modeling is a new and important approach that can enhance science education. DynaLearn [1] is a software tool that facilitates this, enabling students to build conceptual models based on qualitative reasoning [2]. However, conceptual modeling can be difficult, especially for novices. This paper presents the results of a study investigating the difficulties they face. The results will make an important contribution to the ongoing development of automated support for learning by modeling, in particular through tools such as ontology-based feedback [3] and interactive model diagnosis [4] in the DynaLearn software, facilitating the acquisition of conceptual knowledge.

2 Case Study Design

Eighteen post-graduate education students were given a typical biology exam question about osmosis. The question required them to describe, and explain causally, the reason for the observed phenomenon. After a lecture on the DynaLearn software the students were given one hour to build a conceptual model that could be used to explain the given phenomenon. Students used the *basic causal model* learning space of the software which required them to define models representing system structure using entities, quantities and configurations and to represent behavior using notions of positive and negative causal dependencies [1]. The students were given no guidance on what to include in their model other than the exam question and a text book diagram of osmosis. The models created by the students were analyzed for structure,

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 519–521, 2011.

complexity, completeness (occurrence of 12 pre-defined norm concepts) and for errors in implementation and scientific validity.

3 Results

The models built by the students were similar in complexity in terms of the number of key concepts represented. However, they generally represented less than half of the 12 pre-defined norm concepts. Few students exhibited errors in the differentiation of entities and quantities, generally adding 4-6 entities and 4-5 quantities correctly to their models. The greatest variability and largest source of error in the models were the lack or incorrect use of causal dependencies (Fig. 1). Only three of the 18 students fully implemented some of the concepts correctly with suitable causal dependencies.

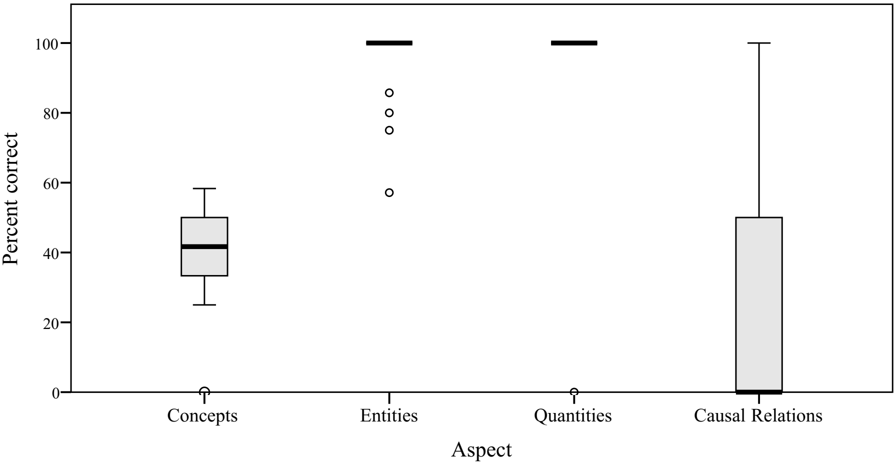


Fig. 1. Box plot of sources of errors in student conceptual models indicating the percentage of required concepts they implemented and the percentage of all entities, quantities and causal relations they added that were correct

4 Discussion

The early stages of learning by modeling carry an overhead both for students and for the teachers providing feedback and support. It was apparent from observation of the students, and their models, that they differed in ability and worked at vastly different rates. However, the difficulties they encountered were similar. Whilst the problems of different ability and work rates are not unique to learning by modeling, or the DynaLearn system, they do pose significant issues for the development of learning activities and lesson planning. However, since the difficulties the students encounter were similar, automated coaching strategies and learning activities can be designed to overcome them. It is apparent from this study that the students faced two main difficulties; identification of the important concepts and representing them causally within the software. Whilst the first difficulty is not unique to learning by modeling it is an important step in the evolution of a conceptual model [5]. Automated support for this problem requires that the software can identify superfluous or missing concepts in the models built by students and suggest additional content that will improve or complete the model. This can be achieved through ontology-based feedback using model comparison with a repository of norm teachers' models for identification of the correct concepts [3]. The second difficulty relates to the students' understanding of, and competency in, the meta-vocabulary required to form good causal explanations and model representations. The results indicate that the majority of naïve students did not understand the representation of causality in the tool. Therefore, coaching and support at this stage of modeling needs to focus on developing the meta-vocabulary for causal representations. This can be achieved using automated support (from model comparison) and interactive learning strategies (model-based behavior diagnosis) focusing on exploration of system behavior and causal representation of phenomena [4].

The DynaLearn project is developing automated tools for model-based diagnosis, ontology-based feedback and virtual-character-based dialogs that facilitate the types of support outlined above [3]. The results shown here inform the development of these tools and the teaching strategies that utilize them.

Acknowledgements. The author would like to thank the University of Hull Centre for Education Studies who took part in this case study. The work was funded by the DynaLearn FP7 Project no. 231526 (http://www.DynaLearn.eu).

References

1. Bredeweg, B., Liem, J., Beek, W., Salles, P., Linnebank, F.: Learning spaces as representational scaffolds for learning conceptual knowledge of system behaviour. In: Wolpers, M., Kirschner, P.A., Scheffel, M., Lindstaedt, S., Dimitrova, V. (eds.) EC-TEL 2010. LNCS, vol. 6383, pp. 46–61. Springer, Heidelberg (2010)
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. Gracia, J., Liem, J., Lozano, E., Corcho, O., Trna, M., Gómez-Pérez, A., Bredeweg, B.: Semantic techniques for enabling knowledge reuse in conceptual modelling. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 82–97. Springer, Heidelberg (2010)
4. Mehlmann, G., Häring, M., Bühling, R., Wißner, M., André, E.: Multiple agent roles in an adaptive virtual classroom environment. In: Safonova, A. (ed.) IVA 2010. LNCS, vol. 6356, pp. 250–256. Springer, Heidelberg (2010)
5. Bredeweg, B., Salles, P., Bouwer, A., Liem, J., Nuttle, T., Cioaca, E., Nakova, E., Noble, R., Caldas, A.L.R., Uzunov, Y., Varadinova, E., Zitek, A.: Towards a structured approach to building qualitative reasoning models and simulations. Ecological Informatics 3(1), 1–12 (2008)

Learning on Semantic Social Networks: A Distributed Description Logic-Based Approach

Mourad Ouziri and Salima Benbernou

LIPADE, Université Paris Descartes, 45 rue des Saints-Pères,
75270 Paris Cedex 06, France
{mourad.ouziri,salima.benbernou}@parisdescartes.fr

Abstract. The paper addresses the problem of searching relevant learning resources in social networks, shared between members, Therefore, a network of shared leaning resources is build up. We propose a distributed description logics based semantic model to share learning resources. Using the logics allows to infering additional interesting relatioships between learning resources that are not expressed by members. Moreover we provide a distributed reasoning algorithm that allows searching of all relevant learning resources in the social network.

Keywords: Social networks, representing learning resources, distributed description logics.

1 Introduction

The global information world enrolls many services under the Web 2.0 umbrella. The social networking services grow in an increasing way. They allow teenagers, collegians, and students to participating in online social networks (i.e. for instance Facebook, MySpace etc), consuming social media and creating media contents. A question rises, how can universities, colleges can react to such new social network world and exploiting the new applications to changing learning? How blogged education can be handled? We try in this paper to provide an answer to the above questions and discussing the learning activity in social network. The aim is to design a system, providing relevant learning resources to learners. So, the efficient description and representation of learning resources is needed. The Web provides a huge datasource in which efficiency of seach is reduced whereas social network is a small subset of the Web. Member relationships in social networks are real appreciation for information searching. They guide search process to relevant resources. In this paper, we present a semantic approach to get relevant learning resources from a given member in a social network. The approach features a model based on distributed description logics (DDL) theory. DDL is a peer-to-peer knowledge representation formalism that is suitable to represent connections between members's learning resources. DDL provides reasoning algorithms that search relevant resources starting from a member knowledge base and following connections.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 522–524, 2011.

2 Connecting Learning Resources in Semantic Social Networks

Searching relevant resources in semantic social networks is performed into two steps: (1) indexing learning resources using concepts (2) querying the indexed resources using cpncepts of indexation. Semantic indexing consists to annotate each resource using concepts of an ontology that describes the meaning of the resource content. Semantic indexing is represented in a learning knowldge base (KB).Then, the search of learning resources uses annotation concepts to express and process queries.

Let's illustrate the approach towards the motivating example depicted as follws:

- Learner *L1* is interested on *databases* and sorts his courses into basic courses and advanced courses that he annotates using the concepts *BasicCourse* and *AdvancedCourse* (respectively).
- Learner *L2* holds a repository of courses on databases and programming that he sorts and annotates using concepts *BeginningCourse*, *MediumCourse* and *ExpertCourse*. All courses have an attribute called *hasField* that gives field (*database* or *programming*) of the course.

2.1 Model of Learning Resources

Annotation of learning resources made by each learner is represented using description logics [1]. A description logics knowledge base is composed of two parts: TBox and ABox. TBox contains concept descriptions and ABox contains assertions (individuals assignation to concepts).

In the continuation of the above example,the knowldge base KB_1 of Learner *L1* is given as follows:

KB_1 = <$TBox_1$, $ABox_1$> where,
$TBox_1$ = *{1:BasicCourse □1 : Course , 1:AdvancedCourse □1 : Course }*
$ABox_1$ = *{1:BasicCourse (m1), 1:BasicCourse (m2), 1:AdvancedCourse(m3)}*
where *m1, m2,m3 are* learner *L1's* annotated courses.

2.2 Model of Inter-connections between Learning Resources

Semantic connection of learning resources is made by connecting concepts that describe them (Figure 1).

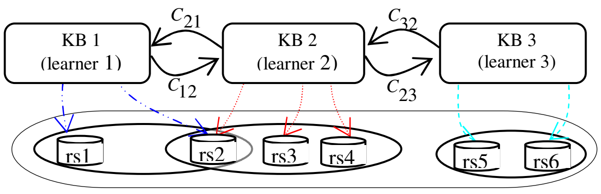


Fig. 1. Semantic connections between learner knowledge bases

We use distributed description logics (DDL) [2] to represent connections between concepts of different knoledge bases. In the continuation of the example:

- The learner L_1 considers that both beginning and medium courses of learner L_2 correspond are basic courses. So, the following connection axioms are added to L_1 knowldge base (C_{21}):
  1:BasicCourse □ 2:BeginningCourse   *1:BasicCourse □ 2:MediumCourse*

- The learner L1 is interested only on databases and not on programming courses of learner *L2*. So, the following connection axiom is added to *L1* knowldge base:
  1:Course □ 2:Course □ ∀2:hasField.DB

Given the connecting model, we are ready to discuss the reasoning mechanism that processes learner queries.

3 Distributed Reasoning Over Connected Member Ontologies

Learners express queries using their defined concepts. As example, the learner *L1* query *1:BasicCourse* returns all the basic courses, as defined in the knowledge base of the learner *L1*.

The reasoning algorithm processes this query by adding the resources annotated by the concept *2:BeginningCourse* as *1:BasicCourse □ 2:BeginningCourse*.

Insertion of connected concepts to the query result is done using propagation rules [1]. A propagation rule is defined for each operator. It consists to add an extension to the resolution system with respect to the semantics of rule (see [1] for more details).

The rules are applied following the follwing distributed algorithm:

```
DSat$_{KBi}$( i:C( x)) :
   (1) Initial constraint system is $S0 = \{x : i:C\}$
   (2) Apply the local propagation rules on $S0$, $Sat_{KBi}$( i: C( x))
   (3) For each constraint of the forms $j:D(x)$ or $j:R(x, y)$,
       where $D$ and $R$ are foreign concept and role (respectively)
       of the knowldge base $KBj$, then send the message $DSat_{KBj}$(
       j:D( x)) and $DSat_{KBj}$( j:R(x,y)) (respectively) to the $KBj$.
   (4) Add returned constraints to the system and apply local
       propagation rules.
```

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
2. Borgida, A., Serafini, L.: Distributed description logics: Assimilating information from peer sources. Jounal of Data Semantics 1, 153–184 (2003)

Generating Task-Specific Next-Step Hints Using Domain-Independent Structures

Luc Paquette, Jean-François Lebeau, Jean Pierre Mbungira, and André Mayers

Université de Sherbrooke, Québec, Canada
{Luc.paquette,Andre.Mayers}@USherbrooke.ca

Abstract. With the ASTUS framework, our aim is to facilitate the creation of tutors showing sophisticated pedagogical behaviors while keeping the authoring effort in line with that required by other well-known frameworks. ASTUS is thus based on knowledge structures that can be manipulated by the tutor's modules in order to, for example, generate task-specific next-step hints. An experiment suggests that the generated hints can be as effective and as well appreciated as those authored by a teacher.

1 Introduction

Creating an MTT from the ground up requires costly effort by highly trained individuals with knowledge of AI programming, cognitive psychology and the tutored task. Indeed, it has been estimated that an hour of educational material can take up to 300 hours of preparation [1]. The use of an authoring framework reduces the level of specialized expertise and the time required to create a tutor.

We created ASTUS [2], a model tracing tutor (MTT) authoring framework [3], to allow the reuse of sophisticated domain-independent pedagogical behaviors that are adapted using task-specific content [4]. Our motivation is to produce sophisticated behaviors with comparable effort to that required by a framework such as the Cognitive Tutors [5]. In this paper, we present empirical data suggesting that framework-generated hints can be as appreciated as teacher-authored hints and that the learning gains they produce are comparable.

2 Hint Generation

A hint is characterized by its structure and its content. While the same structure can be used to tutor multiple tasks, the content is specific to a task. Our hypothesis is that, given an adequate task model, a framework can automate the generation of efficient hints by using predefined structures and extracting the content from the task model.

ASTUS's knowledge components are defined in a way that facilitates their manipulation by the tutor's modules (see [2] for a detailed description of ASTUS's knowledge representation system). This allows the framework to extract their task-specific content and incorporate it into the corresponding generic hint structures.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 525–527, 2011.

Table 1. Examples of framework-generated hints (task-specific content is italicized)

	Sequence procedure	While procedure	Conditional procedure
Hint	In order to *separate the bit field* you need to: 1) *separate the sign field* 2) *separate the exponent field*	In order to *convert the binary number from right to left* you need to: *convert the next position* while there exists a *position not converted.*	You need to *find the mantissa with a period* since *result_value* is a *scientific binary number with a period.*

For example, in order to generate next-step hints, ASTUS uses domain-independent hint structures (message templates) associated to different types of procedures (a given type of knowledge component). The templates are designed to mirror the internal scripts used by each type of procedure. These scripts produce a set of subgoals (intentions) according to a specific algorithm (sequence, iteration and selection). The task-specific content of the hints is provided by "human readable names" assigned by the author to each relevant knowledge components. Table 1 gives examples of hint templates, instantiated using our float conversion tutor.

3 Results

We recruited all students (N=38) from a computer science course entitled "System Programming" at the Université de Sherbrooke. They used our tutors to solve floating-point number conversion problems. We randomly assigned them to one of two conditions: framework-generated next-step hints (FH) and teacher-authored next-step hints (TH). Of the 38 students, 15 completed the study for the FH condition, 19 for the TH condition and 4 didn't participate. A pretest and a posttest were used to evaluate the tutors' learning gains and an appreciation survey was used to evaluate the learners' opinion of the next-step hints.

Table 2. Results of the statistical analyses

	Stat	*P*	Effect size	Power
Pretest scores	$t(32) = -1.258$	0.218	$d = 0.43$	22.67%
Learning gain (FH)	$t(14) = -3.485$	0.004**	$d = 0.79$	89.65%
Learning gain (TH)	$t(18) = -4.926$	< 0.001***	$d = 0.86$	97.49%
ANCOVA	$F(1, 31) = 0.234$	0.632	$\eta^2_p = 0.0075$	7.56%
Hint appreciation	$t(28) = 0.358$	0.723	D = 0.13	6.40%

No statistically significant difference was observed between the pretest scores for the two conditions, but the medium effect size could indicate a notable difference. If this is the case, a statistically significant result could have been found using a more powerful test. The learning gains between the pretests and posttests were statistically significant for both conditions. An ANCOVA, with the pretest scores as the covariate, did not show a significant difference between the posttest scores. The very small effect size of this test seems to indicate that there is no actual difference even though a more powerful test may have found that this effect size is statistically significant. No significant difference was found between the participants' appreciation of the hints they received. Table 2 summarizes the statistical analyses.

The participants were also asked to give their comparative appreciation for four pairs of hints (framework vs. teacher). Table 3 presents a summary of those analyses.

Table 3. Results for the comparison of hint appreciation

	Stat	*p*	Effect size	Preferred	Power
While	$t(31) = 3.913$	< 0.001***	$d = 0.69$	teacher	96.56%
Conditional	$t(29) = 2.904$	0.007**	$d = 0.53$	generated	80.11%
Sequence	$t(30) = 11.998$	< 0.001***	$d = 2.16$	generated	100.00%
Sequence with inferences	$t(30) = -1.147$	0.260	$d = 0.21$	neither	19.89%

4 Conclusion

We showed that ASTUS can reduce the modeling costs of next-step hints thanks to a knowledge representation system that enables their generation. Empirical data suggests that those hints can be as appreciated and can produce comparable learning gains to those authored by a teacher. Further improvements would be to standardize the format of the generated hints using an explicit pedagogical theory and improving the readability of the hints using natural language techniques.

References

1. Murray, T.: An Overview of Intelligent Tutoring System Authoring Tools: Updated Analysis of the State of the Art. In: Murray, T., Blessing, S., Ainsworth, S. (eds.) Authoring Tools for Advanced Technology Learning Environments, pp. 491–544. Kluwer Academic Publishers, Dordrecht (2003)
2. Paquette, L., Lebeau, J.-F., Mayers, A.: Authoring Problem-Solving Tutors: A Comparison Between ASTUS and CTAT. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) Advances in Intelligent Tutoring Systems, pp. 377–405. Springer, Heidelberg (2010)
3. Lebeau, J.-F., Paquette, L., Fortin, M., Mayers, A.: An authoring language as a key to usability in a problem-solving ITS framework. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 236–238. Springer, Heidelberg (2010)
4. Paquette, L., Lebeau, J.-F., Mayers, A.: Integrating Sophisticated Domain-Independent Pedagogical Behaviors in an ITS Framework. In: Aleven, V., Kay, J., Mostow, J. (eds.) ITS 2010. LNCS, vol. 6095, pp. 248–250. Springer, Heidelberg (2010)
5. Aleven, V.: Rule-Based Cognitive Modeling for Intelligent Systems. In: Nkambou, R., Bourdeau, J., Mizoguchi, R. (eds.) Advances in Intelligent Tutoring Systems, pp. 33–62. Springer, Heidelberg (2010)

Causal Modeling of User Data from a Math Learning Environment with Game-Like Elements

Dovan Rai and Joseph E. Beck

Computer Science Department, Worcester Polytechnic Institute
{dovan,josephbeck}@wpi.edu

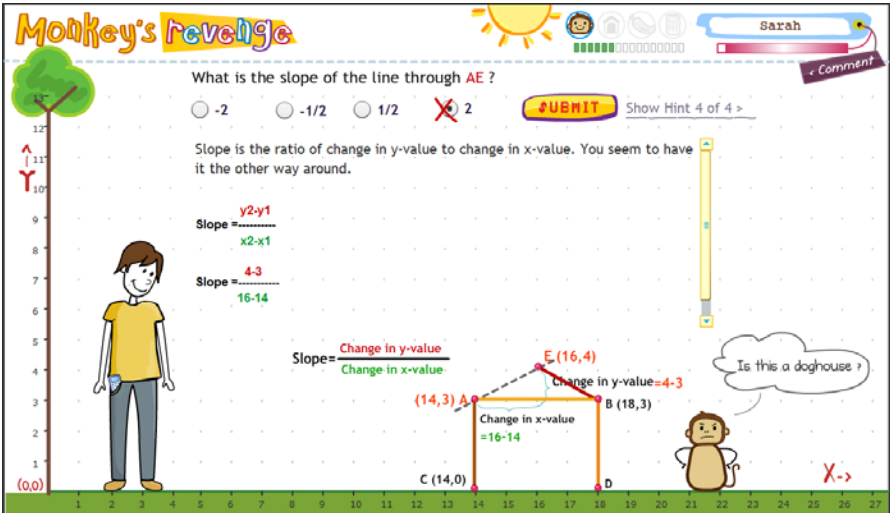
Abstract. We have created a math learning environment with game-like elements such as narrative, visual feedback, personalization, collection, etc. We made a study with four different versions of the tutor with different degree of 'game-like' and found that students preferred more 'game-like' tutor but we were not able to detect any conclusive difference in learning. Based on the data we collected through survey, logs and tests, we also built a causal model to understand the interrelationships between different student and tutor variables.

1 Introduction

Although educational games intend to make learning more enjoyable, they may add additional cognitive load among learners and have been empirically shown to generally be less effective than intelligent tutors in terms of learning gains [1]. Hence, instead of completely integrating educational content into a game framework, we choose to build a tutor integrating game-like elements, elements of games that are responsible for their engaging nature. We have developed *Monkey's Revenge*, a coordinate geometry tutor, which consists of a sequence of coordinate geometry problems wrapped in a visual cover story. We integrated game-like elements such as narrative, immediate visual feedback, personalization, and collection. Students can request hints and get bug messages as they stumble on misconceptions. Our aim is to iteratively assess each game-like element in terms of its engaging nature and impact on learning so that we can find an optimal balance of engagement and learning. To make comparative analysis of the game-like elements, we created four different versions of *Monkey's Revenge* with different degree of "game-like" (Table 1). A total of 297 middle school (12-14 year olds) students from four Northeastern schools in the United States participated in this study. The students were randomly assigned to the experimental conditions and we collected their survey data along with tutor logs. The students also did an 8-item pre- and post-test. We found that students who had a more "game-like" tutor reported liking the tutor more, but we found no conclusive difference in learning gain. But beyond confirming the main effect of the intervention, we are also interested in making exploratory analysis of the user data and have used causal modeling [2] approach using a software, TETRAD [3]. A causal model makes the additional assumption that the links between nodes represent causal influence.

We used factor analysis to reduce 16 survey questions into six variables. We also specified the causal hierarchy among the variables in the form of knowledge tiers [3] as specified in Table 2; variables in a lower tier cannot affect variables in higher tiers.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 528–530, 2011.



What is the slope of the line through AE ?

○ -2 ○ -1/2 ○ 1/2 ● 2

Slope is the ratio of change in y-value to change in x-value. You seem to have it the other way around.

Fig. 1. Screenshot of *Monkey's Revenge*

Table 1. Students' data across experimental conditions (means and 95% CI))

Tutor	Like tutor (max 5)	Learning gain (max 10)
Monkey's Revenge	3.9+/-0.3	0.41+/-0.6
Monkey's Revenge without visual feedback	3.8+/-0.3	0.88+/-0.6
Monkey's Revenge without narrative	3.6+/-0.3	0.31+/-0.6
Basic tutor with hints and bug message	2.8+/-0.3	0.45+/-0.6

Table 2. Student and tutor variables in causal model

Gender –tier 1	
Students' attitude and preference (survey)-tier 2	***likeMath*** (math is interesting); ***mathSelfConcept*** (I am afraid of Math.; I am afraid of doing word problems.); ***pedagogicalpreference*** (I like learning from computers, I like real world examples)
Prior knowledge –tier 3	***preTestScore*** (students' score on pre test)
Tutor activities (tutor logs) –tier 4	***%correct*** (ratio of correct problems); ***avgAttemptTime*** (average time student spent on each attempt); ***avgHints*** (average number of hints students asked on each question)
Opinion on tutor (survey)-tier 4	***tutorHelpful; tutorConfusing; likeTutor*** (This tutor looks interesting. I liked this tutor. I will recommend to a friend. This is better than other computer math programs I have)
Learning (test) –tier 5	***prePostGain*** (students' gain score from pre to post test score)

2 Interpretation of Model: Causal Claims and Causal Inference

We made a randomized controlled trial on the tutor's degree of "game-like". Other than this variable, the inferences we are making from our causal models are solely based on statistical independencies within data, domain knowledge we added, and causal assumptions of TETRAD's inference algorithm. We are interested to see how different student subpopulations receive the tutor intervention and how their characteristics are related to their tutor activities and overall performance and gain.

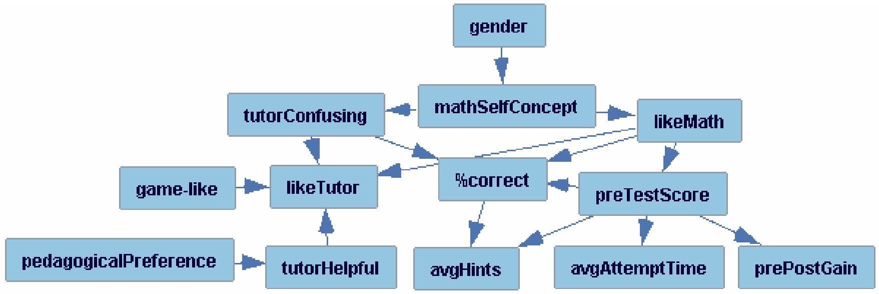


Fig. 2. Causal model created using TETRAD (p<0.05)

Gender: Female students have poorer self concept in math which makes math less enjoyable to them.

Math attitude: Students who like math have higher prior knowledge (*preTestScore*) and higher performance (*%correct*). Students who have poor self concept in math found the tutor confusing and that also influenced their performance

Pedagogical preference: Students who had preference for computer and real world examples found the tutor helpful which made them like the tutor more.

Engagement, performance and learning: We did not find any support for student engagement leading to better learning (*prePostGain*). However, we found that the students who like math have better performance (*%correct*) irrespective of their pre test (direct link *likeMath*→*%correct*). While it is possible that pre test did not capture all variance in prior knowledge, an alternative hypothesis for this indirect effect is that is that the students who like math are more engaged and perform better.

Since the causal model is limited to observed variables and causality is underdetermined by correlation, we cannot necessarily make causal claims based solely on the model. However, the causal inferences made by the model have generated some interesting hypotheses that we would like to further investigate.

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. Pearl, J.: Causality. Cambridge University Press, Cambridge (2000)
3. Sprites, P., Glymour, C., Scheines, R.: Causation, Prediction, and Search. MIT Press, Cambridge (2001)

Facilitating Communication with Open Learner Models: A Semiotic Engineering Perspective

Peter Reimann[1] and Susan Bull[2]

[1] MTO Psychologische Forschung und Beratung GmbH, Germany
peter.reimann@mto.de
[2] Electronic, Electrical and Computer Engineering, University of Birmingham, UK
s.bull@bham.ac.uk

Abstract. We introduce the notion of open learner models as artifacts and resources, situated in school level to illustrate a context of multiple user types. We suggest a new direction for research, focussing on open learner models as a facilitator of communication from a semiotic engineering perspective.

Keywords: Open learner models, communication, multiple users.

1 Introduction

Open learner models (OLM) are learner models that are accessible by users. We here consider the multiple users of OLMs at school level. We envisage that communication around OLMs will not only be of the 'classic' computer-user type, but will increasingly involve the OLM as a focus of communication between students, teachers, parents and peers. The role of an OLM may be primarily: (i) a communication artefact; (ii) a resource. We use the term *artefact* because an OLM is a representation that stands for itself, and can be read and interpreted, like a report. We refer to it as a *resource* because an OLM will provide information for the learner and, moreover, it may also play a part in communication amongst users such as interactions between student and teacher, and teacher and parents. In such discussions the OLM will be referred to, and will hence become a resource that participants in the communication will draw upon. We consider this distinction helpful as OLMs become more widely used in schools for purposes of formative assessment, capturing a potentially wide range of data on learning, and being used in a variety of communication contexts.

A pivotal requirement is that an OLM must be understandable by humans, at least in its main aspects [1]. In the case of OLMs for a range of user types (as considered here), additional issues need to be addressed. For example, students are likely to have different needs for an OLM than their teachers. Teachers will likely be better able to understand pedagogical issues displayed in a learner model than parents. Therefore OLM presentations that are suitable for all users, according to their purpose of use (which will often include communication and/or collaboration), need to be considered. The challenges for OLMs to support communication amongst users are therefore substantial. The following section gives an overview of OLM communication in school contexts. Semiotic engineering is then suggested as a basis for future work.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 531–533, 2011.

2 Open Learner Models for Communication in School Settings

Education professionals such as teachers, head teachers and school inspectors have recognised the potential for OLMs to support formative assessment and promote learner reflection [2]. OLMs can also be used by teachers and parents [1]. OLMs may be presented to the user in a variety of ways (e.g. text, concept map, tree structures, simple or complex graphical representations, animations, audio), according to user preferences and purpose of use [1]. Crucial questions to address therefore include: (a) To what extent should or may artifacts and resources be presented and/or explained to different users in different ways, and in what circumstances? (b) To what extent does the above point depend on the combination of discussants (e.g. individual, peer-peer, student-teacher, student-parent, parent-teacher)? (c) To what extent do the above depend on the specific information displayed? (d) To what extent do the above depend on the purpose of viewing the model? We distinguish common communication contexts in Table 1. The cells indicate some of the more likely (but not exclusive) interaction types (or purposes). The row headers indicate the person seeking interaction; the column headers refer to the person sought.

Table 1. Communication contexts using open learner models

	With Self	*With Student(s)*	*With Teacher*	*With Parent*
Student	awareness, plan, reflect, formative assessment	collaboration, peer help, competition, confirmation	learning support	learning support
Teacher	assessment, plan individual or group support	support, set up groups, inform on-the-spot decisions	team teaching, teacher training, quality assurance	interpret or explain OLM, plan learning support
Parent	evaluate, plan learning support	learning support	question, seek clarification	-

3 The Requirement for Continuous Semiosis

From a communication perspective, OLMs (in the school context) are sign systems that mediate communication between designers, teachers, students, and other stakeholders (e.g. parents, education system managers, potential employers). Taking such a semiotic perspective (adopting Peirce's [3] notion of signs as *anything that stands for something else, to somebody, in some respect or capacity*) directs our attention to the fact that OLMs are dependent on interpretation and that they, like any sign system, may have very different valid meanings, that their mutual intelligibility widely depends on cultural conventions and mechanisms to negotiate shared meanings, and that ultimately they have no fixed meaning [4]. Hence, the interpretation of information in an OLM needs to be open to continuous re-interpretation. In the case of school-oriented OLMs, effective semiosis between the OLM designer and the teacher is of particular importance because often the information in the OLM will be communicated to the student mediated by the teacher. Opportunities for "misinterpretations" (divergent semiosis) hence get multiplied. It is therefore an important requirement for school-based OLMs that effective methods are used to communicate the designer's intentions: a challenge for which semiotic engineering aspires to provide solutions.

In this framework [4], software designers use the interface to communicate their design intention to users. Design intention refers to the purpose of a software artifact--the kind of tasks the tool or representation is designed to deal with--and to the general approach to these tasks. Since the designer is (usually) not present in communication with the end user, semiotic engineering introduces the notion of "designer's deputy": a communicating agent that can tell the designer's message. The deputy is, hence, an interface agent engaging with the user in a meta-communication about what the computational artifact can do. The first generation of such meta-agents (the most notorious example being the animated Microsoft Office assistants) has had mixed success.

In our research on OLMs in the context of the Next-TELL project (www.next-tell.eu), we are currently working on extending the semiotic approach in order to capitalise on the affordances of web-based applications in general and web-based OLMs in particular. The project will develop an approach that treats the OLM as a human- and machine-usable web service (or a bundle of such services), thereby allowing communication with the end-user in a much more open and dynamic manner than is the case for desktop-oriented applications. For instance, we foresee incorporating the interpretations users give to elements of an OLM into the OLM itself (e.g. as annotations). The value is perhaps most obvious in cases where the teacher provides information to a learner model, which may help a learner to interpret this information as intended. Communication about their respective learner models amongst peers could also benefit highly from annotations, as learners may interpret the models in different ways. Finally, teachers are likely to gain more of an insight into a child's learning outside the classroom, if parents can also provide annotations. As a result of allowing user annotations in the OLM, we may come some way towards supporting the kind of communication contexts referred to in Table 1.

Acknowledgements

This project is supported by the European Community (EC) under the Information Society Technologies (IST) priority of the 7th Framework Programme for R&D under contract number 258114 NEXT-TELL. This document does not represent the opinion of the EC, and the EC is not responsible for any use that might be made of its content.

References

1. Bull, S., Kay, J.: Open Learner Models. In: Nkambou, R., Bordeau, J., Miziguchi, R. (eds.) Advances in Intelligent Tutoring Systems, pp. 318–338. Springer, Heidelberg (2010)
2. Kerly, A., Bull, S.: Open Learner Models: Opinions of School Education Professionals. In: Koedinger, K., Luckin, R., Greer, J. (eds.) AIED 2007, pp. 587–589. IOS Press, Amsterdam (2007)
3. Peirce, C.S.: Collected papers of Charles Sanders Peirce, vol. 1-8, Harthshorne, C., Weiss, P. (eds.). Harvard University Press, Cambridge (1931-1958)
4. de Souza, C.S.: The Semiotic Engineering of Human-Computer Interaction. MIT Press, Cambridge, MA (2005)

When Off-Task is On-Task: The Affective Role of Off-Task Behavior in Narrative-Centered Learning Environments

Jennifer Sabourin, Jonathan P. Rowe,
Bradford W. Mott, and James C. Lester

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

Abstract. Off-task behavior is the subject of increasing interest in the AI in Education community. This paper reports on an investigation of the role of off-task behavior in narrative-centered learning environments by examining its interactions with student learning gains and affect. Results from an empirical study of students interacting with the CRYSTAL ISLAND environment indicate that off-task behavior generally has negative impacts on learning. However, further analyses of students' affective transitions suggest that some students may be using off-task behavior as a strategy to regulate negative emotions.

Keywords: Narrative-centered learning environments, off-task behavior, affect.

1 Introduction

Narrative-centered learning environments contextualize problem solving in interactive story scenarios. While narrative-centered learning environments present significant opportunities for enhancing engagement, they may also invite behaviors that are not learning oriented. Concerns about off-task behavior are reinforced by recent findings, which indicate that going off-task is detrimental to learning [1]. There is also evidence that off-task behavior may be associated with students' emotional states, such as *boredom* and *frustration* [2]. However, off-task behavior may play an important productive role in educational settings. Rather than serving as an unproductive diversion, off-task behavior could offer a means for students to take a needed "break" from complex or challenging learning activities. In this manner, off-task behavior may function as an emotion regulation mechanism that students use to renew their motivation to participate in productive learning activities.

The work presented in this paper investigates the impact and affective role of off-task behavior in narrative-centered learning environments. It extends previous work that characterized the relationship between off-task behavior and learning in the CRYSTAL ISLAND learning environment [3]. Data from emotion self-reports collected during a study with CRYSTAL ISLAND is used to investigate relationships between students' moods, affect transitions, and off-task behaviors.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 534–536, 2011.

2 Investigating Off-Task Behavior in CRYSTAL ISLAND

Our work on off-task behavior is situated in CRYSTAL ISLAND, a narrative-centered learning environment [3]. Several in-game actions are identified as off-task, including: (1) interactions with in-game objects that re not relevant to the illness scenario, (2) moving a task-related object to an unrelated location, (3) spending too much time in a location irrelevant to the task, or (4) exceeding a height achievable by normal navigation (e.g., climbing on top of trees or boxes). Intervals of time in which several off-task behaviors occur in succession are aggregated and considered as a single duration of off-task behavior. No actions from the first five minutes of game play are designated as off-task, in order to provide ample exploration time.

In order to investigate the role of off-task behavior in narrative-centered learning environments, data from 260 eighth grade students from a rural North Carolina middle school was used. During the week prior to the study, students completed a curriculum test involving 19 microbiology questions. Students interacted with CRYSTAL ISLAND until they solved the mystery or 55 minutes of interaction elapsed. Afterward, students completed the same curriculum test used in the pre-survey.

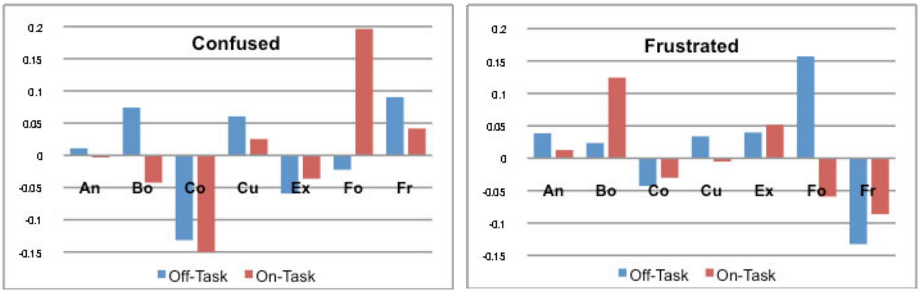
Students' affect data was collected during the learning interactions through regular self-report prompts. Students were prompted every seven minutes to self-report their current mood and "status" through an in-game smartphone device. Students selected one emotion from a set of seven options, which included: *anxious*, *bored*, *confused*, *curious*, *excited*, *focused*, and *frustrated*.

An investigation of student learning indicated that on average students answered 2.11 (*SD* = 3.25) more questions correctly on the post-test than they did on the pre-test, which was statistically significant, $t(259) = 10.46$, $p < 0.0001$. Students spent approximately 4.58% (*SD* = 6.82) of their time off task, with a range of 0% to 63.2%.

A previous investigation using an earlier version of CRYSTAL ISLAND found that students' overall learning gains were not affected by the frequency of off-task behaviors [3]. However, the current data reveals a negative correlation between off-task behavior and normalized learning gains, $r(258) = -0.18$, $p = 0.004$. These findings indicate that off-task behavior may be more harmful to learning in CRYSTAL ISLAND than previously believed.

An analysis of transition likelihoods, L, between emotional states and off-task behaviors was conducted [4]. The analysis indicated that no emotional states were more likely than chance to lead to off-task behavior, with $\alpha = 0.05$. Next, similar analyses were conducted that compared intervals where off-task behaviors occurred between emotion self-reports and intervals where students remained on task. In particular, the transitions originating from *confusion* and *frustration* revealed differences in how students transition to new emotions depending on their off-task behavior (Figure 1). Students who remained on-task after reporting *confusion* were likely to next report feeling *focused*. Alternatively, students who went off-task after reporting *confusion* were likely to report *boredom* or *frustration* next.

While off-task behavior indicated negative emotional transitions for students who were *confused*, the opposite was true for students experiencing *frustration*. *Frustrated* students who went off-task were more likely to report being *focused* at the next report, which suggests that these students may have used off-task behavior to temporarily distance themselves from problem solving. Alternatively, *frustrated* students who



(a)

(b)

Fig. 1. Average likelihoods for transitioning from a) *confused* and off/on-task to a particular emotion, and b) *frustrated* and off/on-task to a particular emotion

remained on-task were likely to report *boredom* at the next report. These students may have remained on-task even when it would have been beneficial to take a break.

These findings provide insight into how narrative-centered learning environments might best respond to off-task behavior. It appears that while in a state of *confusion*, students should be encouraged to continue working on the task and not be distracted by extraneous elements of the environment. Alternatively, once this *confusion* has reached the point of *frustration*, students should not only be permitted, but perhaps encouraged to explore non-learning aspects of the environment as a short reprieve.

Acknowledgments. The authors wish to thank members of the IntelliMedia Group for their assistance, Omer Sturlovich and Pavel Turzo for use of their 3D model libraries, and Valve Software for access to the Source^{TM} engine and SDK. This research was supported by the National Science Foundation under Grants REC-0632450, DRL-0822200, and IIS-0812291. This material is based upon work supported under a National Science Foundation Graduate Research Fellowship.

References

1. 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: Proc. of the ACM Conf. on Computer-Human Interaction, pp. 383–390 (2004)
2. Baker, R.S., D'Mello, S.K., Rodrigo, M.M.T., Graesser, A.C.: Better to Be Frustrated than Bored: The Incidence, Persistence, and Impact of Learners' Cognitive-Affective States during Interactions with Three Different Computer-Based Learning Environments. Intl. Journal of Human-Computer Studies. 68(4), 223–241 (2010)
3. Rowe. J., McQuiggan, S., Robison, J., Lester, J.: Off-Task Behavior in Narrative-Centered Learning Environments. In: Proc. of the 14th Intl. Conf. on AI in Education, pp. 99–106 (2009)
4. D'Mello, S., Taylor, R.S., Graesser, A.: Monitoring Affective Trajectories During Complex Learning. In: Proc. of the 29th Annual Meeting of the Cognitive Science Society, pp. 203–208 (2007)

Evaluating the Use of Qualitative Reasoning Models in Scientific Education of Deaf Students

Paulo Salles, Mônica M.R. Pereira, Gisele M. Feltrini,
Lilian Pires, and Heloisa Lima-Salles

Institute of Biological Sciences, University of Brasilia
Campus Darcy Ribeiro, Asa Norte, Brasilia – DF. 70.910-900, Brazil
{psalles,hsalles}@unb.br, monicamresende@terra.com.br,
{gisele_morisson,liliancpires}@yahoo.com.br

Abstract. The present work describes a case study on evaluation of qualitative reasoning (QR) models as a tool for the acquisition of scientific concepts, the improvement of linguistic skills and the development of inferential reasoning of deaf students. Specific didactic material was developed in a DVD and presented to explain in LIBRAS (Brazilian Sign Language) and in Portuguese how to build and to use qualitative models to deaf secondary school students. After using the material, their performance was compared to a control group. The experimental group showed significant improvement in tests exploring environmental science concepts; their ability to follow long chains of causal relations and to make inferences also improved, as shown in written texts. Besides the learning results, the paper also contributes for the discussion about methodological aspects regarding the preparation of didactic material based on QR models and how to bring it into the classroom.

Keywords: deaf, qualitative models, science education.

1 Introduction

The Brazilian educational system is nowadays faced with the task of promoting the education of deaf students along with hearing students in inclusive classrooms. Understanding the requirements is a condition for the inclusion of the deaf to be successful. Previous work [2; 4] has shown that qualitative reasoning (QR) models [6] are powerful tools for the education of deaf students. This paper aims at evaluating learning with QR-based didactic material. The material was developed after a series of activities involving secondary school teachers and a group of deaf students who developed and validated specific signs to express modeling primitives used to build and to simulate qualitative models [1; 3]. The didactic material (assignments, a bilingual glossary, motivation texts and two models) was compiled into a DVD. Models and modeling primitives are explained both in LIBRAS and in (spoken) Portuguese, the latter also presented as a written version [1]. In order to evaluate both QR models and the didactic material as learning tools, the acquisition of scientific concepts and vocabulary, and the development of logical reasoning skills in written texts (in Portuguese as a second language) of deaf students were investigated [3]. Accordingly, this study seeks to answer the question: *Are qualitative models effective*

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 537–540, 2011.

to improve the deaf students' ability to acquire scientific concepts and to improve their deductive reasoning as well as their linguistic skills, expressed in written Portuguese texts?

2 Methodological Aspects of Evaluation

Secondary school deaf students aged between 15-18 years from three public schools in the Federal District were invited to participate of this study. Both the experimental group and the control group consisted of 30 students. The experimental group had eight meetings of circa 1h40min each. In the beginning the students answered a pre-test, including objective questions and a written essay about the theme "Algal bloom". In the following classes, the teacher used the DVD for exploring the model "Tree and Shade" to teach the qualitative modeling language. Next, the students were exposed to a more advanced model, the "Global warming" [1]. At the end of the course they had a lecture based on qualitative models and simulations about the algal bloom, and answered a post-test. The control group had two meetings of circa 1h40min each (similar to the experimental group). In the beginning of the first meeting, the students answered to the same pre-test. After that, the students had a expositive lecture about algal bloom, and by the end of the second lecture they answered the same post-test about this theme. The linguistic performance of the students in the essays was assessed in accordance to the Relevance Theory [5]. Relevant information is defined as information modifying and improving an overall representation of the world. When the information is relevant, the human deductive device yields only non-trivial conclusions. In contrast, trivial conclusions leave the content of the assumptions unchanged (except for the addition of arbitrary material).

Questions in pre and post tests of both experimental and control groups involving understand of concepts were measured by the number of correct answers. The number of total, non-trivial and trivial conclusions found in the essays was counted. Statistical analyses procedure was the following: data were tested for the normality by means of the Shapiro-Wilk test. If the data fit to a normal distribution, the Paired t-test was used, otherwise, the non parametric Wilcoxon signed rank test was used. Tests were run in R 2.12.0 (R Development Core Team, 2010), at the significance level of 5%.

3 Results

The evaluation results have shown that the experimental group presented significant improvement in conceptual understand after the use of qualitative models.

Table 1. Results of the pre and post tests exploring ecological concepts applied to students in the experimental (n= 30) and control (n= 30) groups of deaf students

	Student t	Wilcoxon	
Pre-Control X Pre-Experim.	t = 1,56 p = 0,12	v = 553 p = 0,13	NS
Pre-Control X Post-Control	t = -4,12 p < 0,01	v = 24 p < 0,01	HS
Pre-Experim.X Post-Experim.	t = -9,92 p < 0,001	v = 5 p < 0,001	HS
Post-Control X Post-Experim.	t = 8,34 p < 0,001	v = 832 p < 0,001	HS

NS= non significant at the level of 5%; HS = highly significant at the level of 5%

Although a significant difference between the pre and post-test results of the control group indicate a learning effect in the expositive lecture, comparison between the results of post-tests in both groups supports the conclusion that the use of qualitative models has produced better results (Table 1). The students' linguistic performance in written essays improved, as shown by the increase in the number of trivial conclusions and reduction in the number of non-trivial conclusions (Table 2).

Table 2. Results of statistical analyses comparing written texts by experimental (n= 26) and control (n= 26) groups of deaf students

Group DEAF	Inferences	Wilcoxon Test	Significance
Pre-Control X Pre-Experim.	Total of inferences	V= 272,5; p= 0,45	NS
	Non-trivial conclusions	V= 252,5; p= 0,81	NS
	Trivial conclusions	V= 261,5; p= 0,56	NS
Pre-Control X Post-Control	Total of inferences	V= 128,5; p= 0,06	NS
	Non-trivial conclusions	V= 129; p= 0,06	NS
	Trivial conclusions	V= 28; p= 0,39	NS
Pre-Experim. X Post-Experim.	Total of inferences	V= 69,5; p= 0,03	S
	Non-trivial conclusions	V= 51; p < 0,001	HS
	Trivial conclusions	V= 56; p= 0,19	NS
Post-Control X Post-Experim.	Total of inferences	V= 132; p < 0,001	HS
	Non-trivial conclusions	V=113,5; p < 0,001	HS
	Trivial conclusions	V= 299; p= 0,13	NS

NS= non significant at the level of 5%; S=significant; HS = highly significant

4 Discussion and Final Remarks

This paper shows that QR models may enhance learning scientific concepts and improve inferential reasoning and writing skills of deaf students. Besides that, the paper reports the development of a methodology – how to create bilingual didactic material, based on QR models and a visual pedagogy to develop scientific vocabulary and reasoning. The didactic material created for this work was positively evaluated by the students. QR models may become the basis for a community of practice of deaf and hearing students that learn scientific concepts while developing the written language.

Acknowledgements. We would like to thank the deaf students, teachers and interpreters of LIBRAS who participated in the study described here. We are also grateful to PROESP/ CAPES / MEC for the financial support to the project "Português como segunda língua na educação científica de surdos", and to DynaLearn, a project co-funded by the EC within FP7, grant number 231526.

References

[1] Feltrini, G.M.: Aplicação de Modelos Qualitativos na Elaboração de Materiais Didáticos para o Ensino de Ciências a Estudantes Surdos. MSc Dissertation. University of Brasília, Brazil (2009)

[2] Lima-Salles, H., Salles, P., Bredeweg, B.: Qualitative Reasoning in the Education of Deaf Students: scientific education and acquisition of Portuguese as a second language. In: Forbus, K. (ed.) Proceedings of the 18th International Workshop on Qualitative Reasoning, QR 2004 (2004)

[3] Resende, M.M.P.: Avaliação do uso de modelos qualitativos como instrumento didático no ensino de ciências a estudantes surdos e ouvintes. MSc Dissertation. University of Brasília, Brazil (2010)

[4] Salles, P., Lima-Salles, H., Bredeweg, B.: The Use of Qualitative Reasoning Models of Interactions Between Populations to Support Causal Reasoning of Deaf Students. In: Looi, C.-K., McCalla, G., Bredeweg, B., Breuker, J. (eds.) Artificial Intelligence and Education: Supporting Learning Through Intelligent and Socially Informed Technology, pp. 579–586. IOS Press/Omasha, Amsterdam (2005)

[5] Sperber, D., Wilson, D.: Relevance: Communication and Cognition. Blackwell Publishers Ltd., Oxford (1995)

[6] Weld, D., de Kleer, J. (eds.): Readings in Qualitative Reasoning about Physical Systems. Morgan Kaufmann, San Francisco (1990)

TORMES Methodology to Elicit Educational Oriented Recommendations

Olga C. Santos and Jesus G. Boticario

aDeNu Research Group, Artificial Intelligence Department, UNED,
Calle Juan del Rosal, 16, Madrid 28040, Spain
{ocsantos,jgb}@dia.uned.es
http://adenu.ia.uned.es

Abstract. The *TORMES* methodology is based on the ISO standard 9241-210 and aims to involve educators in the process of designing educationally oriented recommendations through user centred design methods and data mining analysis. In this paper, we focus on the iteration to elicit educational oriented recommendations.

Keywords: Educational recommender systems, user centred design, data mining analysis.

1 *TORMES* Methodology

The state of the art in recommender systems in education [1] shows that current approaches focus on recommending learning objects that have been contributed to complement the instructional design of the course, but they do not take into account the particularities of the educational domain when designing the recommendations to be offered. To cope with this gap, the *TORMES* methodology (i.e. Tutor-Oriented Recommendations Modelling for Educational Systems) supports the design of educationally oriented recommendations by involving educators in the process through user centred design (UCD) methods and data mining analysis. *TORMES* is based on the ISO standard 9241-210 [2], which provides guidance on human centred design activities throughout the life cycle of computer-based interactive systems. The UCD cycle outlines four essential activities: 1) understanding and specifying the context of use, 2) specifying the user requirements, 3) producing design solutions to meet user requirements, and 4) evaluating designs against requirements. Several usability methods can be used in the UCD cycle [3]. Moreover, *TORMES* integrates data mining analyses on past interaction data to i) identify troublesome or promising situations, ii) tune the design of the recommendations proposed by educators, and iii) adjust the recommendations' design after the course experience. Due to its iterative nature, *TORMES* can support several kinds of iterations. In this paper, we focus on the iteration to 'Elicit educational oriented recommendations'.

2 Eliciting Educational Oriented Recommendations

The goal of this iteration is to support educators in understanding the recommendation needs in their scenarios and design educational oriented

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 541–543, 2011.

recommendations for them. The input for this iteration is the available knowledge about the context of use (if any) coming from some previous iteration or from the design plan. The expected result is a set of validated and semantically modelled educational oriented recommendations that are ready to be delivered to the learners in a learning management systems (LMS) through a semantic educational recommender system (SERS) [4]. Figure 1 shows this iteration cycle representing the activities from the ISO 9241-210 standard (in bold) and the UCD methods suggested to be carried out in them (in brackets).

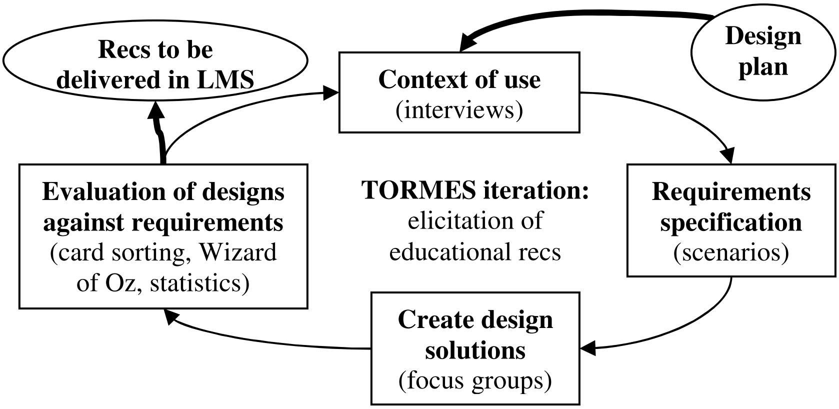

Recs to be delivered in LMS
Context of use (interviews)
Design plan
Evaluation of designs against requirements (card sorting, Wizard of Oz, statistics)
TORMES iteration: elicitation of educational recs
Requirements specification (scenarios)
Create design solutions (focus groups)

Fig. 1. *TORMES* iteration to 'Elicit educational oriented recommendations'

Very briefly, the four ISO 9241-201 activities have been particularised as follows:

- **Context of use:** the context of use can be enriched through *individual interviews* with educators who identify recommendations that, when delivered, provide inclusive educationally oriented adaptive navigation support to their learners. Data mining analysis from past courses supports the extraction of additional information that can complement the initial description of the context of use.
- **Requirements specification:** the *scenario based approach* [5] is used to extract knowledge from the educators on what the requirements are for the recommendations within the given context of use and identify an initial set of recommendations. The information mined in the previous activity is use here to adjust the applicability conditions of the recommendations proposed.
- **Create design solutions:** *focus groups* are used to involve several educators in validating the initial set of recommendations elicited from the scenarios in the previous activity. The goal is to revise the recommendations obtained in the solution scenario and come to an agreement.
- **Evaluation of designs against requirements:** educators and learners can evaluate the designed recommendations by rating their relevance and classifying them with a closed *card sorting* [6]. The running prototype can be a functional system or a Wizard of Oz. The results obtained are to be analysed with descriptive statistics.

3 Discussion

We have introduced *TORMES*, a methodology that integrates the UCD cycle (i.e. context of use, requirements specification, create design solutions and evaluation of designs against requirements) in supporting educators in identifying the recommendation needs in formal e-learning scenarios (i.e. those learning scenarios in which educators are involved). The outcomes from the UCD methods are complemented with findings coming from the application of data mining methods, which for instance, can identify situations where non-collaborative learners may receive a particular recommendation. Once educators identify the recommendations, they model them through a semantic recommendation model [4]. To this, they follow a rule-based approach, which makes use of semantic descriptions. Finally, recommendations are automatically delivered to learners by a SERS integrated via web services with existing LMS, e.g. dotLRN [7].

The benefits of the work presented in this paper lay on providing a methodology for educators to understand the recommendation needs in formal e-learning scenarios and to supports them in eliciting sound recommendations that address the changing educational needs as well as cognitive, meta-cognitive, social and affective issues required when learners interact with their courses delivered via an LMS.

Acknowledgements

This work is framed in the context of the projects carried out by the aDeNu group. In particular, EU4ALL (IST-2006-034778), ALTER-NATIVA (DCI-ALA/2010/88), A2UN@ (TIN-2008-06862-C04-00/TSI) and CISVI (TSI-2008-020301-2008-21).

References

1. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., Koper, R.: Recommender Systems in Technology Enhanced Learning. In: Kantor, P., Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook: A Complete Guide for Research Scientists and Practitioners (2010)
2. ISO Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems. ISO 9241-210:2010
3. Bevan, N.: UsabilityNet Methods for user centered design. In: Jacko, J., Stephanidis, C. (eds.) Human-Computer Interaction: Theory and Practice (Part 1), vol. 1, pp. 434–438. Lawrence Erlbaum, Heraklion (2003)
4. Santos, O.C., Boticario, J.G.: Modeling recommendations for the educational domain. In: Proceedings of the 1st Workshop 'Recommender Systems for Technology Enhanced Learning' (RecSysTEL 2010), pp. 2793–2800 (2010)
5. Rosson, M.B., Carroll, J.M.: Usability engineering: scenario-based development of human computer interaction. Morgan Kaufmann, San Francisco (2001)
6. Spencer, D.: Card Sorting. Designing Usable Categories. Rosenfeld Media (2009)
7. Santos, O.C., Granado, J., Raffenne, E., Boticario, J.G.: Offering Recommendations in OpenACS/dotLRN. In: 7th Int. Conf. on Community Based Environments (2008)

Will Structuring the Collaboration of Students Improve Their Argumentation?

Oliver Scheuer[1], Bruce M. McLaren[1,2], Maralee Harrell[2], and Armin Weinberger[1]

[1] Saarland University, Saarbrücken, Germany
[2] Carnegie Mellon University, Pittsburgh, PA U.S.A.
{o.scheuer,a.weinberger}@mx.uni-saarland.de,
bmclaren@cs.cmu.edu, mharrell@andrew.cmu.edu

Abstract. Learning to argue in a computer-mediated and structured fashion is investigated in this research. A study was conducted to compare dyads that were scripted in their computer-mediated collaboration with dyads that were not scripted. A process analysis of the chats of the dyads showed that the scripted experimental group used significantly more words and engaged in significantly more broadening and deepening of the discussion than the non-scripted control group.

Keywords: computer-supported collaborative learning, argumentation.

1 Introduction

Researchers have been increasingly more interested in studying how to use technology to help students learn argumentation skills [1]. This work follows, in particular, from others who have investigated the effect of scripts [2] on the learning of argumentation. We present initial results of our approach to engage student dyads in critical debate in a computer-mediated setting. Their task was to critically review argumentation texts on a controversial issue (global warming ethics) and to jointly take a reasoned position. Our main research question is: Will structured student collaboration lead to higher quality argumentation?

2 Research Design

Based on insights we obtained from the literature (e.g. [3, 4, 5]) we devised an instructional design founded on three principles: **(P1)** students should have time to form a personal opinion on a controversial issue before engaging in social interaction, **(P2)** better discussions and more learning can be expected when a conflict of opinions exists between students, and **(P3)** through instructional guidance, productive collaboration and discussion norms can be stimulated. Our hypothesis is that an intervention based on **P1**, **P2** and **P3** will lead to a higher quality of argumentative interaction (**H1**) and in turn to more learning (**H2**). This paper focuses on **H1** and a process analysis to evaluate it; an investigation of **H2** is deferred to future work.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 544–546, 2011.

We carried out a study in the context of an “Introduction to Philosophy” course at a U.S. university. Three sessions with required attendance were conducted. A quasi-experimental pretest-intervention-posttest design with two conditions was employed. The final data analysis is based on 8 control dyads and 11 experimental dyads.

Fig. 1 depicts the experimental procedure. The data was collected on Nov 19th and Dec 3rd, 2010. In preparation for the experimental sessions, students read two texts that advocate different policies with respect to global warming ethics (“drastic GHG reductions” versus “moderate GHG reductions plus smart policies to remedy other problems mankind is suffering from”). The task environment consisted of Google Documents (https://docs.google.com/) that contained instructions, input fields to answer essay questions, and a chat tool.

The control group worked collaboratively and in a self-organized manner on both days (unscripted collaboration). On Nov 19th students were asked to paraphrase the arguments from both texts (Q1 and Q2), and to decide jointly which argument was more compelling (Q3). They were allowed (and encouraged) to consult the two source texts. On Dec 3rd students were asked to argue for and justify the text they considered to be more compelling, without access to the source texts. Instead, they received their answers from the Nov. 19th session. We expected livelier discussion when students use their own interpretations rather than skimming through the source texts again.

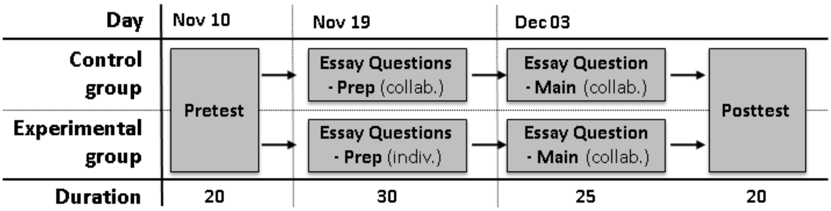

Day
Nov 10
Nov 19
Dec 03
Control group
Experimental group
Pretest
Essay Questions - Prep (collab.)
Essay Question - Main (collab.)
Essay Questions - Prep (indiv.)
Essay Question - Main (collab.)
Posttest
Duration
20
30
25
20

Fig. 1. Experimental procedure

The experimental group differed from the control group in several respects. On Nov 19 they worked individually (**P1**). To increase the chances of creating different preferences we used two slightly different versions of the essay questions Q1 and Q2, which were biased towards one of the two positions (reproduce one argument and rebut the other argument). Analogous to the control group (yet individually), students decided on the argument they preferred (Q3). On Dec 03 students with different preference were paired up (**P2**). Collaboration was scripted in this session through a set of instructions (**P3**). The task itself was identical to that of the control group.

We analyzed the chat protocols using a code-and-count approach. We used the two *Rainbow* [6] categories focused on collaborative argumentation: (1) “Argumentation” (statements used to increase / decrease the believability of a thesis) and (2) “Broaden & Deepen” (arguing and elaborating on arguments, e.g., rebutting an argument or discussing or concepts central to an argument), as well as a third category of our own design (3) “Text Talk” to code messages that elaborate content but not in an argumentative fashion. In order to fairly compare the control and experimental group interactions, we compared the Dec 03 experimental protocols with the combined control protocols of the Nov 19 and Dec 03 sessions.

3 Results and Conclusion

Table 1 summarizes the results with respect to the three codes discussed above. Note, first of all, that the experimental dyads produced more than 5 times as many instances of “Broaden & Deepen” (4.27 vs. 0.75 messages), a significant and large effect. On the other hand, notice that approximately the same amount of “Text Talk” and “Argumentation” took place in the two groups. Yet, the experimental group required less than half the time for the same amount of this elaborative activity.

Table 1. Comparison of conditions based on argumentation codes

Code	**Control**		**Experimental**		**Comparison**			
	M	**SD**	**M**	**SD**	**Diff**	**F**	**p**	**d**
Argumentation*	2.13	2.17	2.09	1.45	-0.04	0.00	1.00	0.00
Broaden & Deepen*	0.75	1.04	4.27	3.74	3.52	6.62	0.02*	1.20
Text Talk^	1.88	2.64	2.09	2.59	0.21	0.03	0.86	0.08

* - From the Rainbow coding system; ^ - Newly defined code

It can be concluded that the experimental intervention was successful in improving the argumentative quality of interaction. Yet, the overall quality of interaction in both conditions was relatively low. We will use the theoretical and technical conclusions of this experiment for the design of future studies.

Acknowledgements. The German Research Foundation (DFG) provided support for this research. We would also like to thank Donald Brown and Maria Daskolia.

References

1. Scheuer, O., Loll, F., Pinkwart, N., McLaren, B.M.: Computer-Supported Argumentation: A Review of the State of the Art. ijCSCL 5(1), 43–102 (2010)
2. Weinberger, A., Stegmann, K., Fischer, F.: Learning to Argue Online: Scripted Groups Surpass Individuals (Unscripted Groups do not). Comp. in Hum. Beh. 26, 506–515 (2010)
3. Weinberger, A., Fischer, F.: A Framework to Analyze Argumentative Knowledge Construction in Computer-Supported Collaborative Learning. Computers & Education 46(1), 71–95 (2006)
4. Rummel, N., Spada, H.: Learning to Collaborate: An Instructional Approach to Promoting Collaborative Problem Solving in Computer-Mediated Settings. Journal of the Learning Sciences 14(2), 201–241 (2005)
5. Nussbaum, E.M.: Collaborative Discourse, Argumentation, and Learning: Preface and Literature Review. Contemporary Educational Psychology 33(3), 345–359 (2008)
6. Baker, M., Andriessen, J., Lund, K., van Amelsvoort, M., Quignard, M.: Rainbow: A Framework for Analyzing Computer-Mediated Pedagogical Debates. ijCSCL 2(2-3), 247–272 (2007)

Investigating the Relationship between Dialogue Responsiveness and Learning in a Teachable Agent Environment

James R. Segedy, John S. Kinnebrew, and Gautam Biswas

Vanderbilt University, Nashville, TN 37235, USA
{James.R.Segedy,John.S.Kinnebrew,Gautam.Biswas}@vanderbilt.edu

Abstract. Using the Betty's Brain Teachable Agents learning environment, we explored a potential relationship between a student's responsiveness to pedagogical agent feedback and the student's learning and performance in the system. We found that both dialogue and action responsiveness metrics were significantly correlated with learning gains in pre- to post-tests, but only action responsiveness was significantly correlated with task performance scores. Dialogue responsiveness was also a better predictor of learning gain than were standardized test scores.

Keywords: learning environments, responsiveness, data collection.

1 Introduction

In this paper, we examine a result from the Betty's Brain learning environment [1] to explore a potential relationship between *student responsiveness* and learning and performance metrics. We define student responsiveness to agents ('responsiveness') as the degree to which students are accepting of advice provided by the agents. In Betty's Brain, agents ask for permission before delivering feedback. For instance, an agent might say 'Excuse me, but you seem to be having trouble. Would you like some help?' Students who are not currently interested in advice can respond by clicking 'no' from a list of options and dismiss the feedback. When students instead click 'yes', they are considered to be 'responsive to dialogue' from the agent. Similarly, when students follow the advice of an agent, they are considered to be 'responsive by action' to the agent's advice.

We conducted a study in 7th grade science classrooms that shows a correlation between student responsiveness and learning gain. Additionally, the dialogue responsiveness was better correlated with learning gains than is a test of prior academic achievement. This result suggests that the responsiveness metrics may be used, in conjunction with other metrics, for more effective system adaptation to individual learners.

2 Classroom Study and Results

We have conducted several classroom studies where students use Betty's Brain to learn and gain a better understanding of a variety of science topics. In these

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 547–549, 2011.

studies, the science content provided by Betty's Brain is closely linked to the school's science curriculum. At the beginning of each study, the science teacher introduces students to the topic during regular classroom instruction. The intervention phase starts with an overview of causal relations and concept maps during a 45-minute class period. This is followed by a hands-on training session with the system on the next day. Over the following 4-5 days, the students learn about the science topic use it to complete their learning task.

The Betty's Brain learning task implements the learning-by-teaching paradigm to help middle school students develop cognitive and metacognitive skills in science and mathematics domains [1,2]. It features Betty, an agent that students teach, and Mr. Davis, an agent that mentors students as they teach. Students using Betty's Brain must read about a scientific topic and structure their newly-acquired knowledge in a causal concept map. Betty uses this concept map to answer questions and take quizzes, and students succeed in the learning task when they have successfully taught Betty everything she needs to know.

In the present study, we worked with 28 7th-grade students in middle Tennessee science classrooms. We have analyzed the data from this study to investigate three research questions: (1) Would more responsive students show greater learning gains? (2) Would more responsive students build more complete concept maps? (3) Is student responsiveness in Betty's Brain more predictive of learning gains and performance measures than standardized test scores?

Learning gains were assessed as the normalized learning gain on pre- and post-tests. The test included 18 multiple-choice questions on climate change and 16 multiple-choice questions on causal reasoning in general. Task performance was calculated based on the completeness and accuracy of each student's final concept map. We define a student's *map score* as the number of correct links minus the number of incorrect links in the student's final concept map.

More responsive students, we hypothesize, will score higher on our learning and performance measures described above. Additionally, if responsiveness strongly affects learning gains and task performance, we expect that it will predict these values at least as well as a student's prior academic achievement. We use student performance on the Tennessee Comprehensive Assessment Program (TCAP) standardized test as a measure of prior academic achievement. The results of this analysis are presented as correlations in Table 1.

These results show that both metrics of responsiveness were more correlated with learning gain than were TCAP scores. Additionally, TCAP scores and

Table 1. Correlation (R) of Learning and Performance with Responsiveness and TCAP (* $p < 0.05$)

	Normalized Learning Gain	Map Score
Dialogue Response	0.477*	0.149
Action Response	0.402*	0.431*
TCAP	0.245	0.405*

action responsiveness were significantly correlated with map score, but dialogue responsiveness was not correlated with map score. One possible interpretation of these results is that TCAP scores are a better predictor of an ability to navigate an open-ended learning environment like Betty's Brain. Students better able to navigate such environments should achieve more success at building their maps. In addition, TCAP scores were not strongly correlated with learning, especially compared to responsiveness. This might indicate that students who were less adept at building concept maps were more willing to listen to advice to read and think carefully about the domain knowledge.

3 Conclusion

In this paper, we have presented results from a study that show the potential value of using student responsiveness metrics as predictors of student performance and learning. As we move forward in this line of work, we will expand our study to obtain stronger evidence supporting the validity of the responsiveness metrics.

If further verified, these metrics could provide easily-calculated indicators of student learning. When combined with other metrics, such as current performance, they could be used to help determine whether or not a student needs more advanced scaffolding or more directed feedback. This would allow us to develop more powerful, adaptive methods for helping unresponsive students re-engage with the Betty's Brain learning task. We will explore these possibilities further as we continue our research.

Acknowledgments. This work has been supported by US Department of Education IES grant #R305H060089, NSF REESE Award #0633856, and NSF IIS Award #0904387. Additionally, this work would not be possible without the generous cooperation and collaboration of the Metro Nashville Public School system and teachers.

References

1. Biswas, G., Leelawong, K., Schwartz, D., Vye, N., Vanderbilt, T.: Learning by teaching: A new agent paradigm for educational software. Applied Artificial Intelligence 19(3), 363–392 (2005)
2. Schwartz, D., Blair, K., Biswas, G., Leelawong, K., Davis, J.: Animations of thought: Interactivity in the teachable agent paradigm. In: Lowe, R., Schnotz, W. (eds.) Learning with Animation: Research and Implications for Design, pp. 114–140. Cambridge University Press, UK (2007)

Using Graphical Models to Classify Dialogue Transition in Online Q&A Discussions

Soo Won Seo[1], Jeon-Hyung Kang[1], Joanna Drummond[2], and Jihie Kim[1]

[1] University of Southern California Information Sciences Institute,
[2] University of Pittsburgh
{soowonse,jeonhyuk}@usc.edu, jmd73@pitt.edu,
jihie@isi.edu

Abstract In this paper, we examine whether it is possible to automatically classify patterns of interactions using a state transition model and identify successful versus unsuccessful student Q&A discussions. For state classification, we apply Conditional Random Field and Hidden Markov Models to capture transitions among the states. The initial results indicate that such models are useful for modeling some of the student dialogue states. We also show the results of classifying threads as successful/unsuccessful using the state information.

Keywords: Student online discussions, Q&A discussion classification.

1 Introduction

Online discussion boards have been a medium for students and instructors to share their ideas in web-enhanced traditional courses and web-based distance-learning courses. This work focuses on the student discussion board that is used by an undergraduate computer science course at the University of Southern California. The course contains programming projects, where a student needs timely support from the instructor or other students to improve his or her performance.

As a step towards assessing student learning in online discussions and assisting instructors, we are investigating whether it is possible to characterize successful versus unsuccessful question and answer (Q&A) type discussions. First, a four-state model was generated based on an analysis of sample discussion threads and its dialogue status [1]. With this states, we use information sharing 'speech acts' and user dialogue roles as features for generating the classifiers. The initial results indicate that graphical models such as HMM and CRF are useful for identifying some of the states. Using annotated state information, the system can classify the discussion successfulness with 96% accuracy.

2 Characterizing Successful vs. Unsuccessful Threads with a State Transition Model

We define *successful discussion* as a discussion in which all of an information seeker's questions get resolved, including initial questions, related questions, similar

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 550–553, 2011.

questions, and questions about derived problems. A four-state model was developed based on an analysis of sample discussion threads: An *initiation* state, an *understanding* state, a *solving* state and a *closing* state [2].

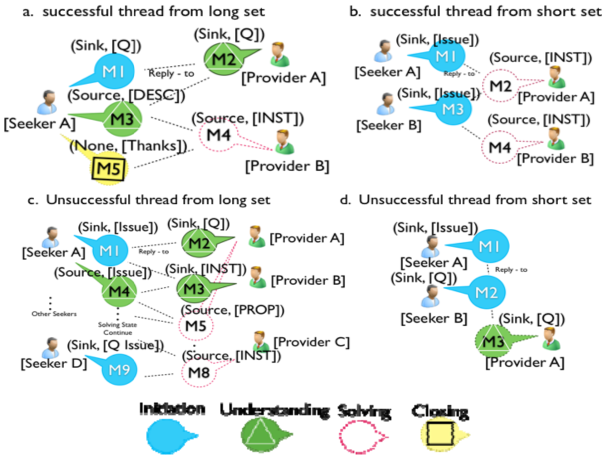


Fig. 1. Discussion thread examples (**a**: I-U-S-C | **b**: I-S-I-S | **c**: I-U-S-I | **d**: I-U)

In the first state (initiation), there must be a problem that exists, which is almost always proposed by the information seeker. In the second state (understanding), the problem is elaborated through communication with other users, who need to understand why this problem exists. In third state (solving), information providers give instructions, propositions, or hints that suggest solutions or actually solve the problem. In Figure 1, we describe four discussion thread examples with the transition model Threads a. and c. are long, and threads b. and d. are short. We labeled user roles (seeker or provider), message roles (sink or source), and speech acts, such as question, instruction, description, done, issue, and proposition that can be automatically labeled by our classifiers [3], [4]. Thread a. has all four states in sequence, ending with a *closing*. Thread b. doesn't go through the *understanding* state and *closing* is missing, but it ends with a *solving* state without an additional issue. Threads c. and d. are both considered unsuccessful since thread c. ends at the seeker's *initiation* state and thread d. ends at the provider's *understanding* state.

3 Experiment and Discussion

A total of 73 threads, containing 254 posts, were used to build a model for state transition. 151 of these posts were labeled solving, 93 were labeled initiation, 8 were labeled closing, and 2 were labeled understanding. Regarding features, we decided to use all sink/source information and THANK relation between posts, which is much correlated to the closing state because people tend to appreciate when they got what they want in a thread. For classification methods, we chose to investigate using decision trees, hidden Markov model (HMM) and linear-chain Conditional Random

Field (CRF). To test supervised learning classifiers, we performed 10-fold cross-validation. For implementation, we used Jahmm for HMM, Mallet for linear-chain CRF and Weka for decision tree, SVM and Logistic Regression.

State Classification

Table 1 shows precision, recall scores and accuracy for the three classifiers. Linear-chain CRF shows highest accuracy although it cannot recognize understanding state, which mainly comes from the fact that only two out of 254 posts are in understanding state.

Table 1. Precision and Recall for Rand, Decision Tree, HMM and linear-chain CRF models

	Precision				**Recall**				**Accuracy**
Model	**I**	**U**	**S**	**C**	**I**	**U**	**S**	**C**	
Tree (J48)	0.7317	0.0000	0.9516	0.7143	0.9677	0.0000	0.7815	0.6250	0.8386
HMM	0.6691	0.5000	1.0000	0.6250	0.9785	0.5000	0.7152	0.6250	0.8071
LCCRF	0.9733	0.0000	0.8721	0.5714	0.7849	0.0000	0.9934	0.5000	0.8937

Discussion Thread Classification

We used the above state information and the final post sink/source labels for classifying successful versus unsuccessful discussion threads. We have the same accuracy of 95.83% in three supervised learning algorithms which are decision tree, support vector machine and logistic regression. The results indicate that state information and the final post sink/source labels are worthwhile to be used in classifying successful threads in online discussion boards.

Table 2. Precision, Recall and Accuracy of classifying Successful/Unsuccessful Threads

Model	***Accu(%)***	***Accu(%) (Short)***	***Precision***	***Recall***	***Accu(%) (Long)***	***Prec***	***Recall***
Tree (J48)	*95.83*	*95.65*	*0.97*	*0.90*	*96.30*	*0.98*	*0.88*
SVM	*95.83*	*95.65*	*0.97*	*0.90*	*92.56*	*0.85*	*0.85*
Logistic Regression	*95.83*	*94.48*	*0.92*	*0.90*	*92.56*	*0.85*	*0.85*

We have presented a model for automatically analyzing patterns of student interactions within discussion threads. As we already have automatic classifiers sink/source, we plan to generate end-to-end automatic classifiers. By combining these automatic classifiers, we hope that we can create assessment tools for instructors.

Acknowledgment

This work is supported by the NSF CISE IIS award (#0917328) and the CRA-W DREU program.

References

[1] Kim, J., Chem, G., Feng, D., Shaw, E., Hovy, E.: Mining and assessing discussions on the web through speech act analysis. Proceedings of the Workshop on Web Content Mining with Human Language Technologies at the 5th International Semantic Web Conference (2006)

[2] Kang, J.H., Kim, J., Shaw, E.: Modeling Successful versus Unsuccessful Threaded Discussions. In: Workshop on Opportunities for Intelligent and Adaptive Behavior in Collaborative Learning Systems, vol. 13 (2010)

[3] Kang, J., Kim, J.: Profiling Message Roles in Threaded Discussions using an Influence Network Model, internal project report (2010)

[4] Drummond, J., Kim, J.: Role of Elaborated Answers on Degrees of Student Participation in an Online Question-Answer. American Educational Research Association (2011)

An Integrated Framework as a Foundation to Develop Meta-learning Support Systems

Kazuhisa Seta[1], Hiroshi Maeno[1], Motohide Umano[1], and Mitsuru Ikeda[2]

[1] Osaka Prefecture University, 1-1, Gakuen-cho, Naka-ku, Sakai, Osaka, 599-8531, Japan
[2] JAIST, 1-1, Asahi-dai, Nomi, Ishikawa, 923-1292, Japan
{seta,umano}@mi.s.osakafu-u.ac.jp, ikeda@jaist.ac.jp

Abstract. It is difficult to generalize and accumulate experiences of system development as methodologies for building meta-learning support systems. Therefore, we need to build a framework that is useful to design and evaluate meta-learning support systems. Thus we propose a framework as a basis to design and evaluate meta-learning support systems.

Keywords: meta-cognition, model, model-based development.

1 Introduction

Kayashima et al. present a sophisticated framework by which we can understand factors of difficulties in performing meta-cognitive activities in performing problem-solving processes [1, 2, 3]. They clarify factors of difficulties based on cognitive psychology knowledge, e.g., segmentation of process, simultaneous processing with other activities, simultaneous processing with rehearsal, a two-layer working memory, etc. [3]. It also clarifies design rationales of each meta-cognition support system.

In this paper, we'll propose a model as a foundation to develop meta-learning support systems: developers can design reasonable meta-learning systems based on the understanding of the characteristics of target learning.

2 Meta-learning Process Model

We provide a detailed model of meta-learning activities in Fig. 1, which depicts a meta-learning process model by extending Kayashima's computational model [4]. It is classified as three layers. At the lowest layer in the figure, i.e., schema level, it represents *"real status"* of a learner's understanding state by performing learning activities.

Upper two layers capture meta-learning processes in a learner's mind. Changing processes of the learner's understanding state by monitoring own schema are situated at the lower layer in WM. Separate representation of schema level and lower layer of WM makes it possible to represent differences between "leaner's *real* state of his/ her understanding" and "learner's *belief* on his/ her own understanding states.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 554–556, 2011.

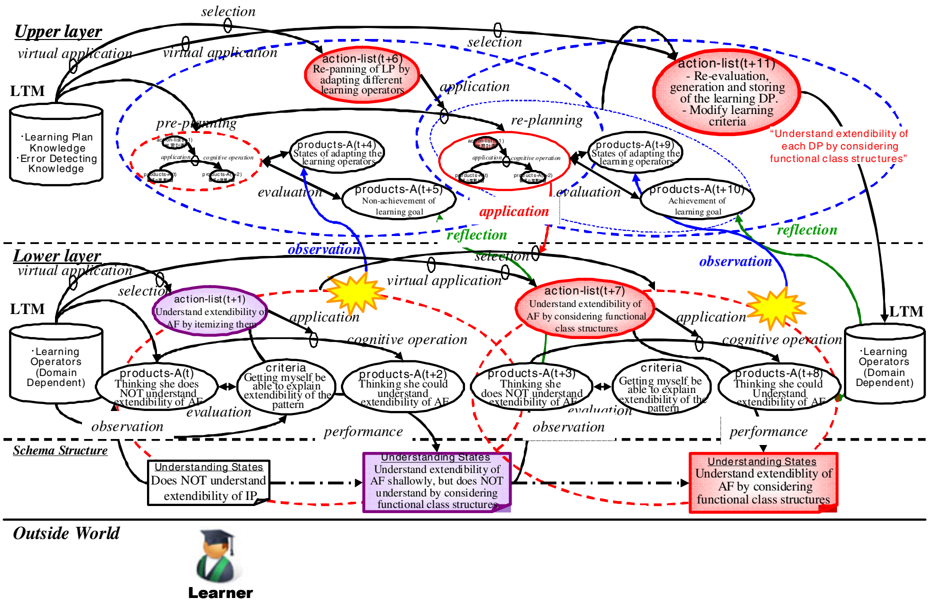

Upper layer
virtual application
virtual application
selection
selection
LTM
·Learning Plan Knowledge
·Error Detecting Knowledge
action-list(t+6)
Re-panning of LP by adapting different learning operators
application
pre-planning
action-list(t+1)
application
cognitive operation
products-A(t)
products-A(t+2)
products-A(t+4)
States of adapting the learning operators
evaluation
products-A(t+5)
Non-achievement of learning goal
re-planning
action-list(t+1)
application
cognitive operation
products-A(t)
products-A(t+2)
products-A(t+9)
States of adapting the learning operators
evaluation
products-A(t+10)
Achievement of learning goal
application
reflection
reflection
action-list(t+11)
- Re-evaluation, generation and storing of the learning DP.
- Modify learning criteria
"Understand extendibility of each DP by considering functional class structures"
observation
observation
Lower layer
virtual application
selection
selection
virtual application
LTM
·Learning Operators (Domain Dependent)
action-list(t+1)
Understand extendibility of AF by itemizing them
application
cognitive operation
products-A(t)
Thinking she does NOT understand extendibility of AF
criteria
Getting myself be able to explain extendibility of the pattern
products-A(t+2)
Thinking she could understand extendibility of AF
evaluation
observation
performance
action-list(t+7)
Understand extendibility of AF by considering functional class structures
application
cognitive operation
products-A(t+3)
Thinking she does NOT understand extendibility of AF
criteria
Getting myself be able to explain extendibility of the pattern
products-A(t+8)
Thinking she could Understand extendibility of AF
evaluation
observation
performance
LTM
·Learning Operators (Domain Dependent)
Schema Structure
Understanding States
Does NOT understand extendibility of IP
Understanding States
Understand extendibility of AF shallowly, but does NOT understand by considering functional class structures
Understanding States
Understand extendiblity of AF by considering functional class structures
Outside World
Learner

Fig. 1. Meta-Learning Process Model

3 Design Concepts for Meta-learning Support Scheme

Table 1 shows five concepts supporting meta-learning: SHIFT, LIFT, REIFICATION, OBJECTIVIZATION, TRANSLATE. They play a guiding role in the design of theory-based meta-learning support systems. We conceptualize from the engineering viewpoint of system development in a specific system in-dependent manner as a basis of functional design for facilitating meta-cognitive learning. By making the concepts as a basis of learning system design explicit and building learning systems based on them, we can accumulate the knowledge for building sophisticated learning systems.

4 Integrating Meta-learning Process Model and Design Concepts

Meta-Learning process model clarifies the factors of difficulties in performing meta-learning processes. Third row in Table 1 shows them.

Table 1 represents correspondence among conceptualizations and their targets to eliminate/ remove factors of difficulties in performing meta-learning processes by integrating two models. For example, SHIFT removes factors of simultaneous processing with other activities and eliminates management of resource, although it *increases* factors of inference of cognitive operation: it does not require on-going monitoring but prompts reflective-monitoring.

The right row in the table illustrates concrete supports implemented in our presentation based meta-learning scheme [4]. For example, based on SHIFT principle, we set presentation task whereby the learner makes a presentation material about already learned topic.

Table 1. Correspondence among Concrete functions Based on Support Concepts and Their Targets

Conceptualization	Meaning	Target to eliminate factors of difficulties	Learning Scheme Design
SHIFT	Stagger the time of developing learning skills after performing problem-solving processes	• Simultaneous processing with other activities • Management of resource + Inference of cognitive operation	Task Design (giving a presentation topic the learner had already learned)
LIFT	Make the learner be aware of learning skill acquisition	• Invisibility • Simultaneous processing with rehearsal	Visualization Environment
		• Acquisition of learning operator • Acquisition of criteria for learning	Guidance Function
REIFICATION	Give appropriate language for his/her self-conversation to acquire learning skills	• Segmentation of process	Providing Domain Specific Terms of Learning Activities
TRANSLATE	Transfer the learning skill acquisition task (LSAT) to a problem-solving task that includes same task structure of LSAT.	• A two-layer WM • Multiple Processing	Task Design (giving a presentation task to explain to other learners)
OBJECTIVIZATION	Objectify her/his self-conversation processes by externalizing them for learning communications with other learners	• (triggering cognitive conflicts)	CSCL Environment

5 Concluding Remarks

In this paper, we proposed an integrated model of meta-learning process model and our conceptualizations. As a result, we can understand which factors of difficulties we should eliminate and how we should realize. It plays an important role to accumulate and share experiences of individual learning system development.

References

1. Brown, A.L., Bransford, J.D., Ferrara, R.A., Campione, J.C.: Learning, Remembering, and Understanding. In: Markman, E.M., Flavell, J.H. (eds.) Handbook of Child Psychology, 4th edn. Cognitive Development, vol. 3, pp. 515–529. Wiley, New York (1983)
2. Flavell, J.H.: Metacognitive aspects of problem solving. In: Resnick, L. (ed.) The Nature of Intelligence, pp. 231–235. Lawrence Erlbaum Associates, Hillsdale (1976)
3. Kayashima, M., Inaba, A.: What Do You Mean by to Help Learning of Metacognition? In: Proc. of the 12th Artificial Intelligence in Education (AIED 2005), Amsterdam, The Netherlands, pp. 346–353 (18-22, 2005)
4. Seta, K., Noguchi, D., Ikeda, M.: Presentation-Based Collaborative Learning Support System to Facilitate Meta-Cognitively Aware Learning Communication. The Journal of Information and Systems in Education (in press, 2011)

Managing the Educational Dataset Lifecycle with DataShop

John C. Stamper[1], Kenneth R. Koedinger[1], Ryan S.J.d. Baker[2], Alida Skogsholm[1], Brett Leber[1], Sandy Demi[1], Shawnwen Yu[1], and Duncan Spencer[1]

[1] Carnegie Mellon University, Human-Computer Interaction Institute
[2] Worcester Polytechnic Institute, Department of Social Science and Policy Studies
{jstamper,krk,alida,bleber,sdemi,shanwen,dspencer}@cs.cmu.edu,
rsbaker@wpi.edu

1 Introduction

An ideal scenario for educational research is to perform an experiment, report and publish results, make the results and data available for verification, and finally allow the data to be used in follow up experiments or for secondary analyses. Unfortunately, this scenario often fails after the results are published. Researchers move on to new data and the old data may linger on a legacy server for a short while before disappearing or becoming impossible to comprehend. Managing the dataset lifecycle is a way to address this problem. DataShop (http://pslcdatashop.org) is a central hub for data management of educational data, and in this paper we show how DataShop fits into the dataset lifecycle.

DataShop is an open data repository and set of associated visualization and analysis tools accessible on the web[2]. DataShop has data comprised of millions of student 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 year-long courses. As of April 8, 2011, over 270 datasets are stored including over 58 million student actions and over 165,000 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" (KCs) needed to perform that action. Visualizations and statistical analysis tools in DataShop are designed to help model builders and analysts find potential flaws in an existing student model. In the hands of trained users these tools provide a method for discovery of KCs that better match student learning data. As the developers of DataShop, we often overlook the repository features in favor of the tools, but the DataShop open repository is rich in features and provides a strong foundation to follow the steps of the educational dataset lifecycle.

2 The Educational Dataset Lifecycle

We define six steps in the educational dataset lifecycle that are illustrated in Figure 1.

Data Design is the most important step in the lifecycle. As part of a research design, the data design should identify what data will be necessary for analysis to confirm research hypotheses, but should also be forward thinking about what data

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 557–559, 2011.

could be used in additional analyses. Any data that is easy to collect, whether or not it will impact the initial research, should be considered. Although this step is not specifically linked to DataShop in Figure 1, DataShop can inform the data design process by providing detailed documentation on our tutor message format[1], which will not only make data easy to import into the database, but also provide an excellent reference for data that researchers should strive to collect. Also, by accessing publicly available projects, researchers can explore the data design of other studies that have used DataShop.

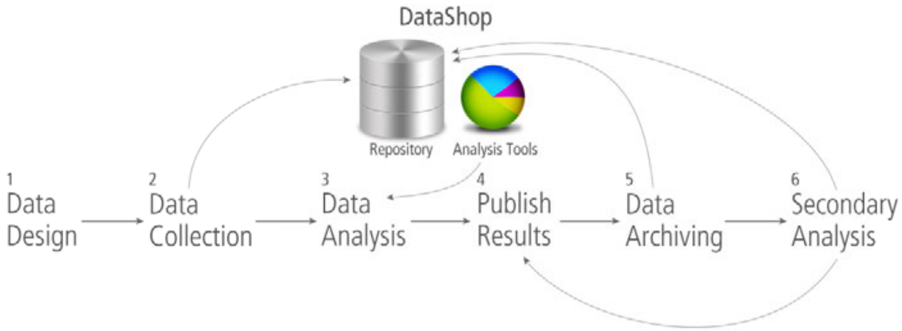


Fig. 1. The six steps of the Educational Dataset Lifecycle showing interactions with DataShop

Data Collection is the actual logging and storing of data. DataShop offers a number of ways to enable data collection. Direct data logging via our logging API allows for data to be automatically imported into DataShop. The CTAT authoring tool [1], which allows non-programmers to create adaptive tutors, has built-in logging tools for DataShop based on the API, and others have used this functionality as well. Data import via text file or the more robust XML format are also available. For transactional data, such as student logs, custom fields are available in the database. For more unusual types of data that may not fit the structure of DataShop's internal database, there is a facility to attach files directly to a project. There are no restrictions on what these files contain (the exception being legal and copyright issues), and the files might be text logs, spreadsheet information, or even video or audio data files.

Data Analysis is the fundamental part of a research project. A strong data design will make setting up the analysis easier. Some analysis can be performed using the tools available in DataShop. The current tools focus on knowledge component (KC) analysis. Learning curve and error analysis are also provided at the student, problem, or problem step level. Understanding that many researchers may not find these tools useful for their research activities, DataShop provides several different export views of the data into a text file format readily accepted into most analytical tools.

Publish Results is an important step in the lifecycle. In addition to just presenting the results, publications should include information about the structure of the data or at least suggest where this information can be found. DataShop provides the ability to

[1] `http://pslcdatashop.org/dtd`

link research papers to a dataset. Linking papers not only provides a background on the dataset; it also increases the visibility of the linked papers to other researchers.

Data Archival differs from data collection in that the archival process must focus on making the data accessible and understandable to future researchers. Archival should include background information that clearly explains the structure of the data. Including additional data files, such as pre and post test materials, as part of a dataset is also important. Once the initial research is completed on a dataset, it should be archived in such a way that others could recreate the experiment, and others can clearly understand the data to allow for secondary analysis.

Secondary Analysis can provide tremendous value to the community but is rarely done. The main obstacle with secondary analysis is that the dataset is often missing critical metadata needed to make sense of the data. This is especially the case when the researcher performing the secondary analysis was not part of the original research. If a project is archived properly, any researcher should be able take the data and recreate the original analysis. It is important that the data is adequately described so that the data is not misunderstood or taken out of context in secondary analysis. If the data is structured in a defined format, such as the tutor message format in DataShop, analyses setup to run on one dataset can be applied to many datasets in the same format. This opens up opportunities for educational data mining studies to cover a large number of domains in an efficient manner. To date, over 75 secondary analyses studies have used datasets in DataShop.

DataShop is focused on becoming the premier repository for educational data. We recognize our current data model does not meet every educational researcher's needs and are working to expand the data model to be more inclusive. We are also working to improve the meta tagging available to allow researchers to better document their datasets, and to make the metadata easier to search.

As the cost of collecting and storing data continues to decrease, researchers will become increasingly inundated with larger and more robust data. This is a good thing, but without a sound data management plan, the data could become worthless or, even worse, become misinterpreted and lead to incorrect conclusions. The US National Science Foundation has recognized the importance of data management, and is now requiring a data management plan to be included in every research proposal submitted. We believe that following the steps of the educational dataset lifecycle and incorporating the DataShop repository will enhance data management and allow for better research in the future. DataShop is supported through NSF award 0836012.

References

1. Aleven, V., McLaren, B.M., 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. 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. (eds.) Handbook of Educational Data Mining, pp. 43–56. CRC Press, Boca Raton (2010)

Eliciting Intelligent Novice Behaviors with Grounded Feedback in a Fraction Addition Tutor

Eliane Stampfer, Yanjin Long, Vincent Aleven, and Kenneth R. Koedinger

Human Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
{stampfer,ylong,aleven,krk}@cs.cmu.edu

Abstract. Standard intelligent tutoring systems give immediate feedback on whether students' answers are correct. This prevents unproductive floundering, but may also prevent students from engaging deeply with their misconceptions. This paper presents a prototype intelligent tutoring system with grounded feedback that supports students in evaluating and correcting their own errors. In a think-aloud study with five fifth-graders, students used the grounded feedback to self-correct, and solved more fraction addition problems with the tutor than with paper and pencil. These preliminary results are encouraging and motivate experimental work in this area.

Keywords: Intelligent Novice, Grounded Feedback, Visual Feedback, Situational Feedback, Fraction Addition.

1 Grounded Feedback and the Intelligent Novice

Intelligent tutoring systems often give immediate *explicit feedback* telling students whether a step is correct, for example by coloring wrong answers red. However, given some scaffolding, students may be able to determine that they have made an error without explicit feedback. If students' actions have consequences that the students recognize as being desirable or undesirable, they can use these consequences to recognize and often learn from their errors [5]. When students correctly interpret the consequences of an action in light of their prior knowledge, we refer to those consequences as *grounded feedback* [2]. Grounded feedback in an Excel formula tutor and an equation-writing tutor has been shown to lead to better learning than explicit feedback [3, 4].

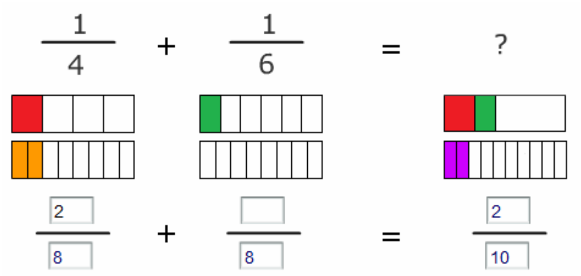
This paper presents a grounded feedback tutor for fraction addition, and a discussion of how students interact with the tutor. For each symbolic fraction n/d, the feedback shows a rectangle divided into d parts, with n colored in. The rectangles allow for easy comparison between the given fractions in the problem and the student-inputted converted and sum fractions. The tutor updates the feedback to reflect the fractions students enter. This tutor contains the key elements of grounded feedback: the feedback by itself does not indicate correctness, and it gives clues about the nature of students' errors (for example it shows if the students' fractions are too big or too small). We found that students connected the grounded feedback to their prior

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 560–562, 2011.

knowledge and used the feedback to correct errors. Students also displayed *intelligent novice* behaviors: they made errors, found them without explicit feedback, corrected them, and appeared to learn from them.

1.1 The Grounded Feedback Tutor

The tutor (see Fig. 1) displays two symbolic fractions and a question mark representing their sum. Below the symbolic forms, fraction bars represent the given fractions and the answer fraction. Below the first set of fraction bars, a second set displays the fractions the students input at the bottom of the interface. The goal is to allow students to see if the original (1/4) and converted (2/8) fractions are equivalent, and whether their answer fraction is equivalent to the sum of the two given fractions (in this case the answer 2/10 is too small). The tutor does not give explicit feedback on the correctness of intermediate steps during problem solving.



$$\frac{1}{4} + \frac{1}{6} = ?$$

$$\frac{2}{8} + \frac{}{8} = \frac{2}{10}$$

Fig. 1. The tutor interface with a composite of typical student errors (converting to eighths works for the first fraction but not the second; adding the given numerators and denominators to find the sum). The first row of fraction bars are given and the second row updates based on the student entries in the text boxes below. Entering a denominator produces dividing lines.

In a think aloud study, participants are asked to perform a task while verbalizing their thoughts [1]. We conducted our think aloud sessions with paper-and-pencil problems followed by tutor problems to determine 1) whether the students correctly interpret the grounded feedback, 2) how students use the feedback, and 3) what intelligent novice behaviors students display.

1.2 The Fraction Addition Think Aloud

Five fifth graders from an all-girls school in Pittsburgh volunteered to participate in the think aloud (all of them had participated in a similar think-aloud with an earlier version of the tutor). According to their math teacher, the girls had learned about fractions but not fraction addition. Each student participated individually in a 20-25 minute think aloud with the experimenters. Students solved three categories of problems: same denominator, one denominator is a multiple of the other, and unrelated denominators. Students solved one problem from each category on paper and one new problem from each category with the tutor. In addition to the grounded feedback, the tutor included a 3-level succession of on-demand hints that first told students to find a common denominator, then gave a general, then problem-specific suggestion for how. The hints did not give students the answer to the specific next step.

With grounded feedback alone, the five students correctly solved more problems with the tutor (12/14) than on paper (8/15). One student did not start the last tutor problem. Students' first attempts with the tutor reflect their problem solving without grounded feedback. Out of eight incorrect first attempts, students self-corrected and ultimately solved six problems (75%) with the grounded feedback alone. Students solved the remaining two tutor problems using on-demand hints (for example, to find a common denominator). After finishing the tutor problems, two students returned to their unrelated-denominator paper problems and corrected their earlier mistakes, suggesting that they learned from the tutor.

Students' comments show how they ground the tutor's feedback in their prior knowledge. For example, one student converted 1/4 to 1/8, but changed it to 2/8 after seeing the fraction bar. The student explained, "a) I looked at the picture and realized they weren't matched up and b) I realized that I'd doubled the bottom but not the top." The interface already displayed the given fraction 1/4, and the student saw the fraction she had entered, 1/8, was much smaller than 1/4. The difference between her expectation (the pictures would match) and the consequences of her action (they did not) alerted her to her error, which she then corrected. The student seemed to already understand how to convert fractions and the images reinforced why that procedure works.

This study suggests grounded feedback can effectively elicit intelligent novice behaviors for fraction addition. Students connected the grounded feedback to their prior knowledge, and used it to evaluate and correct their errors. Although this formative research does not conclude that grounded feedback is better than the alternatives, the results are encouraging, especially in conjunction with existing studies on tutors with grounded feedback.

Acknowledgments. We thank Noboru Matsuda, our participants, and their math teacher. This research was supported in part by the Pittsburgh Science of Learning Center through NSF award SBE-0836012, and the Institute of Education Sciences, U.S. Department of Education, through Grant R305B090023 to Carnegie Mellon University.

References

1. Gomoll, K.: Some Techniques for Observing Users. In: Laurel, B. (ed.) The Art of Human-Computer Interface Design, Reading, MA, pp. 85–90 (1990)
2. Koedinger, K.R., Alibali, M.W., Nathan, M.M.: Trade-offs between Grounded and Abstract Representations: Evidence from Algebra Problem Solving. Cognitive Science 32(2), 366–397 (2008)
3. Mathan, S., Koedinger, K.R.: Fostering the Intelligent Novice: Learning From Errors With Metacognitive Tutoring. Educational Psychologist 40(4), 257–265 (2005)
4. Nathan, M.J.: Knowledge and Situational Feedback in a Learning Environment for Algebra Story Problem Solving. Interactive Learning Environments 5, 135–159 (1998)
5. Ohlsson, S.: Learning from Performance Errors. Psychological Review 103(2), 241–262 (1996)

Competence-Based Knowledge Space Theory as a Framework for Intelligent Metacognitive Scaffolding

Christina M. Steiner and Dietrich Albert

Graz University of Technology, Knowledge Management Institute,
Cognitive Science Section, Brückenkopfgasse 1/VI, 8020 Graz
{christina.steiner,dietrich.albert}@tugraz.at

Abstract. To help learners in acquiring metacognitive skills that are necessary for successfully planning, monitoring, and regulating their learning, metacognitive support and scaffolding mechanisms are needed. Competence-based Knowledge Space Theory constitutes a theoretical framework mainly used for personalising learning to individual learners' domain-specific competence. The paper outlines how this theoretical framework can be utilized for adaptive metacognitive scaffolding tailored to individual learners' needs.

Keywords: self-regulated learning, metacognition, scaffolding, adaptation, Competence-based Knowledge Space Theory.

1 Introduction

The psycho-pedagogical approach of self-regulated learning calls for increased learner control, thus resulting in giving learners greater responsibility over their learning. Self-regulated learning is usually described as cyclical process of forethought, performance, and reflection [1]. Metacognition [2] is a core component of self-regulated learning and refers to processes of goal setting, planning, monitoring, regulating, and self-reflecting. An individual learner may not (yet) have available the necessary metacognitive skills that are necessary for successfully accomplishing a certain self-regulated learning task. As a result, there is a need of providing assistance and scaffolding in order to foster the development of metacognition and thus, of capable self-regulated learners.

In technology-enhanced learning, support for self-regulation and metacognition is realised by providing prompts or tools that assist the different self-regulated learning phases [3], [4]. In particular, intelligent educational adaptation can be exploited to provide the necessary assistance to learners. While research and development in this field originally focused on improving learning of domain competence, by tailoring learning content, sequences, and presentation to the individual user, meanwhile the potential of using adaptive and intelligent tutoring technologies for effectively supporting metacognition is being increasingly acknowledged [5]. In addition, ideas of self-regulated learning inspire the idea of providing learners also control over their user model [6].

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 563–565, 2011.

2 Adaptive Metacognitive Scaffolding

Competence-based Knowledge Space Theory (CbKST) [7] provides a set-theoretic framework for modelling domain and learner knowledge. In its original approach, a knowledge domain is represented by a set of problems. The knowledge state of an individual is the set of problems he/she is capable of solving. Mutual dependencies between the problems of a domain are captured by a so-called prerequisite relation and restrict the number of potential knowledge states that can actually occur. The collection of knowledge states corresponding to a prerequisite relation is called a knowledge structure. Competence-based extensions of the framework take into account the latent cognitive constructs underlying observable behaviour and assume the existence of a set of fine-grained skills that are required for solving problems or that are taught by learning objects of a domain [7]. The subset of skills that a learner has available represents the competence state of this person. By identifying prerequisite relationships among the skills, a competence structure can be built in analogy to a knowledge structure. The relationship between latent competence and observable performance is established through mappings between skills and learning objects or problems of a domain. The theoretical structures of CbKST build the basis for realising intelligent educational adaptation to the current knowledge and competence state of a learner. Thereby, the skills, their structure, and the mapping to learning contents, as well as user information are stored in ontologies, which are then queried and exploited via a learning system's reasoning services.

Until now, CbKST has been applied in technology-enhanced learning primarily as a basis for providing adaptation at a macro- and micro-level with a focus on domain-specific competence [7], [8]. For elaborating a sophisticated approach to metacognitive scaffolding, a detailed consideration of learners' metacognitive abilities can be obtained by modelling metacognitive skills (e.g. based on [2]) in the sense of CbKST and establishing a competence structure of metacognition. The established structures then serve the execution of a non-invasive assessment on the skills available for a learner [9]. To this end, the skills on metacognition are mapped to events of user interaction within the learning system (e.g. use of certain tools). The assessment is realized by updating the probabilities of competence states with each relevant action. Thereby, the probabilities for relevant skills and according states are increased, and probabilities for lacking skills are decreased. This continuous update feeds the user model on the current metacognitive state of a learner. Information explicitly provided by the learner in the tradition of open learner modelling (e.g. self-evaluation, motivational aspects) can be exploited for further augmenting and refining user modelling. The derived assumptions on the metacognitive skills of a learner serve the provision of adaptive interventions tailored to the learner. Adaptation rules in tight relation to the metacognitive skills (indicating threshold values on skill probabilities) specify whether and when an adaptive intervention from a menu of different categories and types (e.g. targeting planning or reflection) is triggered. These interventions realize metacognitive scaffolding during the learning process through interactive dialogues or by recommending the use of certain strategies or tools and may, in turn, prompt user actions and lead to user model updates. A visual sketch of the micro-adaptive scaffolding process is presented in Fig 1.

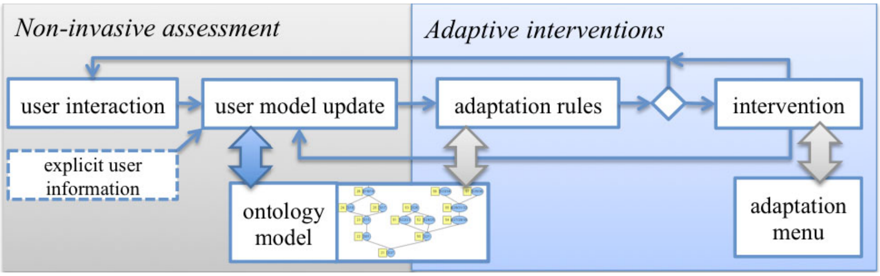


Fig. 1. Adaptive scaffolding process for metacognition

3 Conclusion

The present paper has presented CbKST as a theoretical framework that can be used for adaptive metacognitive scaffolding. In addition to domain-specific adaptivity, intelligent adaptation techniques may support the acquisition of metacognitive abilities, which in turn can improve learning in the subject domain. The detailed elaboration of competence structures and adaptation strategies for metacognition is part of the work done in the ImREAL project (www.imreal-project.eu).

Acknowledgments. The work presented in this paper is supported by the European Commission under the FP7 ICT research priority (ImREAL, project number 257831).

References

1. Puustinen, M., Pulkkinen, L.: Models of self-regulated learning: A review. Scandinavian Journal of Educational Research 45, 269–286 (2001)
2. Pintrich, P.R.: The role of metacognitive knowledge in learning, teaching, and assessing. Theory into Practice 41, 219–225 (2002)
3. Bannert, M.: Promoting self-regulated learning through prompts. Zeitschrift für Pädagogische Psychologie 23, 139–145 (2009)
4. Narciss, S., Proske, A., Körndle, H.: Promoting self-regulated learning in web-based learning environments. Computers and Human Behavior 23, 1126–1144 (2007)
5. Roll, I., Aleven, V., McLaren, B., Koedinger, K.: Designing for metacognition – applying Cognitive Tutor principles to metacognitive tutoring. Metacognition and Learning 2, 125–140 (2007)
6. Dimitrova, V., McCalla, G., Bull, S.: Open learner models: Future research directions. International Journal of Artificial Intelligence in Education 17, 217–226 (2007)
7. Heller, J., Steiner, C., Hockemeyer, C., Albert, D.: Competence-based knowledge structures for personalised learning. International Journal on E-Learning 5, 75–88 (2006)
8. Kickmeier-Rust, M.D., Albert, D.: Micro adaptivity: Protecting immersion in didactically adaptive digital educational games. Journal of Computer Assisted Learning 26, 95–105 (2010)
9. Augustin, T., Hockemeyer, C., Kickmeier-Rust, M., Albert, D.: Individualised skill assessment in digital learning games: Basic definitions and mathematical formalism. IEEE Transactions on Learning Technologies 99 (2010)

Emotion Regulation during Learning

Amber Chauncey Strain and Sidney K. D'Mello

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

Abstract. Learning episodes are rife with emotional experiences, so it is critical that learners regulate negative affective states as they occur. In the present study, learners were instructed to use two forms of cognitive reappraisal to regulate negative emotions that arose during a one hour learning session. Our findings suggest that cognitive reappraisal is an effective strategy for regulating emotions during learning and can help learners achieve better comprehension scores than a do-nothing control.

Keywords: Emotion, emotion regulation, cognitive reappraisal, ITSs.

1 Introduction

Although it is widely known that emotions such as boredom, anxiety, and frustration can negatively impact engagement, task persistence and learning gains [1,2], it is unclear how best to help learners regulate these emotions as they arise. Previous research, outside of learning contexts, has demonstrated that cognitive reappraisal is one of the most effective ways of regulating negative emotions [3]. Cognitive reappraisal involves changing the perceived meaning of a situation to alter its emotional content. The goal of the present study was to examine whether cognitive reappraisal is useful for managing negative emotions during learning. If so, then ITSs can be equipped with the capacity to teach these strategies to help learners regulate negative emotions as they arise.

The present study analysed the effect of cognitive reappraisal on learners' self-reported emotions and performance outcomes. We hypothesized that learners who were instructed to use cognitive reappraisal would report less negatively valenced emotions and achieve better comprehension than learners who received no explicit instruction on the use of cognitive reappraisal.

2 Method

Participants were 103 individuals who volunteered for monetary compensation on Amazon Mechanical Turk™ (AMT). All participants who completed the experiment were paid $5.00. Participants were randomly assigned to one of three cognitive reappraisal conditions: *deep* (*N* = 38)*,* *shallow* (*N* = 33), or *no* reappraisal (control, *N* = 32). Participants in the deep and shallow reappraisal conditions were asked to imagine that they were applying for a job at a powerful law firm and were required to fulfill

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 566–568, 2011.

one special task in order to get the job. Participants in the *deep reappraisal* condition were instructed to imagine that their task was to read a document and check for comprehensibility. Participants in the *shallow reappraisal* condition, on the other hand, were instructed to imagine that their task was to check the document for typos and grammatical errors. Participants in the *control* condition received no instructions about cognitive reappraisal.

In a web-based learning session consisting of 18 trials, participants were asked to learn about the U.S Constitution and Bill of Rights, answer questions about what they learned, and report their affective states at multiple points.

The U.S Constitution and Bill of Rights were presented one page at a time, with approximately 500 words per page. After reading each page, participants were presented with a multiple choice question about what they had just read. Following every page, participants were prompted to report their affective states along the dimensions of valence and arousal on the Affect Grid [see 4].

3 Results

We calculated each participant's mean (across the 18 trials) valence and arousal self-report scores from the Affect Grid. The results yielded a significant effect for valence, *F* (2, 99) = 3.90, *MSE* = 1.95, partial η^2 = .072. Planned comparisons revealed that participants in the deep (*M* = 5.47, *SD* = 1.15) and shallow (*M* = 5.68, *SD* = 1.18) reappraisal conditions reported more positive valence than the control condition (*M* = 4.76, *SD* = 1.46). We also found a significant effect of condition on participants' self-reported arousal, *F* (2, 99) = 4.22, *MSE* = 2.05, partial η^2 = .078. Participants in the deep (*M* =5.26, *SD* = 1.29) and shallow (*M* = 5.51, *SD* = 1.22) reappraisal conditions reporting more arousal than participants in the control condition (*M* = 4.52, *SD* = 1.75).

Figure 1 indicates that learners who use cognitive reappraisal are not only more likely to experience positively valenced emotions; they are also more likely to experience *activating* positive valence like alertness and engagement; these emotions are positively correlated with learning outcomes [2]. Learners who do not use cognitive reappraisal may be more likely to experience negatively valenced, *deactivating* emotions like boredom which is negatively correlated with learning [1].

Proportional scores on the multiple choice questions served as a measure of reading comprehension. We found a marginally significant effect of condition, *F* (2, 94) = 2.74, *MSE* = .025, *p* = .07, partial η^2 = .055. Planned comparisons revealed that participants in the deep (*M* = .799, *SD* = .118) and shallow (*M* = .793, *SD* = .991) reappraisal conditions achieved significantly higher comprehension scores than those in the control condition (*M* = .740, *SD* = .116).

Taken together, these findings indicate that the use of cognitive reappraisal can lead to more positive activating emotions (i.e. positive valence and high arousal) [2] and better comprehension than using no reappraisal.

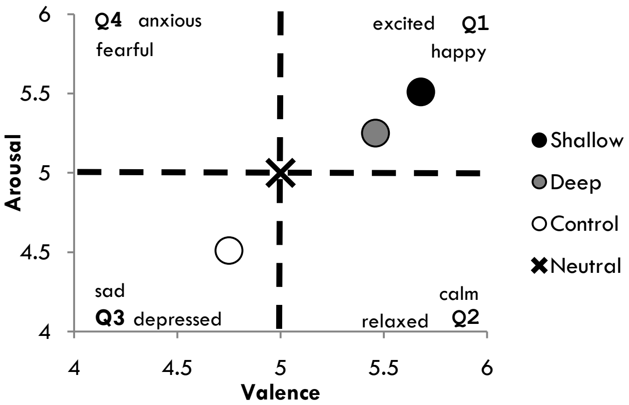


Fig. 1. Mean valence and arousal scores mapped on the Affect Grid

4 General Discussion

We conducted an experiment to test the effect of cognitive reappraisal on affective states and comprehension scores during a reading comprehension task. In general, we found that cognitive reappraisal can be a useful method for regulating emotions and improving comprehension.

These findings have implications for the development of affective-sensitive computerized learning environments and ITSs. According to our findings, intelligent tutoring systems could benefit from not only detecting learner affect, but also providing and scaffold useful emotion regulation strategies that can increase positive emotions, arousal, task-persistence, and learning.

Acknowledgments. This research was supported by the National Science Foundation (ITR 0325428, HCC 0834847). Any opinions, findings and conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of NSF.

References

1. D'Mello, S., Graesser, A.: Emotions during Learning with AutoTutor. In: Durlach, P., Lesgold, A. (eds.) Adaptive Technologies for Training and Education. Cambridge University Press, Cambridge (in press)
2. Pekrun, R.: Academic emotions. In: Urdan, T. (ed.) APA Educational Psychology Handbook, vol. 2, American Psychological Association, Washington (2010)
3. Gross, J.J., Thompson, R.A.: Emotional regulation: Conceptual foundations. In: Gross, J.J. (ed.) Handbook of Emotion Regulation, pp. 3–26. Guilford Press, New York (2007)
4. 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, 493–502 (1989)

Towards a Physical and Personal Math Coin Tutoring System

Georgios Theocharous, Nicholas Butko, and Matthai Philipose

Intel Labs

Abstract. Many elementary mathematics teachers believe that learning improves significantly when students are instructed with physical objects such as coins, called manipulatives. Unfortunately, teaching with manipulatives is a time consuming process that is best with personalized 1-to-1 tutoring. In this paper, we explore the research challenges and solutions of an automated physical and personal tutoring solution.

1 Introduction

The use of physical objects, such as coins, rods, cubes, patterns and other concrete objects called manipulatives, is a widely accepted approach for teaching abstract and symbolic mathematical concepts in kindergarten and early grades [3]. These researchers showed that interaction with concrete objects provides the basis for abstract thoughts. For example, a child might construct an understanding of the meaning of a 5 cent coin by counting 5 pennies one by one and then associating the value of 5 cents with the physical characteristics of a nickel. Unfortunately, teaching early mathematics with coin manipulatives is a time consuming process and ideally occurs as a personal 1-on-1 tutoring with a teacher. Each session may last up to 30 minutes and may have to be repeated many times through the school season before the student finally develops cognitive structures for the different concepts, which include naming the coins, sorting them by size and value, counting them, and adding their values. In this paper we propose an automated math coin tutor.

Building such a tutor is challenging, because not only do we need to deal with tutoring difficulties of teaching mental manipulations of abstract and symbolic structures but also need to deal with the perception of physical objects. We classified our challenges into 4 areas: Coin Perception, Mood Perception, Teaching Dynamics and Optimal Teaching. Next, we summarize our work in each of the research areas.

2 Research Challenges

Coin Perception. For our first challenge we implemented a vision-based coin detection, trucking and clustering system, combined with a projection engine. The system was motivated by 70 video-taped sessions of teachers interacting with

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 569–571, 2011.

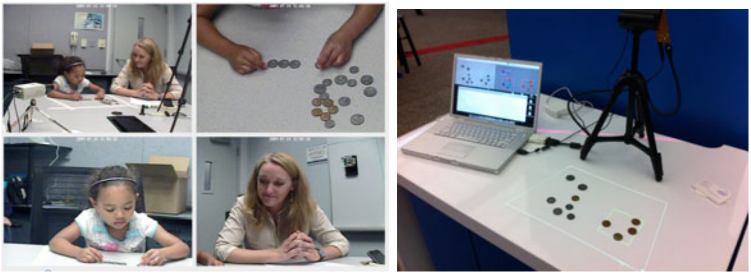


Fig. 1. Four video cameras simultaneously captured the interaction from four angles. The camera/projection system on the right, recognizes the clusters of coins and types within each cluster. In this instance it projects hints on a table to help the student separate the pennies from the rest of the coins.

first grade and kindergarden students as shown in Figure 1. Our coin detector was trained using a cascaded Classifier approach. This approach had about 10-100x fewer false alarms for any level of misses when compared to a base-line Hough transform approach, while analyzing the image over six times faster [4].

Mood Perception. In our second challenge we are concerned with identifying moods the students go though such as tired, confused and thinking (Figure 2) and then building machine learning models for recognizing them. Identifying positive learning moods and implementing strategies that encourage them is an important element of effective tutoring. To encode the various student moods we used facial expression recognition algorithms [2]. We then used machine learning technology to learn to recognize positive and negative moods. Our approach has a precision of 81% for generating 3 labels per second and a 100% for generating 1 label per 10 seconds [1].

Teaching Dynamics. For the third challenge we had to identify the type of lessons and hints that the teachers give, what mathematical concepts each lesson provides, and how do the teachers decide when to give a lesson and a hint. To elicit the relationships between the different concepts that a child needs to learn and how each concept allows for new concepts to be built upon, we looked at the sequence of lessons being taught and how the teacher was able to progress to harder lessons while interleaving those with diagnosing concepts from previous lessons. The concepts and their relationships identified in our experiments involve learning coin names, largest versus smallest, sorting, association of coin names with values, understanding coin values trough counting and learning to count by 1s, 5s and 10s. More detailed results can be found in [4].

Optimal Teaching. Finally, all the elements of the systems need to be encoded in some formal computational decision making approach, which would be able to reason about and diagnose true student moods and concept level and give appropriate lessons and hints to encourage positive mood learning as well as advance the concept level. For this, we chose the framework of partially observable

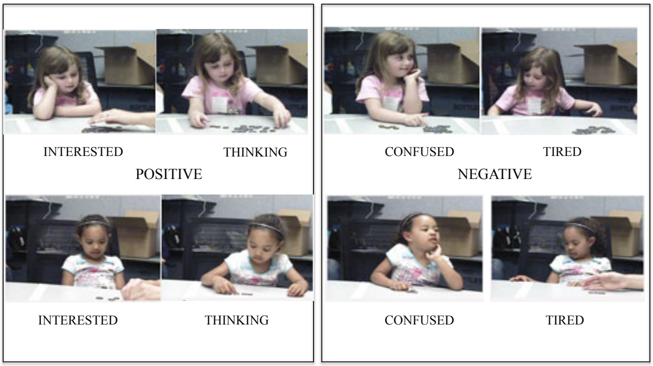


Fig. 2. The figure show positive and negative moods identified. It is is remarkable that different students exhibit similar behavior, which in effect allows to capture the domain computationally.

Markov decision processes (POMDPs). POMDPs are a rigorous mathematical framework for solving problems of sequential decision making under uncertainty, such as tutoring. Uncertainty in tutoring rises from the fact that teacher actions and hints may have uncertain outcomes on the student mood states and concept states and the fact that student mood states and concept states are not directly observable through any sensor (e.g., coin configuration perceptions and facial expression recognition). An early prototype POMDP implementation exhibited a policy that would first try to diagnose the student's level and then try to guide her through the completion of all lessons, by first teaching and then testing, and falling back to teaching when the student could not finish a testing lesson. When the student attention faded off, the system tried to bring it back by asking the student to pay attention [4].

References

1. Butko, N., Theocharous, G., Philipose, M., Movellan, J.: Automated facial affect analysis for one-on-one tutoring applications. In: IEEE International Conference on Automatic Face and Gesture Recognition (2011)
2. Littlewort, G., Bartlett, M.S., Fasel, I., Susskind, J., Movellan, J.: Dynamics of facial expression extracted automatically from video. J. Image And Vision Computing, 615–625 (2004)
3. Piaget, J., Szeminska, A.: The Childs Conception of Number. W. W. Norton Co., New York (1941)
4. Theocharous, G., Butko, N., Philipose, M.: Designing a mathematical manipulatives tutoring system using POMDPs. In: POMDP Practitioners Workshop: Solving Real-world POMDP Problems, ICAPS (2010)

Does Supporting Multiple Student Strategies in Intelligent Tutoring Systems Lead to Better Learning?

Maaike Waalkens[1,2], Vincent Aleven[1], and Niels Taatgen[2]

[1] HCI Institute, Carnegie Mellon University, Pittsburgh PA, USA
[2] Department of Artificial Intelligence, University of Groningen, The Netherlands

Abstract. One feature that makes an Intelligent Tutoring System (ITS) hard to build is strategy freedom, where students are able to pursue multiple solution strategies within a given problem. But does greater freedom mean that students learn more robustly? We developed three versions of the same ITS for solving linear equations that differed only in the amount of freedom. We conducted a study in two US middle schools with 57 students in grades 7 and 8. Overall, students' algebra skills improved. There was no difference in learning gain and motivation between the conditions. Students tended to adhere to a standard strategy and its minor variations, and not pursue alternative strategies. Thus, the study suggests that in early algebra learning, a small amount of freedom is useful, validating, although to a limited degree, one source of complexity in ITS architectures.

Keywords: Intelligent tutoring systems, strategy freedom.

1 Introduction

One feature that increases the complexity of ITS authoring tools and architectures is the ability to support strategy freedom on the part of the student, or, equivalently, multiple solution strategies within the same problem. Researchers and developers have so far assumed that freedom in ITSs is important for learning results. Moreover, it seems counterintuitive to restrict students when, in the given task domain, many solution paths are possible [1].

The issue of whether greater freedom or more structured (or direct) instruction is more educationally effective is being hotly debated in the educational psychology literature [2]. Several researchers claim that students learn with greater understanding when they discover their own procedures instead of only adopting instructed procedures. Others claim that direct instruction is better, partly because discovery learning can overload working memory, or because discovery learning is inefficient, or does not lead to good solutions in the first place. ITSs may be viewed as providing rather direct instruction, but they can be designed in many different ways, to provide differing degrees of structure. The current study investigates the value of allowing multiple solution strategies within any given tutor problem, and thus the value of more complex tutoring architectures capable of supporting them.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 572–574, 2011.

2 Experiment

Three versions of an ITS for solving linear equations were developed, with exactly the same set of 44 equations to be solved. All versions were implemented as example-tracing tutors [3] and differed only in the amount of freedom offered within each problem (or equivalently, the range of solution paths that the tutor recognized as valid solutions). The versions are: (a) *strict standard strategy*, (b) *flexible standard strategy* or (c) *multi strategy*. In the two *standard strategy* conditions, all equations had to be solved with a standard strategy that is widely used in American middle-school mathematics textbooks [4]. Small variations within this standard strategy are allowed in the *flexible standard strategy* version, but not in the *strict standard strategy*. Students had the most freedom in the *multi strategy* condition, where all effective strategies are allowed. The hints are the same in all three versions and (initially) focus on the standard strategy. After every problem-solving step, students explain what they have done, by selecting from a menu.

57 participants (who were starting grade 7 & 8 in fall) from two US middle schools were randomly assigned to one of the three conditions. All students (voluntarily) participated on three consecutive days during the summer holidays, for two hours a day. A paper pre-test at was administered at the beginning of day 1 and a paper post-test at the end of day 3. The pre-test was a subset of the post-test; both tests assessed procedural knowledge, conceptual knowledge and flexibility in equation solving.

3 Results

For the items included in both the pre- and post-test, we used repeated measures ANOVAs to analyze the results. For the “post-test only items,” one way ANOVAs were used. Procedural learning gain was measured with familiar equations (i.e., equations like those encountered in the ITS). On familiar equations there was a significant main effect for test time ($F(1,55)=6.235$, $p=0.016$). Univariate F-test confirmed that students improved significantly from pre- to post-test ($F(1,55)=6.623$, $p=0.0103$). Students did not improve significantly from pre- to post-test on the conceptual items. Further, there were no differences in leaning gain between the conditions on any of the three knowledge types.

In addition, the log data of the student-tutor interactions was analyzed. The learning curve represents changes in student performance over time, subdivided by the knowledge components that make up the overall skill. We used logistic regression to analyze the learning curve. For all skills together the learning curve decreases significantly ($p<0.001$), which means that performance increased over time as students worked with the tutor. We also analyzed the range of strategies used by the students during their work with the tutor. Small variations within the standard strategy were used regularly (17-38%). However, students largely adhered to the *strict standard strategy*, even when they had the freedom to use other strategies. Alternative strategies were rarely used (4-9%).

We also looked at “unnecessarily flagged errors” in the two restricted conditions (i.e., student actions marked wrong that would be allowed in the free condition). In the *strict standard strategy* the average number of these errors per student is 6.67, in the *flexible standard strategy* 0.75.

4 Discussion / Conclusion

In our study, the ITS that we built helped improve students' equation-solving skill, as evidenced by the pre/post learning gains. Surprisingly, the amount of freedom offered by the tutor had no effect on learning. Going strictly by the learning results, we find no support for the architectural complexity needed to support multiple paths. However, the tutor log data indicate that students *do* use the minor variations within the standard strategy in and that especially the strictest condition (*strict standard strategy*) causes many student actions to be unnecessarily flagged as errors, which is clearly undesirable. The strictest version may therefore be too limited. A moderate degree of complexity seems well worth the effort, namely, the ability to track students with respect to a small number of solution paths that are minor variants of a single strategy. By contrast, the different strategies allowed in the free condition (*multi strategy*) were hardly used, so this kind of freedom appears not to be worth the extra effort, at least not in early algebra learning, when the focus (somewhat to our surprise) is on a standard strategy. Overall, the study validates one source of complexity in ITS architectures, though not as strongly as had expected.

We do not mean to argue that tutors with limited freedom are *always* sufficient; in some cases complex systems that support multiple strategies are necessary. Flexibility (the ability to solve equation in multiple ways, preferably with the most efficient method) is an important aspect of skill and understanding and can probably not be mastered with a tutor that offers limited freedom [5]. The current study strongly suggests that allowing strategy freedom in an ITS in itself is not enough to improve flexibility, and that an ITS geared towards fostering strategic flexibility will need to do more than *allow* multiple solutions.

Acknowledgments. Funding for this research was provided by the Pittsburgh Science of Learning Center which is funded by the NSF award No. SBE-0836012.

References

1. VanLehn, K.: The behavior of tutoring systems. International Journal of Artificial Intelligence in Education 16, 227–265 (2006)
2. Dean Jr, D., Kuhn, D.: Direct instruction vs. Discovery: The Long View. Wiley InterScience, Hoboken (2006), `http://www.interscience.wiley.com`
3. 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)
4. Holt, Rinehart, Winston: Holt mathematics teachers edition, Course 2 (2007)
5. Star, J.R., Rittle-Johnson, B.: Flexibility in problem solving: The case of equation solving. Learning and instruction 18, 565–579 (2008)

Observations of Collaboration in Cognitive Tutor Use in Latin America

Erin Walker[1], Amy Ogan[1], Ryan S.J.d. Baker[2], Adriana de Carvalho[1],
Tania Laurentino[3], Genaro Rebolledo-Mendez[4], and Maynor Jimenez Castro[5]

[1] Carnegie Mellon University, [2] Worcester Polytechnic Institute,
[3] SENAI Institute, [4] Universidad Veracruzana, [5] Universidad de Costa Rica
erin.a.walker@gmail.com, aeo@andrew.cmu.edu, rsbaker@wpi.edu,
dikajoazeirodebaker@googlemail.com, grebolledo@uv.mx,
maynorj@gmail.com

Abstract. Cognitive tutoring systems have proven to be effective at improving mathematics learning in economically developed countries, but little is known about how teachers and students use these systems in other cultures. We visited three Latin American countries and observed use of the Middle School Mathematics Tutor in a school in each country. We found that students in these classrooms tended to work more collaboratively than observed students in the United States, in particular engaging in more interdependently-paced work and conducting work away from their own computer. We discuss how cognitive tutors might be improved to be more adaptive to these environments.

Keywords: cognitive tutors, collaborative learning, cultural adaptation.

1 Introduction

There is growing interest in how use of educational software varies across cultures [1]. In our work, we examine the cross-cultural generalizability of cognitive tutors, which compare student problem-solving to a model of behavior, and provide individualized hints and feedback. They have been demonstrated to be successful in classroom contexts [2], but this work has primarily been done in individualist settings. In fact, cognitive tutor design assumes, for the most part, that students are working at their own individual computers and proceed at their own pace. Based on classroom observation, these assumptions are generally met in use of cognitive tutors in American classrooms [3]. It is an open question to what extent the effects of these tutors generalize to collectivist cultures, where individuals are highly integrated into groups and pursue group goals [4]. Assumptions underlying their design (e.g., students work at their own computers and proceed at their own pace) may not be met.

Thus, we visited Brazil, Mexico, and Costa Rica, three countries that are substantially more collectivist than the U.S. [4], and observed student and teacher use of a Scatterplot unit of the Middle School Mathematics Tutor (CT) [3] in each setting. In this unit, students read problem scenarios and plotted two numerical variables on a graph. To do so, they took several scaffolded steps, including labeling axes, choosing

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 575–577, 2011.

a scale, plotting points, and answering interpretation questions. They received feedback on their answers, and could request a hint at any step. We installed the CT in an extant computer lab in each school, which had been donated by the government or a private foundation. These labs were generally unused, as teachers felt that they did not have appropriate educational software or enough time to prepare lesson plans that incorporated technology. Thus, most teachers were enthusiastic about using the CT, which they perceived as requiring little additional preparation, and serving as a good supplement to the exercise-based work that students typically did. All students used the Scatterplot unit for 80 minutes, translated into the local language of instruction. We observed around 100 students in Brazil (12 students per session), 600 students in Mexico (20 to 46 per session), and 90 students in Costa Rica (20 per session). For the most part, sessions were conducted by the students' own math teacher at the school.

2 Patterns of Use in Collectivist Cultures

One major element of cognitive tutor use in these Latin American countries was the interdependent pace of student work. For some CT sessions, particularly in Mexico and Costa Rica, the whole class worked at the same pace, led by the teacher guiding students step by step through the tutor. This approach was common during the first 30 or 40 minutes of the session, when students were unfamiliar with the tutor. The teacher would describe a single tutor step, wait as students executed the step on their own computer, and then give students the correct answer. As students acquired more expertise, teachers would instruct them to do a few steps on their own, and then stop the class to wait for everyone to catch up. During these sessions, students typically did not show exploratory behavior with the tutor; they would wait patiently for the teacher to say they could continue, and follow the teacher's instructions closely. As students moved into an individual work phase, their pace of problem-solving often remained interdependent, but in spontaneously formed groups of two or three people seated at adjacent computers. When one student successfully completed a step, they would inform the other group members of the correct course of action, who would then take the correct step and move on. Within any given group, it varied whether one person always took on the explainer role, or whether it switched as different members of the group were more successful at different steps. During this type of work, the teacher circulated around the classroom to help individual students and groups.

In addition to problem-solving interdependently, we found that students frequently helped each other while working on different problem steps, and thus much of their work did not occur at their own computers. Students interacted either from their own seats or by moving around the class. For example, students would frequently call across the room to ask a friend for help, and the friend would cross the room to give help. In some cases, help-related actions were less directed; a student might go from computer to computer looking for the answer he or she needed, or move around the room giving several classmates information about steps that he or she had solved. When probed on this behavior, students explained that everybody needed to finish, and that the performance of their class was important. Students said that they felt kinship with their classmates, given that they often had the same classmates for several years. Teachers encouraged these collaborative behaviors as they circulated around the classroom. The kinds of help students gave varied between settings. In general, help consisted of verbal content, a demonstration by physically taking control of another person's computer, or a combination of the two. The verbal content of help ranged from domain answers to technology-related help to full explanations, and appeared to be related to the prior knowledge of each collaborating student.

3 Augmenting Cognitive Tutor Design

In this cross-cultural project where we deployed one unit of the CT in schools in three different Latin American countries, we found that, compared to previous work on classrooms in the U.S., students worked more interdependently and spent more time doing work away from their own computers. One improvement to cognitive tutor design suggested by our observations involves modifying knowledge tracing algorithms to account for the possibility that certain students are problem-solving at the same pace. It may even be possible to determine over time which students' performances are linked, by tracking the timing of different students' steps. Students' collectivist behaviors also reflect an opportunity to actively encourage students to seek and give help at appropriate times during their problem-solving, from appropriate people. If a student is clearly struggling, the system could encourage them to go seek help from someone who has already mastered the relevant skill, and if students have mastered a skill quickly, they could be encouraged to help others who have not mastered it. Additionally, when students are judged to be receiving help, it may be effective to introduce scaffolding encouraging students to provide a self-explanation of the demonstrated problem-solving step. In general, regardless of the reasons for the differences observed, it will be productive to expand cognitive tutor design to be more adaptive to collaborative behaviors. If these behaviors are indeed common in CT use in these countries, having systems that can detect and respond to them would likely improve their effects. These behaviors are arguably a desirable way to use cognitive tutors [5], and should be fostered when they occur naturally.

Acknowledgments. Thanks to all the teachers who made this study possible. Thanks to Sofía Pachaco, Xareni Alvarez, Etmon Vega, and Ruth Wylie for their help.

References

1. Nicaud, J.F., Bittar, M., Chaachoua, H., Inamdar, P., Maffei, L.: Experiments With Aplusix In Four Countries. International J. for Tech. Mathematics Education 13(1) (2006)
2. 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, pp. 61–78. Cambridge University Press, Cambridge (2006)
3. 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. ACM, New York (2004)
4. Hofstede, G.: Culture's consequences: Comparing values, behaviors, institutions, and organizations across nations, 2nd edn. Advances in ITS. Springer. Sage, Thousand Oaks (2001)
5. Lou, Y., Abrami, P.C., d'Apollonia, S.: Small group and individual learning with technology: A meta-analysis. Review of Educational Research 71(3), 449–521 (2001)

Cohesion / Knowledge Interactions in Post-tutoring Reflective Text

Arthur Ward[1] and Diane Litman[2]

[1] University of Pittsburgh, Department of Biomedical Informatics,
Pittsburgh, Pa., 15232, USA
akw13@pitt.edu
[2] University of Pittsburgh
litman@cs.pitt.edu

Abstract. In a previous paper we showed that providing a reflective/abstractive text can significantly improve how much middle motivation students learn from qualitative physics tutoring. In this paper we further find that the effect can be substantially improved by adjusting the cohesiveness of that text according to these students' level of prior knowledge. However, in contrast to previous work in the field, we find that our high knowledge students learned significantly more from *high* rather than low, cohesion text.

Keywords: Tutoring, reflection, textual cohesion.

Introduction

In previous work [5], we described a method of improving learning from the Itspoke qualitative physics tutor by giving students a reflective/abstractive text to read after tutoring. This text compared how certain physics concepts (e.g. Newton's Laws) had been applied in different problem situations, which we expected would encourage students to generate better and more abstract representations of those concepts. Our results showed that learning by middle motivation students was substantially improved.

In this paper we further ask if the *cohesiveness* of that text makes it more or less effective. We define cohesion as the extent to which logical, causal or temporal relationships in the text are explicitly stated. In the absence of cohesion, these relationships within a text have to be inferred by the reader. McNamara and her colleagues (e.g. [4]) have shown an interaction between textual cohesion and student knowledge. Students with low domain knowledge sometimes learn better from texts with *high* cohesion. Students with higher domain knowledge can learn better from texts with *low* cohesion.

In this paper we will investigate whether cohesion has similar effects in our reflective/abstractive text. Following McNamara, we hypothesize that our high pre-testers will learn more from a low cohesion text because its cohesive gaps will trigger inference and learning. We expect this inference will improve the students' situation model, and so improve retention for these readers as measured by a delayed post-test. We will also use the same division of students into "high" "middle" and "low" motivation groups as reported previously [5], and investigate interactions between motivation and cohesion.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 578–581, 2011.

1 Study Design, Results and Discussion

In this experiment we use Itspoke, a qualitative physics spoken dialog tutor which is described more completely in [5]. Before tutoring, subjects read background material about physics principles, then took a multiple-choice pre-test to measure their physics knowledge. After this, they engaged the Itspoke tutor in dialogs about five qualitative physics problems. Then they read a post-tutoring reading, then took a post-test which was isomorphic to the pre-test. One week later they returned to take a delayed post-test.

The Itspoke tutor was identical for all subjects, and the only difference between conditions was the content of the post-tutoring reading. In this paper we compare the effects of a high cohesion version of the post-tutoring reflective text (the “hiCoh” condition) to a low cohesion version of the reflective text (the “loCoh” condition).

Table 1. Subject Dist

	loCoh	**hiCoh**
loPre	17	13
hiPre	17	19

As described more completely in [5], subjects were recruited using an extreme groups design [2]. Subjects in the middle third of the pre-test score distribution were dismissed after the pre-test. Subjects with higher scores were retained as high pre-testers (“hiPre”), and subjects with lower scores were retained as low pre-testers (“loPre”). 27 of the remaining students were removed because of incomplete data, and 66 of those remaining were randomized into one of the two cohesion conditions. Their distribution between knowledge category (hiPre, loPre) and cohesion condition (hiCoh, loCoh) is shown in Table 1. As described in [5], we further subdivide these subjects by motivation level. For middle motivation subjects average N per cell was about six.

Both “high” and “low” cohesion versions of our reflective text had similar structure and semantic content. However, the high cohesion version was written to remove places in which inference would be required to understand the low cohesion text. For example, referring expressions were made more consistent, and causal and logical relations that were only implied in the low cohesion version were spelled out.

These differences made the low cohesion text, at 1,541 words, shorter than the high cohesion text, which had 2,161 words. Relevant CohMetrix [3] measures of cohesion were consistently higher for our high cohesion text, supporting the conclusion that cohesive gaps were more prevalent in the low-cohesion text, as we intended.

Results. An anova explaining Normalized Learning Gain (NLG: [post-pre]/[1-pre]) by motivation category (high, mid or low), pre-test category (hiPre or lowPre), cohesion category (hiCoh or loCoh), and their interactions showed a significant three way interaction between motivation category, knowledge level and cohesion type, on the delayed measure of NLG. This suggests that the hypothesized interaction between knowledge level and cohesion type is different at different levels of student motivation.

Next, following the analysis reported in [5], we separately examine the two way interaction between knowledge and cohesion at each of the three levels of motivation. As shown in Table 2, we found that this interaction was significant on the delayed measure of NLG, for students with middle motivation. Middle motivation students also showed a trend toward an interaction on the immediate measure of learning. Highly and poorly motivated students had very non-significant interactions.

Table 2. Knowledge/cohesion interactions for middle motivation students

	pValues			**Mean Norm. Learning Gain**			
NLG			**preTest**	**hiPre**		**loPre**	
Measure	**preTest**	**Cond**	**: Cond**	**hiCoh**	**loCoh**	**hiCoh**	**loCoh**
Immediate	0.535	0.284	**0.071**	0.479	0.306	0.364	0.476
Delayed	0.638	**0.006**	**0.003**	0.506	0.102	0.180	0.312

As can be read from the right four columns of Table 2, our low pre-testers learned more from the low cohesion reflective text than from the high cohesion text. In contrast, the high pre-testers learned more from the high cohesion reflective text.

Post-hoc Tukey-HSD tests indicated that the difference in NLG between cohesion conditions was not significant for the high pre-testers on the immediate post-test ($p = 0.22$), and also not significant for the low pre-testers on either the immediate ($p = 0.79$) or delayed ($p = 0.73$) post-tests. However, the high pre-testers did learn significantly more from high than from low cohesion text, as measured by the delayed post-test ($p = 0.001$).

Discussion. This work shows for the first time that the *cohesiveness* of a reflective text significantly affects learning, and suggests that manipulating cohesion could be helpful for certain students. However the direction of the effect was opposite to what we expected. Other work has shown that high knowledge readers tend to engage and learn more from text (e.g. [1]). We suspect that this effect swamped the effect of low cohesion in triggering inference. Our high knowledge students engaged both texts, and learned more from the text with more explicitly stated content. Our low knowledge readers knew enough physics to make inferences but, having "middle" rather than "high" motivation only did so when triggered by cohesive gaps. This caused them to learn more (although not significantly more) from low cohesion text.

Acknowledgements. This work was supported by the NSF (# 0631930 and 0914615), by the Learning Research and Development Center and the Intelligent Systems Program at the University of Pittsburgh. We thank Maxine Eskenazi for generously supplying the motivation survey used in this work, and Kate Fobes-Riley for useful comments on a previous version of the paper. We are also very grateful to Danielle McNamara, who provided us with copies of her high and low cohesion texts for reference, and also described the process by which they were developed.

References

[1] Boscolo, P., Mason, L.: Topic knowledge, text coherence, and interest: How they interact in learning from instructional texts. The Journal of Exp. Education 71(2), 126–148 (2003)

[2] Feldt, L.: The use of extreme groups to test for the presence of a relationship. Psychometrika 26(3), 307–316 (1961)

[3] Graesser, A., McNamara, D., Louwerse, M., Cai, Z.: Coh-metrix: Analysis of text on cohesion and language. Behavior Research Methods, Instruments, and Computers 36, 193–202 (2004)

[4] McNamara, D.: Reading both high-coherence and low-coherence texts: Effects of text sequence and prior knowledge. Canadian Journal of Exp. Psychology 55, 51–62 (2001)

[5] Ward, A., Litman, D.: Adding abstractive reflection to a tutorial dialog system. In: Proc. 24th Intl. FLAIRS (Florida Artificial Intelligence Research Society) Conference (2011)

Ontology-Supported Scaffolding of Concept Maps

Stefan Weinbrenner, Jan Engler, and H. Ulrich Hoppe

Collide Research Group, University of Duisburg-Essen, Duisburg, Germany
{weinbrenner,engler,hoppe}@collide.info

Abstract. Concept maps are often used in inquiry learning as tools for conceptual modelling, but also as a means to externalise and diagnose conceptual understanding. The latter use is closely related to intelligent feedback and scaffolding. Previous approaches used an expert concept map as a reference to generate intelligent feedback. This paper describes an approach that takes a domain ontology as its only input.

Keywords: pedagogical agents, domain ontologies, concept maps, scaffolding, adaptive support, blackboard architectures.

1 Introduction and Background

The creation of a concept map from a given text requires the reader to identify relevant concepts of the text and find appropriate relations between them. Therefore concept mapping can be used to develop and objectify conceptualisations of scientific knowledge at an early stage of the learning process.

The work presented here has been conducted in the context of the European research project SCY[1]. In SCY-Lab (the SCY learning environment), students work on missions with specific challenges. In order to support science education based on inquiry learning, SCY-Lab provides tools and scaffolds in an adaptive, context-sensitive way. Concept mapping is one of the activities supported in SCY-Lab. Our idea is to support this activity by providing scaffolding based on an ontology.

The use of concept maps can for educational purposes has been studied in a variety of domains with different purposes. Similar to our approach based on certain semantic and structural heuristics, the Reasonable Fallible Analyser (RFA) by Conlon [1] tries to measure a "score" of a concept map. This score is calculated by comparing the concepts of the learner's map with the concepts of a map created by an expert. Conlon claims in [1] that a quantitative score that represents an overall assessment of the concept map could help in giving the feedback to the learner.

Betty's Brain [2] is an example of an educational environment based on the "learning by teaching" paradigm. The learner teaches an avatar called "Betty" by designing a concept map. The learner can ask questions about the concepts and a reasoner tries to answer these questions based on the concepts in the map.

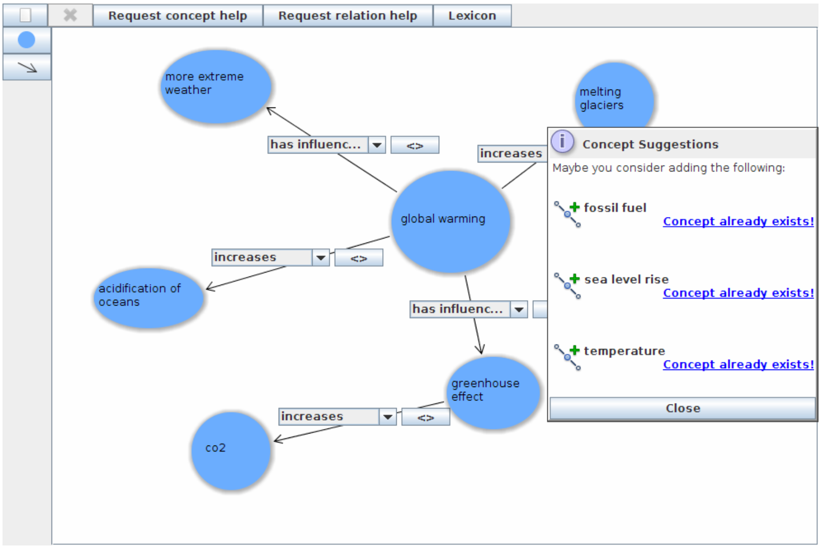
[1] SCY – "Science created by You" is an EU project of the 7th Framework Programme. For more information, see http://www.scy-net.eu (last visited in April 2011).

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 582–584, 2011.

2 Implementation

The knowledge that is used by the system to calculate help proposals is encoded in an ontology that was created in the SCY project as a joint work of the educational and ontology experts. The most relevant part of the SCY architecture here is the pedagogical agent framework [3]. It is based on a blackboard architecture [4], i.e. several agents only communicate over a shared platform (the “blackboard”) and not directly with each other. The SCY blackboard architecture is based on the TupleSpaces approach [5]. In a TupleSpaces system, each client of the central TupleSpaces server is able to read, write and take tuples to and from the server. The concrete platform on which this is implemented is SQLSpaces [6], which comes with a rich feature set, convenient interface and good development support.

Based on this architecture, we implemented a set of agents that extract keywords from a given text and locate the section of the ontology that is relevant in this context. In a next step, the agents compare this section to the learner’s concept map and determine the overlap and the difference. Finally, the most central, but missing concepts and relations are proposed to the learner. Of course, the ontology can only contain a limited amount of terms, which naturally leads to proposals that contain concepts that the learner already inserted under a different label. To solve this issue the learner is able to mark concepts as synonyms of proposals. This will be interpreted by the agents accordingly.



Maybe you consider adding the following:

fossil fuel Concept already exists!

sea level rise Concept already exists!

temperature Concept already exists!

Fig. 1. Screenshot of SCYMapper with ontology-based help

We provide two modes of showing feedback: In the first mode the learner can actively ask for feedback by pushing a “request feedback” button and the other mode provides continuously help without a request by the user. For our purposes, we have modified and extended the existing concept mapping tool in the SCY project called

SCYMapper. The extension mainly focused on adding features to request and show the scaffolds. Figure 1 shows the SCYMapper in the on-demand help mode with the ontology-based concept proposals on the right side. Below the proposals the button to define synonyms is visible.

3 Conclusion

The system proposed in this paper is able to support learners in the process of finding the relevant concepts of a text and transforming them into a concept map. In contrast to similar approaches, this approach does not depend on a manually created expert map, but utilises a given domain ontology. The system was implemented using a blackboard architecture that allows for flexible multi-agent support.

Moreover, a study was conducted that used this system. The goal of the study was to use the ontology to calculate a quality measurement that is not dependent on any human input, but that just takes graph-measures and the agents' results into account. The automatically calculated measures were compared to human assessments. A weighted compound measure including the numbers of concepts and links created as well as their ratio was a good quality predictor in terms of a significant high correlation with human judgements.

Acknowledgements

This study was partially supported by the SCY project funded by the EC under the ICT theme of FP7 (Grant agreement 212814). This text does not represent the opinion of the EC, and the EC is not responsible for any use that might be made of its content. We would like to acknowledge the contributions of the large list of people working on SCY to the work presented here.

References

1. Conlon, T.: 'Please Argue, I Could Be Wrong': A Reasonable Fallible Analyser for Student Concept Maps. In: Proc. of Ed-Media 2004, Lugano, Switzerland, June 21-26 (2004)
2. Gupta, R., Wu, Y., Biswas, G.: Teaching about Dynamic Processes A Teachable Agents Approach. In: Proc. of AIED 2005, Amsterdam, The Netherlands (2005)
3. Weinbrenner, S., Engler, J., Wichmann, A., Hoppe, U.: Monitoring and Analysing Students' Systematic Behaviour - The SCY Pedagogical Agent Framework. In: Proc. of ECTEL 2010, Barcelona (2010)
4. Erman, L.D., Hayes-Roth, F., Lesser, V.R., Reddy, D.R.: The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty. ACM Comput. Surv. 12(2), 213–253 (1980)
5. Gelernter, D.: Generative Communication in Linda. ACM Transactions on Programming Languages and Systems 7(1), 80–112 (1985)
6. Weinbrenner, S., Giemza, A., Hoppe, H.U.: Engineering Heterogeneous Distributed Learning Environments Using TupleSpaces as an Architectural Platform. In: Proc. of ICALT 2007, Los Alamitos (2007)

Character Roles and Interaction in the DynaLearn Intelligent Learning Environment

Michael Wißner[1], Wouter Beek[2], Esther Lozano[3], Gregor Mehlmann[1],
Floris Linnebank[2], Jochem Liem[2], Markus Häring[1], René Bühling[1],
Jorge Gracia[3], Bert Bredeweg[2], and Elisabeth André[1]

[1] Human Centered Multimedia, Augsburg University, Germany
{wissner,mehlmann,haering,buehling,andre}@informatik.uni-augsburg.de
[2] Human-Computer Studies, University of Amsterdam, The Netherlands
{w.g.j.beek,f.e.linnebank,j.liem,b.bredeweg}@uva.nl
[3] Ontology Engineering Group, Universidad Politécnica de Madrid, Spain
{elozano,jgracia}@fi.upm.es

Abstract. In this paper we present the cast of pedagogical agents in the DynaLearn Intelligent Learning Environment. We describe the different character roles and how they interact with the learners. Our aim in using these characters is to increase the learners' motivation.

Keywords: Pedagogical Agents, Virtual Characters, Intelligent Learning Environments, Motivation.

1 Introduction

Virtual characters have been utilized in various learning environments. Most of them feature a teacher-like character that interacts with the learner [1,2,3]. Some make use of a character that can be taught by the user [2]. Some systems feature a fully embodied agent that also communicates non-verbally through gestures [1]. Some feature more than one character [2], but they do not interact with one another. It has also been shown that a one-sided coverage of knowledge transfer or the employment of only a single educational role may either lead to satisfying learning success or motivation, but usually not both at the same time [4].

We therefore hypothesize that a combination of these features, implemented in an integrated set of educational characters may better leverage learning. Hence, in the DynaLearn approach we decided to integrate a whole cast of character roles into our learning environment. DynaLearn is an intelligent learning environment in which learners learn by expressing their conceptual knowledge through qualitative reasoning models [5].

2 The Characters in DynaLearn

As we delineated in [6], the characters in DynaLearn are cartoonish hamsters. Figure 1 shows each of the characters with a typical line of dialog with regard to the model depicted in the center.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 585–587, 2011.

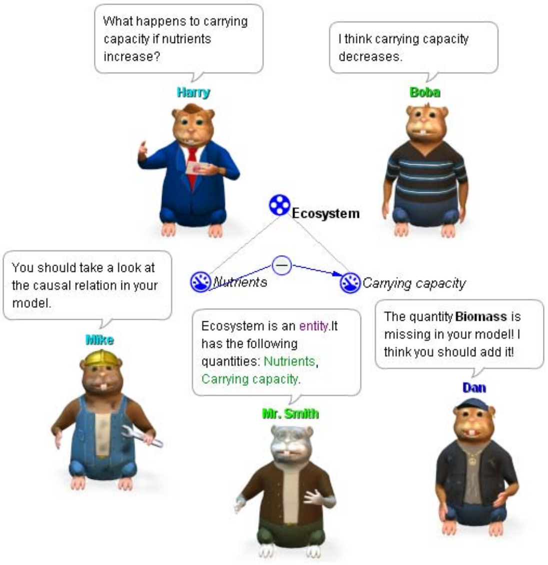

What happens to carrying capacity if nutrients increase?
Harry
I think carrying capacity decreases.
Boba
Ecosystem
Nutrients
Carrying capacity
You should take a look at the causal relation in your model.
Mike
Ecosystem is an entity.It has the following quantities: Nutrients, Carrying capacity.
Mr. Smith
The quantity Biomass is missing in your model! I think you should add it!
Dan

Fig. 1. The DynaLearn Characters (clockwise from top left): Quizmaster (QM), Teachable Agent (TA), Critic, Teacher, Mechanic

Teachable Agent: As the name implies, the TA has a knowledge representation that can be created by the learner. By testing the TA's understanding of the matter through questioning, the learner can evaluate his own presentation of the knowledge and detect mistakes when the TA does not answer as expected. Similar to [2], the interactions learners can perform with their TA in DynaLearn are: Ask (TA answers single questions), Explain (TA provides a step-by-step explanation of an answer) and Challenge (TA takes a quiz).

Mechanic: The task of the mechanic is to support learners in analyzing their model. Oftentimes, the simulation results of the model the learner created are not in line with the learner's expected outcome. An automated diagnostic component (based on [7]) detects these discrepancies, and identifies a minimum number of model components that caused this discrepancy. The mechanic is used to communicate these diagnosis results.

Teacher: In contrast to the mechanic, the teacher offers a more direct kind of help by communicating knowledge related to those aspects of the learning environment that learners can see and interact with. There are three such aspects: First, with respect to any one of the model ingredients, a "What is X?"-question can be posed. Second, with respect to each changing value in a model's simulation a "Why was X derived?"-question can be asked. Thirdly, a list of "How to X?"-questions is constantly generated (where X is a task), based on the tasks that are available from the current context.

Quizmaster: The QM adds a playful element to the software and may be employed in a quiz directly with the human learner or with the learner's TA. The entertaining performance of QM and TA helps to point out flaws and verifies the correct parts of the learner's model. The question generator for the QM is based on the QUAGS question generator [8].

Critic: In contrast to the help provided by the mechanic or teacher characters, the critic's quality feedback about a learner's model is generated through an online repository of models, created by both other learners and experts [9]. Also, while the others are friendly and helpful, the critic is characterized as more strict and unforgiving.

3 Conclusion

We presented our approach to a cast of pedagogical agents, whose interactions with the learner offer a variety of services that help learners to verify and correct their models and conceptual knowledge, while motivating and engaging them at the same time.

Acknowledgments. 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.

References

1. Conati, C., Zhao, X.: Building and evaluating an intelligent pedagogical agent to improve the effectiveness of an educational game. In: Proc. of the 9th International Conference on Intelligent User Interfaces, pp. 6–13. ACM, New York (2004)
2. Biswas, G., Roscoe, R., Jeong, H., Sulcer, B.: Promoting self-regulated learning skills in agent-based learning environments. In: Proc. of the 17th International Conference on Computers in Education (2009)
3. Graesser, A.C., Person, N.K., Harter, D.: The Tutoring Research Group: Teaching tactics and dialog in autotutor. International Journal of AI in Education 12, 257–279 (2001)
4. Kim, Y., Baylor, A.L.: PALS Group: Pedagogical agents as learning companions: The role of agent competency and type of interaction. Educational Technology Research and Development 54(3), 223–243 (2006)
5. Bredeweg, B., Linnebank, F., Bouwer, A., Liem, J.: Garp3 – workbench for qualitative modelling and simulation. Ecological Informatics 4(5-6), 263 (2009); Special Issue: Qualitative models of ecological systems
6. Mehlmann, G., Häring, M., Bühling, R., Wißner, M., André, E.: Multiple agent roles in an adaptive virtual classroom environment. In: Proc. of the 10th International Conference on Intelligent Virtual Agents, pp. 250–256. Springer, Heidelberg (2010)
7. de Koning, K., Breuker, J., Wielinga, B., Bredeweg, B.: Model-based reasoning about learner behaviour. Artificial Intelligence 117, 173–229 (2000)
8. Goddijn, F., Bouwer, A., Bredeweg, B.: Automatically generating tutoring questions for qualitative simulations. In: Proc. of the 17th International Workshop on Qualitative Reasoning, pp. 87–94 (2003)
9. Gracia, J., Liem, J., Lozano, E., Corcho, O., Trna, M., Gómez-Pérez, A., Bredeweg, B.: Semantic techniques for enabling knowledge reuse in conceptual modelling. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 82–97. Springer, Heidelberg (2010)

Effects of Adaptive Prompted Self-explanation on Robust Learning of Second Language Grammar

Ruth Wylie[1], Melissa Sheng[2], Teruko Mitamura[3], and Kenneth R. Koedinger[1]

[1] Human-Computer Interaction Institute, Carnegie Mellon University
[2] Rice University
[3] Language Technologies Institute, Carnegie Mellon University
rwylie@cs.cmu.edu, ms24@rice.edu, {teruko,koedinger}@cmu.edu

Abstract. Prompted self-explanation is a successful intervention for many domains. However, in our previous work within the domain of second language grammar learning, we found no advantage for self-explanation over practice alone. Here, we continue testing the generality of self-explanation through the development of an adaptive self-explanation tutor and report on results of a classroom evaluation (N=92) in which we compare the adaptive tutor to a practice-only tutor. We investigate both procedural and declarative knowledge acquisition as well as long-term retention. Results show that while self-explanation takes more time than practice alone, it leads to greater learning of declarative knowledge. However, there are no differences between conditions on immediate or long-term retention measures of procedural knowledge.

Keywords: Self-explanation, Second Language Learning, Long-term Retention.

1 Introduction

Prompted self-explanation is an instructional strategy in which students provide rationales for steps on solved problems or worked examples. It has been shown to be highly effective for increasing learning in STEM domains [1,2,3]. However, little work has been done in non-STEM domains, and thus, the goal of this work is to test the generalizability of self-explanation. In previous studies on teaching students the English article system, we found that self-explanation led to learning gains, but there was no advantage over a practice-only condition [4]. We also found self-explanation to be relatively inefficient for this domain, but a limitation was a lack of robust learning measures such as long-term retention and declarative knowledge acquisition.

To address these gaps, we built two tutoring systems to teach the English article system (teaching students when to use *a*, *an*, *the*, or *no article*): a practice-only tutor and an adaptive self-explanation tutor. In the practice-only tutor, students see one sentence at a time and select the article that best completes the sentence from the provided menu. In an attempt to make a more efficient tutor, we built an adaptive self-explanation tutor that prompts students to self-explain only when estimates of their prior knowledge for a given article rule are low. Namely, if a student chooses the correct article on their first attempt, they move to the next sentence. If they make an

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 588–590, 2011.

error or ask for a hint on their first attempt, after eventually selecting the correct article, they are prompted to self-explain. In both tutors, students receive immediate feedback and have access to hints.

2 Methodology

Participants were adult English language learners (M=25.5 years, SD=5.3) enrolled in an intensive language program. Instruction and assessments were incorporated into normal classroom activities. We assessed both procedural knowledge and declarative knowledge. The procedural knowledge assessment consisted of problems similar to the tasks students completed as part of tutoring (e.g. *Yesterday, I bought a new car. ___ car is red.*). The declarative knowledge assessment presented students with a feature and asked them to select the corresponding article (e.g. *If a noun has already been mentioned, which article do you use?*). In addition, we also computed instruction time to compare tutor efficiency.

On the day of instruction, students met in the computer lab and began by taking the declarative knowledge pretest. Students were then given a five-minute introduction to both tutoring systems. Students next took the procedural knowledge pretest and were randomly assigned to a tutoring condition. Students then completed both immediate posttests (procedural and declarative knowledge) as well as a demographic survey. Long-term retention procedural knowledge assessments were administered in class one-week and two-months after tutoring.

3 Results

A repeated measures ANOVA on the procedural knowledge (article selection) assessment using the pretest and immediate posttest replicates our previous findings and shows that students in both conditions demonstrate significant pretest to posttest learning gains (F(1,88)=13.1, p=0.001, η^2=0.13). However, there is no difference between conditions (F(1,88)=0.30, p = 0.58) (Table 1). Efficiency results also replicate our previous findings and show that the practice-only tutor is more efficient than the adaptive self-explanation tutor. Students using the practice-only tutor complete the instruction significantly faster (M=15.0 minutes, SD=4.9) than students using the adaptive tutor (M=17.7 minutes, SD=4.3, F(1,90)=7.8, p = 0.006, η^2=0.08).

To test whether the adaptive self-explanation condition leads to more declarative knowledge gain, we conducted a repeated-measures ANOVA on the declarative knowledge assessment. Again, both conditions led to significant pretest to posttest improvement (F(1,77)=86.2, p<0.001, η^2=0.53). Furthermore, results show that the adaptive self-explanation tutor led to greater declarative knowledge gains than the practice-only tutor (F(1,77)=4.39, p=0.04, η^2=0.05) (Table 1).

Finally, we tested whether self-explanation led to better long-term retention. We did a repeated-measures ANOVA using all four instances of the procedural knowledge assessment and found no evidence that self-explanation is better for long-term retention than practice alone (F(3,86)=0.56, p=0.64).

Table 1. Learning gains by condition for the procedural and declarative assessments. Both conditions lead to learning on both assessments, and those using the adaptive tutor make greater gains on the declarative assessment than those using the practice-only tutor.

	Proc. Pretest (SD)	Proc. Posttest (SD)	One-week Retention (SD)	Two-month Retention (SD)	Declarative Pretest (SD)	Declarative Posttest (SD)
Adaptive SE	68.8%	78.0%	82.6%	81.8%	55.8%	90.3%
n=47	(14.2)	(12.9)	(14.8)	(17.5)	(22.5)	(14.3)
Practice-only	71.3%	77.8%	86.2%	83.8%	62.8%	84.6%
n=45	(16.2)	(16.3)	(11.0)	(14.8)	(22.1)	(26.6)

4 Discussion

One of the primary goals of the learning sciences is to understand when and why instructional manipulations succeed. While self-explanation has been called a “domain general” strategy [5], these results suggest that there may be limits to its generalizability depending on the goals of instruction and the nature of the targeted knowledge. Specifically, these results show that self-explanation is generalizable in that it leads to an increase in declarative knowledge over a comparable practice-only condition. However, this additional knowledge does not transfer to better procedural performance, which, for this domain, is the primary goal of instruction. To conclude, this study suggests practical differences between the effects of self-explanation on language learning compared to math and science, and highlights the importance of replicating findings across multiple domains.

Acknowledgments. Special thanks to Erin Walker, Ido Roll, and Amy Ogan, our partner teachers, and the PSLC CTAT and DataShop teams. This work was supported by the PSLC (NSF Award SBE-0354420) and PIER (IES Grant R305B040063).

References

1. Aleven, V., Koedinger, K.: An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. Cog. Sci. 26, 147–179 (2002)
2. Atkinson, R., Renkl, A., Merrill, M.: Transitioning from studying examples to solving problems: Effects of self-explanation prompts and fading worked-out steps. J. of Educ Psych. 95(4), 774–783 (2003)
3. Chi, M., DeLeeuw, N., Chiu, M., LaVancher, C.: Eliciting self-explanations improves understanding. Cog. Sci. 18, 439–477 (1994)
4. Wylie, R., Koedinger, K., Mitamura, T.: Testing the Generality and Efficiency of Self-Explanation in Second Language Learning (submitted)
5. Roy, M., Chi, M.: The self-explanation principle in multimedia learning. In: Mayer, R.E. (ed.) The Cambridge Handbook of Multimedia Learning, pp. 271–286. Cambridge University Press, Cambridge (2005)

Corpus-Based Performance Analysis for Automatically Grading *Use of Language* in Essays

Iraide Zipitria, Jon A. Elorriaga, and Ana Arruarte

The University of the Basque Country, UPV/EHU,
Manuel Lardizabal pasealekua,
20018 Donostia, Spain
{iraide.zipitria,jon.elorriaga,a.arruarte}@ehu.es

Abstract. From its early beginning a big issue in Computer Supported Learning Systems research has been directed to automatically evaluating freely written text. Previous work in use of language grading of summaries showed to be successful identifying critical differences in summary writing maturity. This work, describes further testing discriminating course-to-course improvements of second language learners. Automatic grades are tested on an essay corpus.

Keywords: Use of language, essay grading, corpus-based analysis, behavioural data analysis.

1 Introduction

Using free text allows freedom to write anything that comes to your mind. Therefore, there are greater chances to obtain a better approximation to real learners' knowledge. However, automatic free text evaluation is complex and has to face high levels of uncertainty. Still, developments in Natural Language Processing (NLP) allowed a rebirth with a variety of open-ended approaches in various applications: dialogue systems [1-4], feedback in essays [5], etc. One of the big challenges in automatic grading is to choose adequate diagnosis methods and grading schemes. The work presented here has been carried out in the context of an automatic summary-grading environment. The discourse related grades provided by the environment are adequacy, coherence, cohesion, *use of language* and comprehension. The present study focuses specifically on the impact of *use of language* grading method in essays.

In a previous work, a *use of language* grading model showed to be successful identifying developmentally critical differences in summary writing maturity [6]. This paper aims to observe, (1) if the procedure previously followed with summaries could also be used for essays and, (2) if the model is sensitive enough to perceive course-to-course use of language improvements of second language (L2) learners of Basque language.

2 Grading Use of Language

Cassany [7] claims the relevance of the amount of orthographic, syntactic and lexical errors for *use of language* grading purposes. In a previous work, multiple linear

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 591–593, 2011.

regression analysis was modelled to estimate global *use of language* grades [6]: Four measures based on the Basque spell-checker, 3 measures based on the Error tagger and 8 measures on structure and shallow punctuation error diagnosis were compared to human grades. As a result, the best predictive model (R^2= 0.51, $F(2, 13) = 6.964$, $p = 0.008$, effect size [8] f^2= 0.71 and post hoc power 1-β=0.801) was selected using an error diagnosis tagger (***ETG**$_i$*) available through text parsing [9], and a use of comma diagnosis measure (***UC**$_i$*). ***ETG**$_i$* showed a β_1=- 0.44, t=-3.33 and $p = 0.0054$, and ***UC**$_i$* showed a β_2=-0.45, t=-1.92 and $p = 0.077$.

$$Grade_i = \beta_0 + \beta_1 ETG_i + \beta_2 UC_i + \varepsilon_i$$

3 L2 Learner Corpus Experiment

In the same way that *Grade$_i$* was able to differentiate human *use of language* maturity levels, it should also be able to significantly detect language proficiency level differences in L2 learner essays. Therefore, beginner level students should obtain significantly lower use of language grades than advanced learners.

3.1 Procedure

An essay corpus was automatically graded using the *Grade$_i$* model described in Section 2. The corpus was compound by Basque L2 learner essays gathered from three courses of the same language learning school. The corpus had 226 first course essays, 226 second course essays and 222 third course essays. First course essays had an average length of 515.2 words, 470.78 words in the second course, and 733.01 words in the third one.

3.2 Results

A one-way analysis of variance was run with the aim to observe if *Grade$_i$* measures were sensitive enough to detect *use of language* differences between language mastery levels in a L2 learner corpus. The *Grade*$_i$ method identified significant *use of language* differences between courses; $F(2, 671) = 10.541$ and $p < 0.001$, post hoc power 1-β=0.99. In order to observe course-to-course differences, a Tukey's HSD analysis was applied. Significant differences were found between the first and second courses ($p < 0.001$) and a large effect size (Hedge's $g = 3$). Differences were also significant between the first and the third course ($p < 0.001$, Hedge's $g = 2.9$). However, no significant differences were found between the second and third courses (Hedge's $g = 0.14$).

3.3 Discussion and Conclusions

The *Grade$_i$, use of language grading representation,* has been tested to observe its capability to differentiate L2 learner improvements throughout subsequent courses. *Grade$_i$* proved to be capable to discriminate differences between the first, and second and third courses. But, there was not any difference between the second and third courses. Results are consistent with reports from interviews with L2 teachers who argued that advanced L2 learners show greater improvements in comprehension, cohesion and coherence while beginners gain more improvement in lexicon. In future, it would be interesting to test the same corpus under comprehension, cohesion and coherence measures to observe if expert reports are empirically verified. However, this effect could also be due to the need for a more fine-grained development of the $Grade_i$ measure. We expect to increase the proportion of disambiguation for $Grade_i$ including further grammar error diagnosis in future developments. Finally, results show that the procedure previously followed with summaries can also be applied to essays.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Education (TIN2009-14380), the University of the Basque Country UPV/EHU (UE09/09) and the Basque Government (IT421-10).

References

1. Khuwaja, R.A., Evens, M.W., Joel, A.M., Allen, R.A.: Architecture of CIRCSIM-Tutor. In: Proceedings of the 7th Annual IEEE Computer-Based medical Systems Symposium (1994)
2. Schulze, K.G., Shelby, R.N., Treacy, D., Wintersgill, M.C., VanLehn, K., Gertner, A.: Andes: A Coached Learning Environment for Classical Newtonian Physics. The Journal of Electronic Publishing 1(6) (2000)
3. Graesser, A., Person, B., Harter, D.: Teaching Tactics and Dialog in Autotutor. International Journal of Artificial Intelligence in Education 12, 257–279 (2001)
4. Zinn, C., Moore, J.D., Core, M.G.: A 3-Tier Planning Architecture for Managing Tutorial Dialogue. In: Cerri, S.A., Gouarderes, G., Paraguau, F. (eds.) Proceedings of the 6th International Conference on ITS, pp. 574–584. Springer, Biarritz (2002)
5. Burstein, J., Chodorow, M.: Progress and New Directions in Technology for Automated Essay Evaluation. In: Kapplan, R.B. (ed.) The Oxford Handbook of Applied Linguistics, 2nd edn., pp. 487–497 (2010)
6. Zipitria, I., Arruarte, A., Elorriaga, J.A.: Automatically Grading the Use of Language in Learner Summaries. In: Wong, S.L., Kong, S.C., Yu, F.Y. (eds.) Proceedings of the 18th International Conference on Computers in Education, Putrajaya, Malaysia, pp. 46–50 (2010)
7. Cassany, D.: Didáctica de la Corrección de lo Escrito. Serie lengua, vol. 108. EditorialGráo, de IRIF SL, Spain (1993)
8. Cohen, J.: Statistical Power Analysis for the Behavioral Sciences. Lawrence Erlbaum Associates, Mahwah (1988)
9. Aduriz, I., Aranzabe, M., Arriola, J., Diaz de Ilarraza, A., Gojenola, K., Oronoz, M., Uria, L.: A Cascaded Syntactic Analyser for Basque. In: Proceedings of Computational Linguistics and Intelligent Text Processing, pp. 124–135 (2004)

Evaluating the Effects of a New Qualitative Simulation Software (DynaLearn) on Learning Behavior, Factual and Causal Understanding

Andreas Zitek[1], Michaela Poppe[1], Michael Stelzhammer[1],
Susanne Muhar[1], and Bert Bredeweg[2]

[1] University of Natural Resources and Life Sciences, Department for Water-Atmosphere-Environment, Institute of Hydrobiology and Aquatic Ecosystem Management, Vienna, Austria
{andreas.zitek,michaela.poppe,michael.stelzhammer,
susanne.muhar}@boku.ac.at
[2] University of Amsterdam, Informatics Institute, Amsterdam, Netherlands
b.bredeweg@uva.nl

Abstract. The DynaLearn software, a new intelligent learning environment aimed at supporting a better conceptual and causal understanding of environmental sciences was evaluated. The main goals of these pilot evaluations were to provide information on (1) usability of the software and problems learners encountered, (2) the appreciation of the software, and (3) changes in knowledge and knowledge structure influenced by the activities with DynaLearn. Data were gathered from video analysis, pre-and posttests, and motivation questionnaires. The modeling behavior changed significantly along the use of the different Learning Spaces. Increased causal understanding was documented by an increase of the number of causal expressions from pre- to posttest situation, as well as a significant increase in the degree of abstraction and decrease of wrong causal relations. The results underpin the potential of DynaLearn to support causal and systems based learning in individual and collaborative settings, but also the need for providing additional support and motivating features.

Keywords: DynaLearn, causal reasoning, modeling, knowledge abstraction, qualitative reasoning, environmental education, evaluation.

1 Introduction

Based on promising results of introducing Qualitative Reasoning [1], System Dynamics [2] and Animated Teachable Agents [3] into classrooms for a better, more structured and engaging learning, the DynaLearn project targets at the development of an individualized and engaging cognitive software tool for acquiring conceptual knowledge in environmental science. The software integrates a diagrammatic approach to constructing conceptual models, ontology mapping and semantic technology to ground model building terms and compare to other models, and virtual character technology to provide individualized feedback and enhance learners'

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 594–596, 2011.

motivation [4]. DynaLearn offers six Learning Spaces (LSs) to explore and build models of increasing complexity [5]. The evaluation of the software prototype represents an important part of the project offering first insights in the effectiveness of the available features of DynaLearn to contribute to causal understanding, to evaluate the usability, to detect bugs and collect ideas for improvement as an important basis for adjustment of the upcoming releases.

2 Evaluation Methodology

The evaluations of the prototype of the DynaLearn software primarily aimed at providing information on (1) usability of the software and problems learners encountered when working with the software supporting *'Basic help'*, *'Diagnostic feedback'*, *'Recommendations'*, *'Bug repair'*, (2) the appreciation of the software by students and their impressions and potential ideas for increasing usability, and (3) changes in knowledge and knowledge structure influenced by the activities with DynaLearn.

The first evaluation took place between 19.04. - 22.04.2010 (from 7:50-13:40 each day with breaks) at a technical secondary high school (i:HTL), in Bad Radkersburg, Austria and consisted of 3 days of modeling, with LS1, LS2 and LS4 (each for one day) and a final public presentation of the result by the students at the 4th day. Two students participated, one female and one male, both 16 years old.

The second evaluation was conducted at BOKU University at 19.05.2010 (12:00-17:00) within the course 'Selected Topics of aquatic ecology and river management' as one of 5 afternoons in total with the rest of the course held as PowerPoint presentations. 29 students (12 female, 17 male), 22-39 years old, mainly master students, participated in the event. The event lasted from 13:00-17:00, starting at 12:00 with software installation. Models were developed at LS1, LS2 and LS4.

Expectations from these settings. Videotaping the modeling activities aimed at providing feedback on usability and problems learners encounter with the software. The pre-/posttest (content test) should prove the change in content knowledge. A motivation questionnaire was used to collect attitudes, impressions and ideas.

Data analysis. The data gathered during the pilot evaluations consisted of three components: (1) Video recordings capturing the modeling activities of two i:HTL students, their social interactions, questions and answers, analyzed by using *Transana* software [6]; (2) Textual data, gathered by pre- and posttests and analyzed with the *Atlas.ti* software [7]; (3) Motivation questionnaires.

3 Results

Overall feedback to the DynaLearn approach was rated from neutral to very positive, very interesting and very easy etc. and never negative. It was highly agreed, that the software could also be applied to other fields of science. The model based learning activity as a whole was liked very much. The questions *'Using the software provides a very comfortable way of learning'*, *'The software is easy to use'* and *'The software* *and its features motivated me to build the model'* were rated only slightly above neutral, indicating the need for help functions and other motivating features like teachable agents, grounding by DBpedia or Ontology Based Feedback via a model repository, which are planned to be available for upcoming releases of the software.

The behavior of students differed per LS. Conversation (with student especially in LS2 and with teacher especially in LS4) increased while processing from LS1 to LS4.

Especially LS2 allows an easy translation of ideas into a dynamic model, which can be considered as very important to free up capacity for mastering modeling techniques during early stages of learning to model [8]. This is supported by the finding of [9] that students had difficulties with comprehending a system dynamics modeling formalism, even after they received an instruction. In LS4 they spent almost half of the time discussing their modeling activities, mainly with the teacher, which can be seen as an effect of the advanced modeling possibilities there.

The use of DynaLearn in classrooms led to significant and relevant change in factual knowledge and causal knowledge structure even after a relative short period of working with the software (e.g. one afternoon at BOKU University evaluation).

Acknowledgments. The research presented in the paper is co-funded by the European Commission within the Seventh Framework Programme for Research and Development (FP7, project DynaLearn, number 231526, http://www.dynalearn.eu).

References

1. Bredeweg, B., Forbus, K.D.: Qualitative Modeling in Education. AI Magazine 24(4), 35–46 (2003)
2. Barrientos, M.M.C.: Evaluating system dynamics as a tool for teaching history: an experimental research in classroom. VDM Verlag Dr. Müller, Saarbrücken (2008)
3. Bodenheimer, B., Williams, B., Kramer, M.R., Viswanath, K., Balachandran, R., Belynne, K., Biswas, G.: Construction and Evaluation of Animated Teachable Agents. Educational Technology & Society 12(3), 191–205 (2009)
4. Bredeweg, B., Gómez-Pérez, A., André, E., Salles, P.: DynaLearn - Engaging and Informed Tools for Learning Conceptual System Knowledge. In: AAAI Fall Symposium on Cognitive and Metacognitive Educational Systems (MCES 2009), November 5-7, AAAI Press, Arlington (2009)
5. Bredeweg, B., Liem, J., Beek, W., Salles, P., Linnebank, F.: Learning Spaces as Representational Scaffolds for Learning Conceptual Knowledge of System Behaviour. In: Wolpers, M., Kirschner, P.A., Scheffel, M., Lindstaedt, S., Dimitrova, V. (eds.) EC-TEL 2010. LNCS, vol. 6383, pp. 46–61. Springer, Heidelberg (2010)
6. Mavrou, K., Douglas, G., Lewis, A.: The use of Transana as a video analysis tool in researching computer-based collaborative learning in inclusive classrooms in Cyprus. International Journal of Research & Method in Education 30(2), 163–178 (2007)
7. Lewis, R.B.: NVivo 2.0 and ATLAS.ti 5.0: A Comparative Review of Two Popular Qualitative Data-Analysis Programs. Field Methods 16(4), 439–464 (2004)
8. Hogan, K., Thomas, D.: Cognitive comparisons of students' systems modelling in ecology. Journal of Science Education and Technology 10(4), 319–345 (2001)
9. Sins, P.H.M., Savelsbergh, E.R., van Joolingen, W.R.: The Difficult Process of Scientific Modelling: An analysis of novices' reasoning during computer-based modelling. International Journal of Science Education 27(14), 1695–1721 (2005)

Encouraging Students to Study More: Adapting Feedback to Personality and Affective State

Matt Dennis

Department of Computing Science, University of Aberdeen
m.dennis@abdn.ac.uk

Abstract. My PhD investigates how a conversational agent can adapt feedback to the personality and affective state of learners in order to increase learner motivation. This paper provides an overview of the research area, research questions and work to date.

Keywords: Adaptive feedback, affective states, adaptation, motivation.

1 Introduction

This PhD project investigates how a Conversational Agent (CA) can encourage students to study more. Students fail courses for various reasons such as lack of motivation, being disorganized, and a lack of ability. The CA will aim to help students by providing emotional support messages via adaptive feedback on progress. Addressing a lack of ability is outside the scope of this project.

Modern motivational research has shown that one-size-fits-all theories do not work in the real world, and that motivational levels depend on the individual [1]. We will develop algorithms which enable the CA to modify its behaviour, in particular feedback, based on what it knows about the learner, namely their personality and their affective state.

To model personality, we will use the trait model from psychology, which breaks personality down into characteristics, which we will measure through self-reporting using validated questionnaires (for example mini-markers for the Five Factor Model [2]). Establishing a learner's affective state automatically in real time remains complex, making self-reporting the most popular method [3]. Recent research has shown promising results in gauging a learner's affective state, by tracking pressure on the mouse and seating posture [4], or by analyzing student responses [5]. Personality must, however, affect the propensity of an individual to experience certain emotions. We will attempt to predict the learner's affective state based on their performance and personality. Initial work on this has been done by Zhou et al [6].

Initially we will investigate how the CA can enhance performance-based feedback with affective slants (eg, compare "you are behind" with "you are *slightly* behind"). There is prior research on how to evaluate the emotional effects of this kind of small variations in language [7]. Later, this may be extended to other types of feedback.

Several systems exist which attempt to measure the motivational state of a learner, especially in education [8,9]. Research has been undertaken into how to re-motivate a student [10], and why students drop out from higher education [11]. There has also

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 597–599, 2011.

been research which asks how an intelligent tutoring system could use these tactics to create and maintain motivation in a learner [12]. Building on this research, systems have been developed which recommend a particular strategy to improve a student's motivation to online tutors [13] and implement motivation tactics in an intelligent tutoring system [14]. There is also relevant work on persuasive technology [17].

2 Aims and Objectives

The CA will adapt feedback to learners based on personality and affective state. The project can be broken down into two stages:

Feedback on Progress

1. Which of the currently defined personality traits can be exploited to give feedback that enhances motivation?
2. What algorithms can generate feedback, in producing results from Q1?

Emotional Support on progress

1. Are there simple algorithms for inferring emotions (from personality and progress) which can be exploited to give emotional support that enhances motivation?
2. What algorithms can generate emotional support messages to accompany feedback on progress, taking the results from Q1 into account?

There has been previous research on many of the areas associated with this proposal. However, research on emotional support is focused on the facilitation of learning, whereas this project focuses on motivating people to study more frequently, rather than teaching more effectively, which is a separate field of research.

Intelligent systems do not (currently) modify their feedback to any great extent based on learner personality, and this a goal of the CA.

3 Methodology and Work to Date

Using the trait model, we are establishing which traits need to be considered when adapting feedback. It may be that all traits contribute to some degree, however it would be interesting to know if any can be eliminated. So far, we have undertaken studies to help establish whether the learner's self-efficacy is important [16]. Using the User-As-Wizard method [15], we have investigated whether and how tutors change their feedback as the level of self-efficacy is varied. This results in an algorithm allowing the CA to generate feedback which can then re-judged by humans. The process can then be repeated for other traits, such as neuroticism. Through a process of elimination we can reduce the number of traits that the CA would have to model. A similar approach can also be taken to establish which emotions should be modelled when a student interacts with the CA, in an attempt to augment feedback with support messages, such as "don't worry, other people on the course also struggle with this topic". After this, we can also examine the interaction effects of emotions and personality using similar experiments.

All of this will then be integrated into the CA, which will teach students a simple topic, such as learning Chinese characters. To evaluate the effectiveness of the CA, one group of participants will be taught without the adaptive feedback (control), and another group will have adaptive feedback based on the prior research. Effectiveness will be judged by many factors such as how many students from each group complete the course, time spent studying, learning outcomes, and by surveying the students themselves. Additionally, the impact on existing courses with high drop-out rates may be investigated.

References

[1] Graham, S., Weiner, B.: Theories and principles of motivation. Handbook of Educational Psychology 4, 63–84 (1996)

[2] Thompson, E.R.: Development and validation of an international english big-five mini-markers. Personality and Individual Differences 45(6), 542–548 (2008)

[3] Blanchard, E.G., Volfson, B., Hong, Y., Lajoie, S.P.: Affective artificial intelligence in education: From detection to adaptation. In: AIED 2009, pp. 81–88 (2009)

[4] Cooper, D.G., Muldner, K., Arroyo, I., Woolf, B.P., Burleson, W.: Ranking feature sets for emotion models used in classroom based intelligent tutoring systems. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 135–146. Springer, Heidelberg (2010)

[5] D'Mello, S.K., Dowell, N., Graesser, A.C.: Cohesion relationships in tutorial dialogue as predictors of affective states. In: AIED 2009, pp. 9–16 (2009)

[6] Zhou, X., Conati, C.: Inferring user goals from personality and behavior in a causal model of user affect. In: IUI 2003, Miami, Florida, USA, pp. 211–218 (2003)

[7] van der Sluis, I., Mellish, C.: Towards empirical evaluation of affective tactical NLG. In: Proceedings of the 12th European Workshop on NLG, pp. 146–153 (2009)

[8] Zhang, G., Cheng, Z., He, A., Huang, T.: A WWW-based learner's learning motivation detecting system. In: Proceedings of International Workshop on "Research Directions and Challenge Problems in Advanced Information Systems Engineering, pp. 16–19 (2003)

[9] Mcquiggan, S.W., Mott, B.W., Lester, J.C.: Modeling self-efficacy in intelligent tutoring systems: An inductive approach. UMUAI 18(1), 81–123 (2008)

[10] Hurley, T.: Intervention strategies to increase motivation in adaptive on-line learning. No. 1. Dublin: NCI (2006)

[11] Bruinsma, M.: Motivation, cognitive processing and achievement in higher education. Learning and Instruction 14(6), 549–568 (2004)

[12] Du Boulay, B., Rebolledo Mendez, G., Luckin, R., Martinez-Miron, E.: Motivationally intelligent systems: Diagnosis and feedback. In: AIED 2007, pp. 563–565 (2007)

[13] Hurley, T., Weibelzahl, S.: "MotSaRT" - motivation strategies: A recommender tool for on-line learning facilitators. In: Irish Educational Technology Users' Conference, pp. 1–5 (2007)

[14] Soldato, T.D., Tecnologie, I., Cnr, D., Boulay, B.D.: Implementation of motivational tactics in tutoring systems. J. of Artificial Intelligence in Education 6, 337–378 (1995)

[15] Masthoff, J.: The user as wizard: A method for early involvement in the design and evaluation of adaptive systems. In: Fifth Workshop on User-Centred Design and Evaluation of Adaptive Systems, pp. 460–469 (2006)

[16] Dennis, M., Masthoff, J., Pain, H., Mellish, C.: Does self-efficacy matter when generating feedback? In: Biswas, G., et al. (eds.) AIED 2011. LNCS (LNAI), vol. 6738, pp. 456–458. Springer, Heidelberg (2011)

[17] Fogg, B.J.: Persuasive technology: Using computers to change what we think and do. Morgan Kaufmann, San Francisco (2003)

Motivational and Metacognitive Feedback: Linking the Past to the Present

Alison Hull and Benedict du Boulay

Human Centred Technology Research Group, School of Informatics,
University of Sussex, Brighton, BN1 9QJ, UK
{a.hull,b.du-boulay}@sussex.ac.uk

Abstract. This paper explores the incorporation of metacognitive and motivational feedback into an existing Intelligent Tutoring System (ITS). Both types of feedback are formulated by using the learners' prior experiences and motivational states to improve their ability to successfully engage in problem-solving tasks.

Keywords: Motivation, metacognition, feedback, past experiences, ITS.

1 Introduction

Motivation and metacognition are strongly intertwined [1]. Learners high in efficacy are more likely to use "various cognitive and self-regulatory learning strategies" [2]. Likewise metacognitive skills are required for motivation (e.g. mastery in goal theory requires insight into one's own knowledge and experience). Reflecting and drawing upon prior experience and knowledge are important in the construction of knowledge in terms of utilizing and further developing mental representations and cognitive relationships [3]. Learning from past experience involves metacognitive processes as an act of "reflection on experience" [4]. However, [5] acknowledges that we tend not to be good at recognizing how a past problem can help us with the current one. There have been successes in developing ITSs to address metacognition [6-8], and our research is looking at the relation of this to motivation.

2 Aims and Objectives

Our aim is to improve the learner's focus on the process and experience of problem-solving, by addressing the questions; how effective are different types of feedback (domain, motivational and metacognitive) guided by prior learning experiences and motivational states? What guidelines are required to determine which feedback type to use and when? The potential staging points considered are the start of a session, start of a task, potentially when a learner requests help, end of a task and end of a session. A session is defined as one period of use regardless of length of time.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 600–602, 2011.

3 Methodology

An existing ITS (SQL-Tutor) is being used as the base ITS. SQL-Tutor provides an environment for learners to practice and develop their SQL skills, and has successfully made the transition from research tool to wide-spread use. SQL-Tutor contains a rich open learner model that is based on the Constraint-Based Model approach [9].

The functionality of SQL-Tutor will be extended to include metacognitive and motivational feedback to the learner. This feedback will specifically refer the learner to past metacognitive processes and motivational states to contextualise current issues (such as being stuck). In order to record additional, relevant data, two log files have been designed; one focuses on the timeline of sessions and the other on the timeline of problems. Both log files include activity and self-report data (e.g. help levels encountered, the degree of self-efficacy reported by the learner).

A rules engine will be developed to a) decide when to prompt a learner to self-report on their motivational state (using self-efficacy), b) to determine which prior experience and/or motivational state is relevant for the feedback, and c) to formulate the feedback. While using self-report to gauge motivational state may have potential issues (e.g. interference in the learning itself or the learner pleasing the system [2, 10]), it provides a direct method to capture the learner's thoughts and steps can be taken to minimize any potential issues as discussed in [2]. The feedback will be guided by both previous learning experiences and motivational states of the learner, thereby providing an opportunity for the learner to reflect and draw upon their own learning experiences. In order to achieve this, the concept of relating similar problems by means of templates will be incorporated from a previous study using SQL-Tutor [11].

Two studies will be conducted which will target first year University undergraduate students on computer science and/or business information systems courses. The first is a pilot study which will extend the base SQL-Tutor to include a degree of metacognitive and motivational feedback. It will be used to gain student response to the additional feedback types, including its presentation/timing. The results of this pilot study will be used to direct any changes required before the main study. The main study will be run over a three month period of the participants using SQL-Tutor and will compare learner behaviours and post-activity test results of learners who used different versions of the ITS exploring different feedback regimes, as opposed to an ITS with feedback based only on the current problem.

4 Current State, Expected Contribution and Future Work

The initial design of the additional log files has been completed, along with analysis of the templates used in a previous study of SQL-Tutor. The implementation of the logs in terms of recording is currently underway. The rules engine that governs when a learner is prompted for self-report, as well as formulating the metacognitive and motivational feedback, is currently being designed. The pilot study is scheduled for later in 2011, once the rules engine has been implemented. The results will be studied and further development work will take place before the main study is scheduled for 2012 (the Doctorate work is being undertaken on a part-time basis).

This research aims to contribute to the AIED community by broadening the interaction of an ITS by providing three types of feedback; domain, motivational and metacognitive. The interaction will be further extended by using the prior experiences and motivational states of the learner to formulate the latter two feedback types, as opposed to using just the current affective state (e.g. Prime Climb [12]) or displaying the solution to a past problem as a reminder (e.g. ELM-ART [13]).

References

1. du Boulay, B., et al.: Towards Systems That Care: A Conceptual Framework based on Motivation, Metacognition and Affect. International Journal of Artificial Intelligence in Education (IJAIED) 20(3) (2010)
2. Schunk, D.H., Pintrich, P.R., Meece, J.L.: Motivation in Education: Theory, Research, and Applications, 3rd edn. Pearson Merrill Prentice Hall (2007)
3. Mayer, R.E.: Memory and Information Processes. In: Reynolds, W.M., Miller, G.E., Weiner, I.B.E.i.C. (eds.) Handbook of Psychology, pp. 47–57 (2003)
4. Boreham, N.C.: Learning from Experience in Diagnostic Problem Solving. In: Richardson, J.T.E., Eysenck, M.W., Piper, D.W. (eds.) Student Learning: Research in Education and Cognitive Psychology, pp. 89–97. The Society for Research into Higher Education and Open University Press, Milton Keynes, UK (1987)
5. Robertson, S.I.: Problem Solving. Psychology Press Ltd., Hove (2001)
6. Roll, I., et al.: Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. Learning and Instruction 21(2), 267–280 (2011)
7. Wagster, J., et al.: How Metacognitive Feedback Affects Behaviour in Learning and Transfer. In: Proceedings of the 13th International Conference on Artificial Intelligence in Education. IOS Press, Marina del Rey (2007)
8. Gama, C.: Metacognition in Interactive Learning Environments: The Reflection Assistant Model. In: Proceedings of the International Conference on Intelligent Tutoring Systems. Springer, Berlin (2004)
9. Mitrovic, A., ICTG.Team: Large-Scale Deployment of Three Intelligent Web-based Database Tutors. Journal of Computing and Information Technology 14(4), 275–281 (2006) (Reprinted from Luzar, V., Hljuz-Dobric, V. (eds.) Proc. ITI 2006, Cavtat, Croatia, pp. 135–140 (June 19-22, 2006)
10. de Vicente, A., Pain, H.: Validating the Detection of a Student's Motivational State. In: Proceedings of the Second International Conference on Multimedia Information & Communication Technologies in Education, m-ICTE 2003 (2003)
11. Mathews, M., Mitrovic, A.: Investigating the Effectiveness of Problem Templates on Learning in Intelligent Tutoring Systems. In: Proc. 13th Int. Conf. Artificial Intelligence in Education AIED 2007, Los Angeles (2007)
12. Conati, C., Maclaren, H.: Data-driven refinement of a probabilistic model of user affect. In: Ardissono, L., Brna, P., Mitrović, A. (eds.) UM 2005. LNCS (LNAI), vol. 3538, pp. 40–49. Springer, Heidelberg (2005)
13. Weber, G., Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-based Instruction. International Journal of Artificial Intelligence in Education 12, 351–384 (2001)

Defining Solution Boundaries for EDM Vis

Matthew W. Johnson

Computer Science Department, University of North Carolina at Charlotte,
9201 University City Blvd, Charlotte, NC 28223, USA
mjokimoto@gmail.com

Abstract. Software-tutors like intelligent tutoring systems generate lots of student log-data. However making sense of this data is often difficult, because of the quantity generated. EDM Vis is a visualization tool for interacting, exploring and navigating software-tutor log-data, so educators can *see* what students are doing, and how they are doing it. New methods for clustering similar solution-approaches of students need to be developed so researchers can make better sense of what students are doing, in turn improving software-tutor log-data visualizations and intelligent tutoring systems.

Keywords: Educational Data Mining, Visualization, Student Behavior Modeling.

1 Introduction

Intelligent tutoring systems and computer aided instruction tools have a lot to offer the field of education. However in order to harness the full potential of these tools, researchers, educators and instructional designers need methods of interacting with software-tutor log-data so they can improve their understanding of student learning, from those students' tutor-data. EDM Vis is a visualization tool for interacting, exploring and analyzing the way students solve problems from software-tutor log-data, and is one approach to understanding how students learn in software-tutors. The EDM Vis Tool allows educators and researchers to visualize log data files from computerized tutoring software and *see* how students solved problems. Next educators can use the insights gained about their students' way of thinking to address weaknesses and deficiencies, improve tutoring software or make changes to lectures, in order to support learning.

EDM Vis generates a tree-graph representation of a student-problem model from sequence data, from problems in procedural domains. This model has states and actions which are represented as nodes and edges; depicting many students, working on a single problem. The starting state is the problem definition, and each successor state is the result of performing an action to its parent state. After the tree-graph is made, Bellman-backup [2] is used to assign states a quality-value, as was done by Stamper and Barnes [1]. The purpose of this is to provide a domain independent distance metric which can be used to assess how 'close' a student is to solving a problem, a value based on previous student log data.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 603–605, 2011.

However in some domains, like in the Deep Thought logic tutor [3], student log data can contain more than 40 interactions to solve a problem; making it difficult to efficiently gain insights about the data. However, potentially many of those interactions are part of a sub-solution process, which may be fixed, like the order of operations. Furthermore other similar solutions could in fact be the application of the same approach to solving a problem. For example, in the equation $5x + 6 = 2x + 3$, a student could subtract 2x from both sides than subtract 3 from both sides, or the reverse, subtract 3, than subtract 2x; both procedures are correct, only the order differs. These two 'different' approaches could in fact be considered the same approach pedagogically and can be combined in a visualization, reducing the amount of clutter and redundant information displayed in a visualization of student-problem paths.

2 Aims and Objectives

The goals of this research are to develop data-driven algorithms and metrics for: defining the boundaries of a solution-approach, and defining the boundaries of sub-solution processes.

One method for defining the boundary of a solution-approach is to consider our original tree-graph, and isolate the approaches of two different students. Next calculate the graph edit distance, the number of nodes/edges needed to be removed or added in order to convert one graph into the other, which can be used as a solution similarity metric.

A sub-solution process is a fixed set of steps that presumably leads to a sub-goal. Using the data, I could combine any set of identical actions, two or more, where a new sub-solution is made when its frequency is greater than some threshold, alpha. Once these sub-solutions are determined they could be considered in the solution similarity metric, moved as a single unit, improving the grouping of similar approaches. Another approach would be to get a solution or sub-solution from the tree-graph and treat it as an un-ordered set, identical sets being defined as similar solution approaches, again using the graph distance, ignoring edges this time, as the metric for measuring similarity; in some domains perhaps order does not matter.

3 Methodology

I can develop these new methods for defining boundaries and incorporate them into EDM Vis, then load in previously collected tutor-software log-data from students. Next a user study could be run similar to those found in the Visualization field and I would analyze the insights users were able to gain based on the use of these new techniques. Another facet to the user study would be to measure the usability of the system and compare it to previous usability studies on the EDM Vis Tool. A third option would be to expand on the works of Barnes and Stamper[4], and see if different hints could be provided based on emphasizing sub strategies, for sub-solutions.

One extension of providing different hints would be a comparison of low and high performing students. Looking at entire solution-graphs in the logic domain, has not provided clear results on whether low and high performing students use different strategies to solving logic problems. However, by incorporating the clustering methods for grouping similar strategies and sub-strategies, it could be possible that one group of students use a particular sub-strategy that the other does not. To determine if this exists, we can group our students based on pretest performance. Then compare the usage of particular sub-strategies between the two groups. If the frequency of the different sub-strategies is significantly different, then perhaps we can offer new hints to students depending on the performance group they belong to. Lastly we could compare the success rates of low performing students with the old hints versus the success of similar students provided with the new hints.

4 Contributions

Contributions from this work include: defining sub-solution processes derived from student log-data which can offer an alternative approach for generating knowledge components, though labeling those components may be more difficult. A second contribution is the metric for defining solution-approach similarity, which can be used for defining when two approaches, in a software tutor, should be considered the same or different. Next is the incorporation of these features and the improvements they will offer to EDM Vis, making the exploration of student log-data more efficient and effective, with the potential of providing new hypothesis about tutoring software and learning. Lastly, there is a chance that we could discover fundamental differences between the sub-strategies that students of different performance levels use when solving problems in a variety of different domains.

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. Bellman, R.: A markovian decision process. Indiana Univ. Math. J. 6, 679–684 (1957)
3. Croy, M., Barnes, T., Stamper, J.: Towards an intelligent tutoring system for propositional proof construction. In: Proceeding of the 2008 Conference on Current Issues in Computing and Philosophy, pp. 145–155. IOS Press, Amsterdam (2008), http://portal.acm.org/citation.cfm?id=1566234.1566253
4. 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)

Automatic Generation of Deductive Logic Proof Problems

Behrooz Mostafavi

Department of Computer Science, College of Computing and Informatics, UNC Charlotte
bzmostaf@uncc.edu

Abstract. When developing intelligent tutoring systems, it is necessary to generate questions that reflect the scope of the material and adapt to a student's individual learning needs. Automatic generation of questions for learning tools can provide variation in the questions generated, while eliminating the time cost for the instructor. For courses teaching deductive logic, web-based tools such as Deep Thought allow students to solve deductive logic proofs set by the instructor and record their progress. Our goal is to automatically generate these proofs in such a way that fulfills the parameters set by the instructors, while using the progress recorded to generate further questions specific to the individual student.

Keywords: Question generation, logic proof, intelligent tutoring system.

1 Introduction

Intelligent tutoring systems allow students to use computers to work problems and complete assignments, while adapting to their individual learning needs. These tutoring systems have shown to have a significant effect on learning but take considerable time to construct [5]. Using automatic question generation to provide problems for these tutoring systems can reduce the amount of time required by the instructors to develop these tools, while providing a greater variation of problems for the students. In addition, the data collected from student performance from these tools can be used to generate problems sets that are adaptable to each individual student.

2 Deep Thought

We are developing an automatic question generator for Deep Thought, an intelligent tutoring system for proof solving in deductive logic [3]. It is a web-based tool with a graphical user interface that provides a set of logical premises and buttons for logic axioms that a student must use in order to reach a set conclusion (Fig. 1).

Problem difficulty is based on parameters such as which rules must be used to solve the problem, the number of initial premises, and conclusion complexity. The progress of each student is recorded as they solve the problem, and includes

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 606–608, 2011.

information such as the student identification number, whether they completed the problem successfully, and which rules they used. Currently all problems are set by the instructor, and are static for all students. The data recorded of the student's progress is used for hint generation, and is not used for problem adaptation.

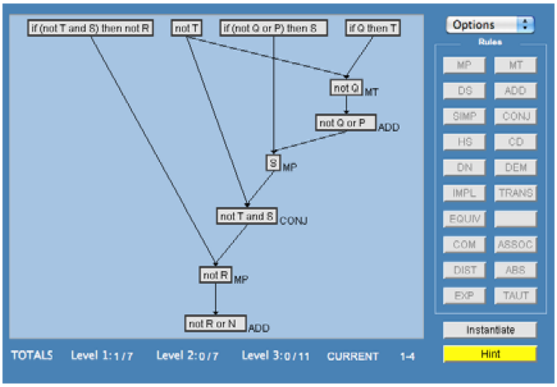


Fig. 1. Deep Thought user interface, showing a successfully completed problem

3 Current Work

We have developed a java-based question generator tool called LQGen (logic question generator). The tool takes as input a hex code that represents the parameters of the problem set by the instructor, and outputs a random problem in the format Deep Thought requires. LQGen currently generates problems using inference rules [2], which represents the first level of problems presented by Deep Thought.

LQGen generates problems using eight inference rules (modus ponens, modus tollens, disjunctive syllogism, addition, simplification, conjunction, hypothetical syllogism, and constructive dilemma), as well as AND, OR, and NOT operators. LQGen also takes as parameters the number of premises used, shared premises (premises that are used for more than one operation), and the number of steps required for completion.

LQGen generates problems by working backwards [2]. It randomly generates a conclusion, and builds a tree of logical statements to a depth of the number of steps required, based on what rules are possible for each step and required by the input parameters. Once the full tree has been generated, the tool then traverses the tree and deletes branches to arrive at the required number of premises. The problem is then checked to determine if it satisfies all the parameters set by the instructor.

For example, if LQGen were instructed to create a problem similar to that in Fig. 1, it would take as its input parameters: four initial premises, simple statement as conclusion, [modus ponens, modus tollens, addition, and conjunction] as required rules, six steps to completion, and shared premises active. It would then create a random simple statement as the conclusion (in this case, not R or N), generate logical statements that were equivalent to the conclusion using a conditionally random required rule (in this case, addition), and then repeat the steps for each newly created statement until the tree depth was at six. It would then prune the tree to reach the four premises.

At each step, LQGen would check the tree to make sure it could satisfy all the parameters. If LQGen determined a satisfactory problem could not be created, it would restart the process.

4 Future Work

We plan to continue developing LQGen to include replacement rules so it can provide all levels of problems in Deep Thought. The data collected by Deep Thought on student progress will be used to influence problem parameters for the LQGen question generation tool, in order to adapt to a student's learning needs.

The current version of LQGen is being tested in a user study to evaluate its effectiveness in generating problems based on parameters set by the instructor, using current instructor-created problems as a basis of comparison. The experiment is a pre-post test design with a control group. About 250 students fill out a pre-test survey, then solve a set of problems in Deep Thought at each difficulty level that vary between existing problems and problems created by LQGen, while filling out a questionnaire asking about each problem's difficulty and any deviations from the set. The progress of each student for each problem is also saved through Deep Thought.

The data gathered from the study are being used to further refine the tool. The study will be ongoing through the rest of the development process of LQGen, with participants having the opportunity to return for re-evaluation of LQGen at each step. At the end of the development process, LQGen will be fully integrated into Deep Thought for its use in course instruction.

References

1. Barnes, T., Stamper, J.: Toward the Extraction of Production Rules for Solving Logic Proofs. In: Proceedings of the 13th International Conference on Artificial Intelligence in Education, Educational Data Mining Workshop (AIED 2007), Marina del Rey, CA (2007)
2. Croy, M.: Problem Solving, Working Backwards, and Graphic Proof Representation. Teaching Philosophy 23, 169–187 (2000)
3. Croy, M., Barnes, T., Stamper, J.: Towards an Intelligent Tutoring System for Propositional Proof Construction. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 373–382. Springer, Heidelberg (2008)
4. McGough, J., Mortensen, J., Johnson, J., Fadali, S.: A web-based testing system with dynamic question generation. In: 31st Annual Frontiers in Education Conference, vol. 3, pp. 23–28 (2001)
5. Murray, T.: Authoring Intelligent Tutoring Systems: An analysis of the state of the art. Artificial Intelligence in Education 10, 98–129 (1999)
6. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education 16, 227–265 (2006)

The MetaHistoReasoning Tool: Fostering Domain-Specific Metacognitive Processes While Learning through Historical Inquiry

Eric Poitras[1,*], Susanne Lajoie[1], Jeffrey Nokes[2], and Yuan-Jin Hong[1]

[1] ATLAS Laboratory, Department of Educational and Counselling Psychology
McGill University, 3700 McTavish St, Montreal, QC H3A 1Y2, CA
[2] Department of History, Brigham Young University, 2130 JSFB,
Provo, UT 84602-6707, USA
eric.poitras@mail.mcgill.ca

Abstract. Learning through historical inquiry requires that learners engage in domain-specific metacognitive regulatory processes. Moreover, there is a pressing need to assist students to regulate certain aspects of their learning. One potential solution is to design technology-rich learning environments as metacognitive tools. In doing so, we aim to evaluate the effectiveness of the scaffolding mechanisms embedded in the MetaHistoReasoning Tool. This computer-based learning environment is designed to support learners in terms of monitoring and controlling the inquiry process as a means to facilitate their construction of coherent multi-layered mental representations of historical events.

Keywords: Historical Inquiry, Metacognitive Tool, MetaHistoReasoning Tool, Top-Down Approach, Bottom-Up Approach.

1 Theoretical Framework

Learning through historical inquiry requires that learners engage in domain-specific metacognitive processes. Conducting an inquiry into complex historical events requires that students analyze, evaluate, and synthesize information gathered from historical sources (e.g., letters, minutes of council meetings, paintings) [1]. As such, learners should engage in metacognitive monitoring processes such as noticing instances of ignorance and asking appropriate historical questions. Learners must then engage in metacognitive control processes such as sourcing, corroboration, contextualization, argumentation, and using substantive concepts [2, 3]. Domain-specific metacognitive knowledge (e.g., using meta-concepts such as historical causation) enables learners to monitor and control their construction of coherent multi-layered mental representation of historical events (i.e., event model composed of representation of texts, events and subtexts) according to discipline-based knowledge and practices [4].

[*] Corresponding author.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 609–611, 2011.

2 Learning Issues

However, the existing empirical evidence regarding learners metacognitive and self-regulatory abilities suggest that there is a pressing need to assist them in regulating certain aspects of their learning [5, 6]. Specifically, learners often have difficulties noticing unexplained historical events, asking themselves why they occurred, and generating tentative causes while reading an historical narrative text. Moreover, learners who are assisted to do so fail to construct coherent mental representations of these historical events. This finding is attributed to the combination of (1) learners' low prior knowledge and (2) the constraints of the task (i.e., the unavailability of relevant historical sources) [5].

3 Research Proposal

We [5] and other researchers [6] have begun to address these issues using the metaphor of designing computer-based learning environments as metacognitive tools [7, 8]. The MetaHistoReasoning Tool (MHRt) is a metacognitive tool designed in order to assist learners to construct coherent multi-layered mental representations of historical events. The scaffolding mechanisms embedded in the MHRt support learners in terms of monitoring and controlling the inquiry process according to discipline-based knowledge and practices. The aim of this research proposal is to empirically evaluate the effectiveness of the elements and principles guiding the design of the embedded scaffolding mechanisms. The research addresses the following questions: does having the benefit of the scaffolding mechanisms embedded in the MHRt result in (1) fostering domain-specific metacognitive processes and (2) constructing coherent multi-layered mental representations of historical events?

3.1 Research Hypotheses

The following hypotheses are tested as part of this research proposal:

1. If the scaffolding mechanisms embedded in the MHRt are an efficient means to facilitate the construction of coherent multi-layered mental representations of historical events, then pre- to post-test shifts in event model are expected to be obtained to a greater degree for learners who have the benefit of the scaffolding mechanisms compared to those who do not, while controlling for the mediating effects of the time spent conducting the inquiry.
2. If the scaffolding mechanisms embedded in the MHRt facilitate pre- to post-test shifts in event model because they foster domain-specific metacognitive processes, then we expect that practicing and refining these skills mediates pre- to post-test shifts in event model.

3.2 Research Design

The research proposal follows a three-group pretest-posttest experimental design with time (i.e., pre- and post-test) as the within-groups and condition (i.e., treatment & silent, treatment & think aloud, and control & silent) as the between-groups factor. We collect, align, and then converge both product (i.e., pre- to post-test shifts in event model measures) and process data (i.e., think aloud protocols augmented with log file trace data and time-stamped video screen capture data) from multiple sources as a means to evaluate the effectiveness of the scaffolding mechanisms embedded in the MHRt.

Participants learn about the Deportation of the Acadians (1755-1763) in our laboratory either using (i.e., treatment) or not using the MHRt (i.e., control & silent). In order to verify the presence and magnitude of reactivity effects in regards to the concurrent think aloud protocols [5, 9, 10], participants undergoing the treatment either learn silently (i.e., treatment & silent) or while performing a concurrent think aloud protocol (i.e., treatment & think aloud). Moreover, the concurrent think aloud measure is combined with unobtrusive on-line measures [11] such as log file traces and time-stamped video screen captures. Data analysis is both quantitative (e.g., ANCOVAs, state-transition analyses) and qualitative (e.g., length and depth of argument chains).

4 Broader Impact of Proposed Research

The broader impacts of the proposed research are to advance domain-specific theories of metacognition, and the role of advanced learning technologies in history education. The scaffolding mechanisms embedded in the MHRt enable learners to regulate certain aspects of the inquiry process that are critical in constructing coherent multi-layered mental representations of historical events. In doing so, the MHRt enables learners to acquire life-long learning skills while performing authentic tasks within their discipline.

References

1. Leinhardt, G., Young, K.M.: Two texts, three readers: Distance and expertise in reading history. Cognition and Instruction 14, 441–486 (1996)
2. Wineburg, S.S.: Reading Abraham Lincoln: An expert/expert study in the interpretation of historical texts. Cognitive Science 22, 319–346 (1998)
3. van Drie, J., van Boxtel, C.: Historical reasoning: Towards a framework for analyzing students' reasoning about the past. Educational Psychology Review 20, 87–110 (2008)
4. Wineburg, S.S.: The cognitive representation of historical texts. In: Leinhardt, G., Beck, I.L., Stainton, C. (eds.) Teaching and learning in history, pp. 85–135. Lawrence Erlbaum Associates, Inc., Hillsdale (1994)
5. Poitras, E., Lajoie, S., Hong, Y.: The Design of Technology-Rich Learning Environments as Metacognitive Tools in History Education. Instructional Science
6. Greene, J., Bolick, C.M., Robertson, J.: Fostering historical knowledge and thinking skills using hypermedia learning environments. Computers & Education 54, 230–243 (2010)
7. Azevedo, R.: Computer environments as metacognitive tools for enhancing learning. Educational Psychologist 40, 193–197 (2005)
8. Azevedo, R.: Understanding the complex nature of self-regulatory processes in learning with computer-based learning environments: an introduction. Metacognition & Learning 2(2-3), 57–65 (2007)
9. Schraw, G.: Measuring self-regulation in computer-based learning environments. Educational Psychologist 45(4), 258–266 (2010)
10. Bannert, M., Mengelkamp, C.: Assessment of metacognitive skills by means of instruction to think-aloud and reflect when prompted. Does the verbalization method affect learning? Metacognition & Learning 3, 39–58 (2008)
11. Azevedo, R., Moos, D.C., Johnson, A.M., Chauncey, A.D.: Measuring cognitive and metacognitive regulatory processes during hypermedia learning: issues and challenges. Educational Psychologist 45(4), 210–223 (2010)

Automatic Identification of Affective States Using Student Log Data in ITS

Ramkumar Rajendran

IITB-Monash Research Academy, IIT Bombay, India

Abstract. Affect-based computing is one of the important research areas in Intelligent Tutoring Systems (ITS). Previous approaches have dealt with affective state analysis based on the data from hardware sensors like eye-tracker, pressure sensitive chairs. However, automatically identifying the affective states only from the student log data is still an important research question. In this proposal, we identify students' affective states by examining patterns in the ITS student log data that contains information about student response, time taken to answer and so on.

1 Introduction

Intelligent Tutoring Systems adapt the learning content to individual students based on the data available in the student model. The student model [1] contains information such as students' background and behaviour, and uses them to predict students' performance, knowledge, score and so on. For effective tutoring, student motivation and affective components should also be identified and considered while tailoring the learning content[2], as it is done in traditional one-on-one learning.

Baker et. al., [3] state that the affective states to be considered in ITS are boredom, frustration, confusion, delight, engaged concentration and surprise instead of the basic affective states like fear, anger and sadness. In order to identify the states, three different methodologies have been suggested [3]: human observation, using hardware sensors and machine learning techniques to identify the affective components from the student log data. While human observations and using hardware sensors are possible in a laboratory setting, it is difficult to implement them in a practical setting which might cater to a few thousand students. In such a real world system, identifying affective states from student log files is more convenient, and sometimes the only viable method.

ITS log files capture students' interaction with the ITS, such as response to questions, number of attempts and time taken for various activities (responding, reading, etc). In this paper, we propose a solution to address the problem of identifying students' affective states in commonly available ITS that contain abundant student data, but not extra features such as biometric sensors. We validate our model by comparing our results with students' self-reported data.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 612–615, 2011.

1.1 Related Work

The system in [4] identifies the frustration from Autotutor based on log data like response time and turn no. The system in [5] identifies emotions like joy/distress from student goal and actions while playing the maths game. The system in [6] identifies average frustration among the students in computer programming exercises across different labs. All the above-mentioned systems are designed for specific game or tutor. In this paper we propose a model which uses the generic features of most ITS to identify the student frustration from log data.

2 Proposed Methodology

2.1 System

Mindspark is a commercial mathematics ITS developed by Educational Initiatives, India. Mindspark is being used as a part of the school curriculum for different age groups (grades) of students [7]. In Mindspark, if the student answers consecutively three questions correctly, he receives a Sparkie (extra motivational points). If the student answers consecutively five questions correctly, she will receive a challenging question which is tougher than normal questions. If the student answers the challenge question correctly, she receives extra points. Every week, the highest Sparkie collector and the student with highest points are identified and their names are published in the Mindspark website[1].

2.2 Modeling the Affective States

In this article, we consider one of the affective states suggested by Baker et. al., [3]. According to the classic definition for frustration from psychology [8]: "Frustration refers to the blocking of behaviour directed towards a goal." The sources of frustration [8] are: "default environmental forces to block motive fulfilment, and personal inadequacies."

Table 1. Student Goals and Blocking factors

Student Goal	Blocking factor	f(blocking factor)
Goal1: To get the current question correct	The answer to the current question is wrong	$f(goal1) = (1 - a_i)$
Goal2: to get a Sparkie	If answers to last two questions are correct and to current question is wrong If answers to last question is correct and to current question is wrong.	$f_{2a} = (a_{i-2} * a_{i-1} * (1 - a_i))$ $f_{2b} = a_{i-1} * (1 - a_i)$ $f(goal2) = f_{2a} + f_{2b}$
Goal3: to get the challenge question	If answers to last four questions are correct and to current question is wrong If answers to last three questions are correct and to current question is wrong	$f_{3a} = (a_{i-4} * a_{i-3} * a_{i-2} * a_{i-1} * (1 - a_i))$ $f_{3b} = (a_{i-3} * a_{i-2} * a_{i-1} * (1 - a_i))$ $f(goal3) = f_{3a} + f_{3b}$

[1] http://www.mindspark.in/

To model frustration we consider the features captured in the log file of the Mindspark ITS. In the preliminary model, we only consider students' response to the question. Few goals and the corresponding blocking factors of the student while interacting with Mindspark are given in Table 1. We define a function $f(blockingfactor)$ corresponding to the blocking factor for each goal. 'a_i' is the student response to the i^{th} question ($a_i = 1$ if correct, $a_i = 0$ if wrong).

We define frustration index F_i at the i^{th} question based on the blocking behaviors of student goals,

$$F_i = \alpha(w_1 * f(goal1) + w_2 * f(goal2) + w_3 * f(goal3)) + (1 - \alpha)F_{i-1}$$

where w_1, w_2, w_3, α are weights. The last term in the above equation, $(1-\alpha)F_{i-1}$, which is the frustration index at the previous question (multiplied by a weight), is added to account for the cumulative effect of frustration building up over consecutive questions. $F_i = 0$ for $i = 1, 2$. The values for weights will be decided during validation process.

2.3 Validation of the Affective States Model

To validate the model, we first use the Mindspark log files to identify the affective states. The results will then be compared with data obtained from students' self-reporting. Students will be asked questions using a pop-up window in the ITS [5], to identify their level of frustration.

3 Future Work

In the future, we propose to enrich the model provided in following dimensions: 1) Expand the definition of frustration beyond goal blockage and redefine the model. 2) Include more features like time spent, difficulty level of the question from log data.

References

[1] Brusilovsky, P., Millán, E.: User models for adaptive hypermedia and adaptive educational systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 3–53. Springer, Heidelberg (2007)

[2] Woolf, B., Burleson, W., Arroyo, I., Dragon, T., Cooper, D., Picard, R.: Affect-Aware Tutors: Recognising and Responding to Student Affect. Int. J. Learn. Technol. 4(3/4), 129–164 (2009)

[3] Baker, R.S.J.d., D'Mello, S.K., Rodrigo, M. M.T., Graesser, A.C.: Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive-affective states during interactions with three different computer-based learning environments. International Journal of Human-Computer Studies 68(4), 223 (2010)

[4] D'Mello, S.K., Craig, S.D., Witherspoon, A., Mcdaniel, B., Graesser, A.: Automatic detection of learner's affect from conversational cues. User Modeling and User-Adapted Interaction 18, 45–80 (2008)

[5] Conati, C., Maclare, H.: Evaluating a probabilistic model of student affect. In: Lester, J.C., Vicari, R.M., Paraguaçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 55–66. Springer, Heidelberg (2004)

[6] Rodrigo, M.M.T., Baker, R.S.J.d.: Coarse-grained detection of student frustration in an introductory programming course. In: Proceedings of the Fifth International Workshop on Computing Education Research Workshop, ICER 2009, pp. 75–80. ACM, New York (2009)

[7] Suchismita, S., Muntaquim, B., Anupriya, G.: Mining information from tutor data to improve pedagogical content knowledge. In: Educational Data Mining (2010)

[8] Morgan, C.T., King, R.A., Weisz, J.R., Schopler, J.: Introduction to Psychology, 7th edn. McGraw-Hill Book Company, New York (1986)

Training Emotion Regulation Strategies During Computerized Learning: A Method for Improving Learner Self-Regulation

Amber Chauncey Strain, Sidney K. D'Mello, and Arthur C. Graesser

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

Abstract. A host of negative emotions such as anxiety, frustration, and boredom inevitably occur during computerized learning. These emotions can have serious negative consequences on students' metacognitive and cognitive processes and learning outcomes. Thus, students should be equipped with the ability to regulate these negative emotions in order to achieve positive learning outcomes. Building on previous research on learning-centered emotions, I (first author) propose a series of investigations into the ways in which emotion regulation strategies can be effectively implemented in an intelligent tutoring system. This paper discusses ongoing experiments, future plans, and the implications of the findings for the development of ITSs that aid in the regulation of students' emotions.

Keywords: Emotion, emotion regulation, cognitive reappraisal, ITSs.

1 Introduction

Learning episodes are replete with emotional experiences. Learners' emotions have been the focus of considerable theoretical and empirical work in the last decade [1-3]. While these endeavors have offered some insight into the kinds of emotions that are likely to arise during learning, none offer effective methods for helping students regulate or alter certain negative emotions once they occur. In contrast, several ITSs have been developed to help learners regulate their cognitive and metacognitive processes [for example, 4-6]. Because emotional processes are equally important to learning as cognitive and metacognitive processes, it follows that ITSs should also have the capacity to help learners regulate their emotions as they arise. This is the goal of the present research.

In this paper, I (first author) describe my ongoing research projects which I am completing under the guidance of the second and third authors, and propose a research plan with several goals. These goals include: (1) discovering which kinds of strategies students typically use to regulate learning-centered emotions, (2) determining which strategies are effective and which are ineffective, (3) devising creative training methods for helping students use effective emotion regulation strategies, and (4) implementing these training techniques in an intelligent tutoring system.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 617–619, 2011.

2 Background and Previous Research

Emotion regulation is defined as the physiological, behavioral, and cognitive processes that enable individuals to manage the experience and expression of emotions [7]. Research in other disciplines of psychology has identified several emotion regulation strategies such as distraction, rumination, suppression, etc. [see 7 for details]. The strategy that has received the most attention is cognitive reappraisal, or changing the way one thinks about a given situation in order to alter its emotional meaning. Previous research (not in learning contexts) has demonstrated that using cognitive reappraisal is an effective method for regulating positive and negative emotions, and can increase memory for important details. But is cognitive reappraisal an effective method for regulating learning-centered emotions and improving comprehension?

This question was investigated in a pilot study where learners were trained to use two forms of cognitive reappraisal to regulate emotions that arose during a 45-minue computerized learning session [8]. Specifically, we explored the efficacy of cognitive reappraisal by examining learners' self-reported emotions, valence, and arousal throughout the learning session, and their learning outcomes. Our findings suggested participants who used cognitive reappraisal reported positive, activating emotions like engagement, while the do-nothing control condition reported negative, deactivating emotions like disinterest. We also found that participants in the cognitive reappraisal conditions achieved better learning outcomes than the controls. This study provided some initial data into the use of cognitive reappraisal as an effective strategy for regulating emotions during learning.

3 Future Research Plans

With the knowledge that even a simple, trained cognitive reappraisal strategy could help learners regulate their emotions and achieve better comprehension, I have set forth a research plan designed to achieve the four goals listed above.

The first goal is to identify the emotion regulation strategies are used by typical learners. Although the trained cognitive reappraisal strategy used in the experiment described above were successful, it is possible that there are other strategies that are more relevant to learning. To address this issue, I plan to conduct a qualitative, survey-based experiment with approximately 100 college students from a southern university in the U.S. Participants will be provided with definitions and examples of a number of emotion regulation strategies and will be asked to rate how frequently they have used each strategy during learning. Additionally, when they indicate that a particular strategy was used, they will be required to fully describe the learning situation when it was used and if they felt that the strategy was effective. This exploratory study is expected to yield a large corpus of data about which emotion regulation strategies are frequently used, and whether students consider them to be effective in regulating their emotions during learning.

Data from this study will be used to develop scripts for training effective strategies to students. In a web-based, between-subjects experiment similar to the one described in Section 2, participants will be trained on the use of various emotion regulation strategies before a one-hour learning session. Participants in each condition will be trained on one specific reappraisal strategy, and their valence, arousal, and discrete emotions (e.g., frustration, confusion) will be collected, along with their comprehension scores. Synchronized videos of participants' face and computer screen will be used to analyze participants' affective responses to the given context. I will then compare each condition to determine which strategies are most effective for regulating learning-centered emotions and improving outcomes.

The third goal of this research is to refine the training scripts and implement them in AutoTutor [5], a mixed-initiative ITS that simulates a human tutor by holding conversation in natural dialogue. While this ITS has traditionally been used to improve learning by being responsive to students cognitive states, the proposed research will endow AutoTutor with the capacity to convey effective emotion regulation strategies to help learners simultaneously manage their emotional states as they occur. A controlled experiment will then compare this emotionally intelligent AutoTutor to the default version that only focuses on learners' cognitive states.

The regulation of emotional states during learning is an area of research that is ripe for innovation and exploration. The research plan described here is a preliminary step toward understanding this typically neglected domain, and has the potential to impact the development of future affect sensitive and responsive ITSs.

References

1. Csikszentmihalyi, M.: Flow: The psychology of optimal experience. Harper and Row, New York (1990)
2. D'Mello, S.K., Graesser, A.C.: Emotions during Learning with AutoTutor. In: Durlach, P., Lesgold, A. (eds.) Adaptive Technologies for Training and Education, Cambridge University Press, Cambridge (in press)
3. Linninbrink, L.A.: The role of affect in student learning: A multi-dimensional approach to considering the interaction of affect, motivation, and engagement. In: Schutz, P.A., Pekrun, R. (eds.) Emotion in Education, pp. 13–36. Elsevier, Amsterdam (2007)
4. Conati, C., VanLehn, K.: Toward computer-based support of meta-cognitive skills: a computational framework to coach self-explanation. International Journal of Artificial Intelligence in Education 11, 398–415 (2000)
5. Graesser, A.C., Lu, S., Jackson, G.T., Mitchell, H., Ventura, M., Olney, A., Louwerse, M.: AutoTutor: A tutor with dialogue and natural language. Behavioral Research Methods, Instrumentation, and Computation 36, 180–193 (2004)
6. Aleven, V., Koedinger, K.: An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. Cognitive Science 26, 149–179 (2002)
7. Gross, J.J., Thompson, R.A.: Emotional regulation: Conceptual foundations. In: Gross, J.J. (ed.) Handbook of Emotion Regulation, pp. 3–26. Guilford Press, New York (2007)
8. Chauncey Strain, A., D'Mello, S.: If you can't change it, change the way you think about it: Training on the use of emotion regulation strategies during learning. Poster Presentated at the 15th International Conference on Artificial Intelligence in Education, Christchurch, New Zealand (2011)

Interactive Events Summary

H. Chad Lane[1] and Brent Martin[2]

AIED 2011 Interactive Events co-chairs
[1] Institute for Creative Technologies
University of Southern California
Playa Vista, CA USA
lane @ict.usc.edu
[2] Department of Computer Science and Software Engineering
University of Canterbury
Christchurch 8140, New Zealand
brent.martin@canterbury.ac.nz

The AIED 2011 organizing committee is pleased to present eleven interactive events at the 15th International Conference on Artificial Intelligence in Education, held in Auckland, New Zealand. Interactive Events provide conference attendees a chance to experience many of the intelligent learning environments that the AIED community is building, from a learner's point of view. Attendees can ask questions of the researchers and students who have developed the systems and discuss new features and plans for the future. This year's program includes systems that support the learning sciences from all directions, including students, educators, and experimental researchers. Attendees can see work related to pedagogical agents, authoring systems, experimental tools, and educational games.

Three of the events involve the use of teachable agents, defined as systems that engage learners through learning-by-teaching. These include *Betty's Brain* (Segedy, et al.), *DynaLearn* (Beek, et al.), and *Brick Game* (Silvervarg, et al.). Three events will demonstrate the use of game-based approaches to learning including *Monkey's Revenge* (Rai, et. al.), *Annie and FixIt* (Thomas, et al.), and again, *Brick Game*. Two systems will demonstrate current approaches to pedagogical authoring, including *ASTUS* (Lebeau, et al.) and *SimStudent* (Matsuda, et al.). Two tutoring systems will be part of the Interactive Event program, including the constraint-based *EER-Tutor* for database systems (Weerashinghe, et al.) and *Beetle II* (Dzikovska, et al.), a dialogue system for reflection on circuit repair. Finally, *Inquire for iPad* (Spaulding, et al.), an interactive Biology textbook and *DataShop* (Stamper, et al.), a data repository and suite of analysis tools, will also be presented as interactive events.

In sum, the interactive events program at AIED 2011 is a highly international and thorough representation of contemporary research in the learning sciences. Full abstracts for all eleven AIED 2011 interactive events appear in these proceedings.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 619, 2011.

Knowledgeable Feedback via a Cast of Virtual Characters with Different Competences

Wouter Beek[1], Jochem Liem[1], Floris Linnebank[1], René Bühling[2], Michael Wißner[2], Esther Lozano[3], Jorge Gracia del Río[3], and Bert Bredeweg[1]

[1]University of Amsterdam, Informatics Institute, Amsterdam, Netherlands
[2]University of Augsburg, Multimedia Concepts and Applications, Augsburg, Germany
[3]Universidad Politécnica de Madrid, Ontology Engineering Group, Madrid, Spain

DynaLearn (http://www.DynaLearn.eu) develops a cognitive artefact that engages learners in an active learning by modelling process to develop conceptual system knowledge. Learners create external representations using diagrams. The diagrams capture conceptual knowledge using the Garp3 Qualitative Reasoning (QR) formalism [2]. The expressions can be simulated, confronting learners with the logical consequences thereof. To further aid learners, DynaLearn employs a sequence of knowledge representations (Learning Spaces, LS), with increasing complexity in terms of the modelling ingredients a learner can use [1]. An online repository contains QR models created by experts/teachers and learners. The server runs semantic services [4] to generate feedback at the request of learners via the workbench. The feedback is communicated to the learner via a set of virtual characters, each having its own competence [3]. A specific feedback thus incorporates three aspects: content, character appearance, and a didactic setting (e.g. Quiz mode). In the interactive event we will demonstrate the latest achievements of the DynaLearn project. First, the 6 learning spaces for learners to work with. Second, the generation of feedback relevant to the individual needs of a learner using Semantic Web technology. Third, the verbalization of the feedback via different animated virtual characters, notably: Basic help, Critic, Recommender, Quizmaster & Teachable agent.

References

1. Bredeweg, B., Liem, J., Beek, W., Salles, P., Linnebank, F.: Learning spaces as representational scaffolds for learning conceptual knowledge of system behaviour. In: Wolpers, M., Kirschner, P.A., Scheffel, M., Lindstaedt, S., Dimitrova, V. (eds.) EC-TEL 2010. LNCS, vol. 6383, pp. 46–61. Springer, Heidelberg (2010)
2. Bredeweg, B., Linnebank, F.E., Bouwer, A.J., Liem, J.: Garp3 — Workbench for qualitative modelling and simulation. Ecological Informatics 4(5-6), 263–281 (2009)
3. Mehlmann, G., Häring, M., Bühling, R., Wißner, M., André, E.: Multiple agent roles in an adaptive virtual classroom environment. In: Proc. of the 10th Int. Conf. on Intelligent Virtual Agents, pp. 250–256. Springer, Heidelberg (2010)
4. Gracia, J., Liem, J., Lozano, E., Corcho, O., Trna, M., Gómez-Pérez, A., Bredeweg, B.: Semantic techniques for enabling knowledge reuse in conceptual modelling. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part II. LNCS, vol. 6497, pp. 82–97. Springer, Heidelberg (2010)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 620, 2011.

Adaptive Intelligent Tutorial Dialogue in the BEETLE II System

Myroslava O. Dzikovska[1], Amy Isard[1], Peter Bell[1], Johanna D. Moore[1],
Natalie B. Steinhauser[2], Gwendolyn E. Campbell[2],
Leanne S. Taylor[3], Simon Caine[3], and Charlie Scott[3,*]

[1] School of Informatics, University of Edinburgh, Edinburgh, United Kingdom
{m.dzikovska,amy.isard,p.bell,j.moore}@ed.ac.uk
[2] Naval Air Warfare Center Training Systems Division, Orlando, FL, USA
[3] Kaegan Corporation, 12000 Research Parkway, Orlando, FL 32826-2944

In this interactive event we present Beetle II, a tutorial dialogue system designed to accept unrestricted language input and to support experimentation with different approaches to tutoring. Encouraging students to produce explanations and giving them detailed feedback is important for effective learning (e.g., [2]). But adding this capability to existing ITS remains a major challenge, due to the limitations of the existing natural language processing techniques. Statistical approaches like Latent Semantic Analysis have been used to interpret long student explanations. However, they require extensive pre-authoring, including anticipating a range of possible correct and incorrect answers, and manually recording tutor's feedback for every possible tutoring situation.

The Beetle II tutor asks students to explain their reasoning and accepts complex sentence-long answers to such open-ended questions. It avoids extensive pre-authoring by using a deep parser and interpreter, together with a tutoring and generation module, to automatically generate tutoring feedback adapted to the system's assessment of the student's answer and previous dialogue history.

The system has undergone a successful evaluation in 2009 [1], which found significant learning gains for students interacting with the system. We collected a rich data set which enables investigating various aspects of tutorial dialogue, e.g., differences between human-human and human-computer interaction; the impact of language understanding problems on learning gain and user satisfaction; ways to improve language understanding techniques and tutoring strategies for use in Intelligent Tutoring Systems. The event participants will interact with the system trying to complete an exercise and discover the correct answer.

References

1. Dzikovska, M., Bental, D., Moore, J.D., Steinhauser, N.B., Campbell, G.E., Farrow, E., Callaway, C.B.: Intelligent tutoring with natural language support in the Beetle II system. In: Proceedings of ECTEL 2010 (2010)
2. Nielsen, R.D., Ward, W., Martin, J.H.: Learning to assess low-level conceptual understanding. In: Proceedings 21st International FLAIRS Conference (2008)

[*] This work has been supported in part by US Office of Naval Research grants N000140810043 and N0001410WX20278.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 621, 2011.

Authoring Step-Based ITS with ASTUS: An Interactive Event

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 the amount of effort needed to build them. Typically, the main factor behind these efforts is the model of 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 support for the development of tutors for well-defined tasks in a wide range of task domains. In such context, building a framework based on a generative model of the task domain was deemed the most interesting approach because it appeared as the only one leading to comprehensive, flexible and re-usable pedagogical behaviors. For instance, the tutor is able not only to show next-step hints, but to generate them by instantiating domain-independent templates with domain-specific knowledge components.

ASTUS's knowledge representation system is based on manipulable knowledge components that encode tutored skills and "black-box" knowledge components that make operational the already mastered ones. Using an authoring language (prototyped with a Groovy-based Domain-Specific Language), the model can be encoded in coherent, easy-to-navigate files, similarly to typical source files. Tools for debugging and visualization are available at runtime.

Our first step with ASTUS was to reproduce tutors built with a comparable framework, for example we replicated a "scatter plot" tutor created with the Cognitive Tutors' "TDK" and we simultaneously developed a "multi-column subtraction" tutor using CTAT's Jess-based Cognitive Tutors and ASTUS. For the ASTUS-based tutor, we then reproduced the pedagogical behavior of the original tutor thanks to domain-independent pattern instead of domain-specific efforts.

As the ITS 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 authors skilled in knowledge-based systems. We are aware that our approach, 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.

Participants will interact with different tutors to observe the pedagogical behaviors offered by the framework and will be walked through authoring a change to a model.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 622, 2011.

Learning by Teaching SimStudent – Interactive Event

Noboru Matsuda[1], Victoria Keiser[1], Rohan Raizada[1], Gabriel Stylianides[2],
William W. Cohen[1], and Kenneth R. Koedinger[1]

[1] School of Computer Science, Carnegie Mellon University
5000 Forbes Ave. Pittsburgh PA 15213 USA
[2] Department of Education, University of Oxford
15 Norham Gardens, Oxford, OX2 6PY UK
{noboru.matsuda,keiser,wcohen,koedinger}@cs.cmu.edu
rraizada@andrew.cmu.edu, gabriel.stylianides@education.ox.ac.uk

SimStudent is an educational software infrastructure which is designed to leverage the tutor effect in an on-line learning environment. Tutor effect is the phenomenon that students learn when they teach others. SimStudent allows students to learn by teaching a computer agent instead of their peers. 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 [1].

In this learning environment, the goal of a student is to tutor SimStudent well enough so that SimStudent passes the built-in quiz prepared by the instructor. The student poses problems for SimStudent to solve, provides feedback for the step SimStudent performs, and provides a hint for any steps that SimStudent cannot perform correctly. To provide a hint, the student simply performs the step.

Additional options will provide self-explanation and game show features. Self-explanation allows SimStudent to ask students to explain why a step is incorrect or what doing a certain step will accomplish. The game show attempts to motivate students, by allowing the SimStudents that they have tutored to compete against one another in an equation solving contest.

The SimStudent program is significant both as a potential educational tool and as a research mechanism. An initial study shows that students learned by using SimStudent if they meet a certain threshold for prior knowledge of solving algebraic equations. This suggests that students will be able to use SimStudent to hone their algebra skills through tutor learning, without requiring that a tutee is available. SimStudent also allows researchers to study the conditions which facilitate tutor learning without the risk of hurting tutees' learning and while controlling for tutee variance. Self-explanation, game show motivation and meta-tutor assistance features are designed to study some of our hypotheses governing the factors of tutor learning.

Reference

1. Matsuda, N., Yarzebinski, E., Keiser, V., Raizada, R., Stylianides, G., Cohen, W.W., et al.: Learning by Teaching SimStudent – An Initial Classroom Baseline Study comparing with Cognitive Tutor. In: Biswas, G., et al. (eds.) AIED 2011. LNCS (LNAI), vol. 6738, pp. 238–246. Springer, Heidelberg (2011)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 623, 2011.

Monkey's Revenge: Coordinate Geometry Learning Environment with Game-Like Elements

Dovan Rai and Joseph E. Beck

Computer Science Department, Worcester Polytechnic Institute
{dovan,josephbeck}@wpi.edu

Educational games intend to make learning more enjoyable, but at the potential cost of compromising learning efficiency. Therefore, instead of creating educational games, we have created a learning environment with game-like elements: the elements of games that are engaging. Our approach is to assess each game-like element in terms of benefits such as enhancing engagement as well as its costs such as sensory or working memory overload, with the goal of maximizing both engagement and learning. We created Monkey's Revenge, a coordinate geometry learning environment with game –like elements such as narrative, immediate visual feedback, personalization, collecting badges, etc. The tutor basically consists of a series of 8th grade (approximately 13-year olds) coordinate geometry problems wrapped in a visual cover story. In the narrative, Mike, a boy is thrown out of 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 equation of the line of the path of the ball. The math content gets more advanced as a student progresses through the story. Students get immediate visual feedback on their response. For example, if a student puts banana at the wrong coordinate, the monkey can reach it and will eat the banana. We are using a very simple and minimalistic approach so as not to overwhelm students who are already struggling with the content.

We built four versions of this tutor with different degree of "game-like" (one without visual feedback, one without narrative and a basic tutor with the same hints and bug messages as the other three versions). Based on a study with 297 students, that students who had more "game-like" tutor reported more liking of the tutor but we found no conclusive difference in learning gain. We had made a very conservative progression from tutor towards game adding as little detail as possible. So, our first concern was to attain optimal engagement so as not to leave students disenchanted. Based on our next study focusing on learning gain, we will decide whether we have to enhance or scale back game-like elements. With such iterative process, we aim to find a "sweet spot" in the tutor game space where we can find optimal engagement and learning.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 624, 2011.

Knowledge Construction with Causal Concept Maps in a Teachable Agent Environment

James R. Segedy, John S. Kinnebrew, and Gautam Biswas

Vanderbilt University, Nashville, TN 37235, USA
{James.R.Segedy,John.S.Kinnebrew,Gautam.Biswas}@vanderbilt.edu

We have developed Betty's Brain [1], a computer-based learning environment that employs the learning-by-teaching paradigm to foster students' acquisition of science knowledge and self-regulated learning strategies. The system provides students with opportunities for self-directed, open-ended learning in science. In this learning environment, students are given a knowledge construction task in which they teach a virtual agent by engaging in an iterative process of reading source material and structuring their knowledge in a causal concept map for a particular science domain (e.g., ecology or thermo-regulation). The agent, then, can use this map to answer questions and take quizzes.

The act of teaching an agent is a self-directed and open-ended activity where one explores, integrates, and structures knowledge first for oneself, and then for others. Our previous work has shown that students are motivated to teach and interact with their teachable agent, and this motivation can further enhance their learning.

The learning process is augmented with *social interactions*. Both the teachable agent and a knowledgeable mentor agent provide conversational feedback about the student's progress and activity patterns. These restricted, popup-based conversations help students (1) understand the science topic, (2) build the correct concept map, and (3) acquire general-purpose problem-solving and metacognitive strategies.

This interactive event will showcase some of the key features of the Betty's Brain system by allowing participants to teach a causal concept map about global climate change. First, they will gain familiarity with the overall task of knowledge construction and map-building in the system: they will search hypertext resources for causal relationships and use them to construct a map. Second, they will learn about monitoring features for exploring their own knowledge explicitly. They will accomplish this by: asking their teachable agent to explain how to answer questions involving complex chains of reasoning, asking their agent to take a quiz, and taking notes about what they know and don't know. Finally, they will encounter conversational dialog from the agents to prompt them to monitor and regulate their own learning along the way.

Reference

1. Biswas, G., Leelawong, K., Schwartz, D., Vye, N., Vanderbilt, T.: Learning by teaching: A new agent paradigm for educational software. Applied Artificial Intelligence 19(3), 363–392 (2005)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 625, 2011.

An Educational Math Game with a Teachable Agent and a Social Chat

Annika Silvervarg[1], Lena Pareto[2], Magnus Haake[3],
Thomas Strandberg[3], and Agneta Gulz[1,3]

[1] Department of Computer Science, Linköping University, Sweden
{Annika.Silvervarg,Agneta.Gulz}@liu.se
[2] Media Production and Informatics Departments, University West, Sweden
Lena.Pareto@hv.se
[3] Lund University Cognitive Science, Sweden
{Magnus.Haake,Thomas.Strandberg,Agneta.Gulz}@lucs.lu.se

We present an educational math game, including a teachable agent and a social chat, that trains basic arithmetic skills with a focus on grounding base-ten concepts in spatial representations. It employs a board-game design with a variety of different sub-games, game modes and levels of difficulty. When a student has learnt to play one of the sub-games, she may teach it to her Teachable Agent (TA). In the *observation mode* the TA "watches" the student play and picks up on game rules and on the student's responses to multiple-choice questions, such as "Why did you choose this card?" Proper (or improper) choices of cards and answers promote corresponding skills in the TA throughout the game. In the *try-and-be-guided mode*, the agent is allowed to propose cards. The student either accepts the agent's suggestion or rejects it and exchanges the agent's card for another one. Again the agent asks for the reasons for the student's behaviour, using the multiple-choice format. In other words, the basic game with the TA contains a form of *on-task* conversation between agent and student. But the game architecture also has been extended with *a chat* where the student can engage in conversation with the TA, writing freely by means of the keyboard and bring up basically any topic in a chat-like manner. We refer to this as *off-task conversation* and distinguish within it between on-domain conversation and off-domain conversation, the former referring to chat conversation related to school, math, the math game, etc., and the latter to any other topic. One reason to include off-task conversation is to enrich the game and its motivational qualities for the age group in question (12-14 year olds). Another is to be able to explore whether such a conversational module can enable pedagogical interventions, such as supporting pupils math self efficacy and change negative attitudes toward math in general. Notably the on-task and off-task conversations have very different formats, but are still designed as two interrelated and complementary activities. A recent study [1] indicates that the added off-task conversation module can i) improve students' game experience, ii) improve learning outcomes, and iii) engage learners in voluntary on-domain chat.

Reference

1. Gulz, A., Haake, M., Silvervarg, A.: Extending a Teachable Agent with a Social Conversation Module – Effects on Student Experiences and Learning. In: Biswas, G., et al. (eds.) AIED 2011. LNCS (LNAI), vol. 6738, pp. 131–138. Springer, Heidelberg (2011)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 626, 2011.

Inquire for iPad: A Biology Textbook That Answers Questions

Aaron Spaulding[1], Adam Overholtzer[1], John Pacheco[1], Jing Tien[1],
Vinay K. Chaudhri[1], Dave Gunning[2], and Peter Clark[2]

[1] SRI International, 333 Ravenswood Ave, Menlo Park, CA, 94025, USA
{spaulding,overholtzer,pacheco,tien}@ai.sri.com,
vinay.chaudhri@sri.com
[2] Vulcan Inc., 505 Fifth Ave, Suite 900, Seattle WA, 98104, USA
{DaveG,PeterC}@vulcan.com

Textbooks are increasingly moving into the digital realm, which presents an opportunity for them to evolve from providing the reader with a static, linear experience, into an interactive application that can adapt to a student as well as to specific learning goals. As a step in this direction, we present *Inquire: Biology*, an electronic textbook that provides question-answering capability.

Inquire: Biology, is a novel electronic textbook that runs on an iPad and embeds in it a rich Biology knowledge base and reasoning system. As a student reads the textbook using *Inquire*, he or she may ask it questions about aspects of the material that are difficult to understand. *Inquire* can provide answers to these questions as well suggest additional questions based on the student's context.

Inquire is an iPad application consisting of three main components: (1), a Biology textbook, which users can highlight and annotate as desired; (2), the question-asking component, which consists of suggested questions, an option for the user to ask freeform questions, and answers; and (3), a set of glossary pages, which contain text capturing the key points about a concept and interactive concept maps.

The *Inquire* application connects to a server running our AURA[1] system, which contains a knowledge base (KB) of biology concepts created from a Biology textbook. AURA can interpret questions posed in simplified natural language, and can produce answers and explanations for questions by reasoning over the KB. For a given section of a textbook that a student may be reading, AURA generates questions, which can help a student review the material they have read and explore the sections of the textbooks that they may not have read.

We have conducted an initial round of user studies, and received invaluable input from a number of different domain experts. The prototype indicates that our reasoning and question asking technology can add useful functionality to an electronic textbook. *Inquire* sets up an inspiring vision towards the textbook of future and can provide a concrete platform in which other educational researchers can plug in their pedagogical approaches and show immediate impact.

Acknowledgement. This work has been funded by Vulcan, Inc.

Reference

1. Gunning, D., Greaves, M., Chaudhri, V., et al.: Project Halo Update–Progress Towards Digital Aristotle. AI Magazine 31(3) (2010)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 627, 2011.

DataShop: A Data Repository and Analysis Service for the Learning Science Community (Interactive Event)

John C. Stamper[1], Kenneth R. Koedinger[1], Ryan S.J.d. Baker[2], Alida Skogsholm[1], Brett Leber[1], Sandy Demi[1], Shawnwen Yu[1], and Duncan Spencer[1]

[1] Carnegie Mellon University, Human-Computer Interaction Institute
[2] Worcester Polytechnic Institute, Department of Social Science and Policy Studies
{jstamper,krk,alida,bleber,sdemi,shanwen,dspencer}@cs.cmu.edu,
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. As of April 8, 2011, over 270 datasets are stored including over 58 million student actions and over 165,000 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 repository users a central hub to satisfy long term data management needs. DataShop also has 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.

In this interactive demo we will show how to use the DataShop tools to explore log data and to create new knowledge component models that fit the data. 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.d., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J., Ventura, S., Pechenizkiy, M., Baker, R.S.J.d.: A Data Repository for the EDM community: The PSLC DataShop. In: Romero, C. (ed.) Handbook of Educational Data Mining, pp. 43–56. CRC Press, Boca Raton (2010)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 628, 2011.

Annie and FixIt: Showcasing Dynamic Guidance for Task-Based Exploratory Learning

James M. Thomas and R. Michael Young

Digital Games Research Center
Department of Computer Science
North Carolina State University, Raleigh, NC USA
jmthoma5@ncsu.edu, young@csc.ncsu.edu

The ITS field has benefitted from a shared consensus of proven techniques for intelligent scaffolding [4], but common techniques to solve the unique challenges of exploratory or inquiry-based tutoring have proven more elusive [3, 1].

Exploratory environments provide students with freedom to choose different courses of action. This complicates the tutor's ability to know what the student it trying to do, which introduces uncertainty in knowing whether or not a student has a misconception about the domain. When the tutor decides a misconception exists, it is difficult to know when is the right time to provide support to remediate that misconception, as the student may have changed focus to a different task. As others have 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."

Our system addresses these problems by leveraging a well-understood computational model of actions and the causal relationships between them used in automated planning.We have previously published the details of the design for this system [2], and recently completed the first full experimental evaluation of the system, which is being submitted to AIED 2011 as a full conference paper.

References

1. 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)
2. 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)
3. Van Joolingen, W., De Jong, T., Dimitrakopoulou, A.: Issues in computer supported inquiry learning in science. Journal of Computer Assisted Learning 23(2), 111–119 (2007)
4. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education 16(3), 227–265 (2006)

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 629, 2011.

Facilitating Adaptive Tutorial Dialogues in EER-Tutor

Amali Weerasinghe and Antonija Mitrovic

Intelligent Computer Tutoring Group, University of Canterbury, New Zealand
amali.weerasinghe@pg.canterbury.ac.nz,
tanja.mitrovic@canterbury.ac.nz

EER-Tutor is a constraint-based intelligent tutoring system that teaches conceptual database design. Students are provided a problem solving environment to design a data model for a real world scenario. We enhanced EER-Tutor with adaptive tutorial dialogues to facilitate discussion of mistakes in a student solution. The dialogues discuss errors in the current problem context as well as the relevant domain concepts. The dialogues are customised based on the student model.

Database design is an ill-defined task. The final outcome i.e. the data model is defined in abstract terms, but there is no algorithm to find it. The tasks supported by other existing dialogue-based tutoring systems support are well-defined: (such as Mathematics, Physics) problem-solving is well-structured, and therefore the explanations that are expected from the learners can be clearly defined [1]. EER-Tutor allows the students to work on any part of the solution facilitating the ill-defined nature of the task. The constraint-based methodology (CBM) that is used to develop EER-Tutor does not impose any restrictions on which on the order at which a student arrives at a solution.

Our model for supporting dialogues consists of three parts: an error hierarchy, tutorial dialogues and rules for adapting them. The error hierarchy categorizes all error types in a domain. At the leaf level, an error type is associated with one or more violated constraints. Remediation is facilitated through dialogues, one of which is developed for each error type. In the case of multiple errors in a student solution, the hierarchy is traversed to select the error most suitable for discussion and the corresponding dialogue is then initiated. Finally, the adaptation rules are used to individualize the dialogues to suit the student's knowledge and reasoning skills by controlling their timing and the exact content. In response to the generated dialogue learners are able to provide answers by selecting an option from a list.

We evaluated the effectiveness of our model in an authentic classroom environment at the University of Canterbury in March 2010. The experimental group participants received adaptive dialogues that were customised based on their student models. The control group received non-adaptive dialogues regards of their knowledge level and the explanation skills. At the end of a single 2-hour session, the performance on pre- and post-tests indicate that the experimental group learned significantly more than their peers. The experimental group also learnt a significantly higher number of constraints.

In this interactive event, the participants will have the opportunity to solve problems in EER-Tutor and engage in dialogues. They will be able to experience how the dialogues are customised based on their knowledge level and their interactions with the dialogues.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 630–631, 2011.

Reference

1. Weerasinghe, A., Mitrovic, A., Martin, B.: Towards Individualized Dialogue Support for Ill-Defined Domains. IJAIED, Special Issue on Ill-Defined Domains 19(4), 357–379 (2009)

First Workshop on Artificial Intelligence in Education to Support the Social Inclusion of Communities (AIEDSIC)

Fábio N. Akhras[1,*] and Paul Brna[2,*]

[1] Renato Archer Center of Information Technology, São Paulo, Brazil
fabio.akhras@cti.gov.br
[2] School of Informatics, The University of Edinburgh, Scotland, UK
paulbrna@mac.com

Summary. About 20 years ago, in a paper entitled "Computational Mathetics: the Missing Link of Artificial Intelligence in Education", John Self argued that AI in Education has missed its connection with formal AI, its theoretical side. Some people argued that this was necessary so that AI in Education (AIED) could be able to deliver real world applications. However, in the real world, half of the population lives with less than 3 dollars a day with many socially excluded from education, health and other basic services. Social inclusion seeks to address the needs of this population, mostly living in underdeveloped countries, and also combat factors that are socially problematic in developed countries such as poor educational attainment, unemployment, poor health/special needs, low income, crime and poor housing/local environment.

The AI in Education community has spent more than 30 years researching the design of adaptive technologies to support learning. However, the issue of supporting social inclusion has never been directly addressed. Has AI in Education also missed an important connection with the real world? We argue that AI in Education systems have a challenging role to play in helping to transform communities but we also accept that much has to be done to establish the ways in which work on AI in Education supports such activities indirectly, and to determine what future work needs to be done.

The European Union made 2010 the European Year For Combating Poverty and Social Exclusion. The key objectives were to improve public awareness and commitment at the political level to fight poverty and social exclusion while some key challenges are: to eradicate child poverty by breaking the vicious circle of intergenerational inheritance, to promote the active inclusion in the society and the labour market of the most vulnerable groups, to overcome discrimination and increase the integration of people with disabilities, ethnic minorities and immigrants and other vulnerable groups. We can start by focusing on AIED's capacity to support these aims. Therefore, the main purpose of this workshop is to identify and discuss the challenges that arise in addressing issues of supporting the social inclusion of communities in the context of AI in Education research and lay the groundwork for future workshops in this area.

Programme Committee: Robert Aiken (Temple Univ., USA), Nicolas Van Labeke (Univ. of Nottingham), Rose Luckin (Inst. of Education, UK), Jack Mostow (Carnegie Mellon Univ., USA), Gilda Olinto (Brazilian Inst. of Information on Science and Technology, IBICT, Brazil), Natasha Queiroz (Federal Univ. of Paraiba, UFPB, Brazil), Rafael Morales (Univ. of Guadalajara, Mexico).

[*] Workshop Co-Chairs.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 632, 2011.

International Workshop on Learning by Modelling in Science Education

Bert Bredeweg[1,*] and Paulo Salles[2,*]

[1] University of Amsterdam, Informatics Institute, Amsterdam, Netherlands
b.bredeweg@uva.nl
[2] University of Brasília, Institute of Biological Sciences, Brazil
psalles@unb.br

Summary. Modelling is nowadays a well-established methodology in the sciences, supporting the inquiry and understanding of complex phenomena and systems in the natural, social and artificial worlds. Hence its strong potential as pedagogical approach fostering students' learning of scientific concepts and skills, in a systemic perspective. Modelling helps learners to express and externalise their thinking; visualise and test components of their theories; and make materials more interesting. Modelling and simulation in education can thus make a significant contribution to improve science learning.

Different kinds of modelling environments have been created. Environments such as *NetLogo*, *Stella* and *Model-It* are some examples that offer innovative environments in which students can construct their own models and simulations to solve problems of interest to them. More recent advancements have delivered interactive diagrammatic representations based on Qualitative Reasoning, e.g. *Betty's Brain*, *Vmodel*, and *DynaLearn*. Environments such as these allow learners to view the invisible and examine complexity in ways that were previously impossible.

Learning by Modelling (LbM) may contribute to students' learning of scientific concepts and skills. LbM tools implemented as constructivist environments have the potential to support the learners' gradual construction of knowledge and mastery of skills, and to increase their motivation to explore scientific phenomena. Moreover, LbM implies the acquisition of skills and perspectives that may become in the long-term powerful intellectual tools for addressing systemic phenomena in new situations and contexts. Hence its status as promising approach for science education.

Computational modelling can serve two roles in approaching these issues. First, creating and evaluating models can serve to help learners deepen their scientific knowledge and skills, and become aware of the joy of understanding scientific topics. Second, computational modelling is an excellent example of daily professional work in scientific laboratories, in which models are used to create understanding of deep and complex scientific problems.

Programme Committee: Rachel Or-Bach (Academic College of Emek Yezreel, Israel), Gautam Biswas (Vanderbilt Univ., USA), Wouter van Joolingen (Univ. of Twente, The Netherlands), Jochem Liem (Univ. of Amsterdam, The Netherlands), David Mioduser (Tel Aviv Univ., Israel), Julie-Ann Sime (Lancaster Univ., UK), Elliot Soloway (Univ. of Michigan Ann Arbor, USA), Andrew Ravenscroft (London Metropolitan Univ., UK), Michael Timms (WestEd, USA), Xiu-Tian Yan (Univ. of Strathclyde, UK).

[*] Workshop Co-Chairs.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, p. 633, 2011.

International workshop on Authoring Simulation and Game-Based Intelligent Tutoring

Paula J. Durlach*

U.S Army Research Institute
Paula.Durlach@us.army.mil

Summary. The use of scenario-based simulations and serious games for training has been well-accepted in many domains. Simulations require active processing and provide intrinsic feedback in an environment in which it is safe to make mistakes; however, reaping training benefits from this kind of training is often highly dependent on support from human instructors who select training scenarios, observe trainee behavior, and provide feedback, prompts, and reflective discussion. Applying the techniques of intelligent tutoring to simulation-based training could reduce reliance on human instructors. Schatz, Bowers, and Nicholson (2009) refer to this integration of intelligent tutoring strategies with simulation-based training as "advanced situated tutors." Advanced situated tutors include student models whose data are used to apply adaptive instructional strategies to selection of simulation events, instructional content, and instructional support.

While a small collection of advanced situated tutors exist, similar to standard intelligent tutors, creation of these systems requires a wide range of expertise and substantial resources. Many organizations that use traditional multimedia training for their personnel recognize the potential benefits of adding simulation-based or game-based elements to training, and would readily accept advanced situated tutors were it not for the high upfront costs the creation of these systems currently entail. Authoring tools for advanced situated tutors could facilitate the development process and reduce cost. This workshop will help characterize the current state of the art and identify outstanding issues and future potential approaches to meeting this objective. The workshop will be a combination of presentations and discussion.

Organizing Committee: Antonija Mitrovic (Univ. of Canterbury), Stephen Gilbert (Iowa State University), Stephen Blessing (University of Tampa).

Review Committee: Brandt Dargue (Boeing Research and Technology), Lewis Johnson, (Alelo Inc.), Allen Munro (University of Southern California), Robert Sottilare (Army Research Laboratory), Alicia Sage (Alelo, Inc.), Randy Spain (U.S. Army Research Institute), Bruce Perrin (Boeing Research and Technology), Chas Murray (Carnegie Learning), Brendon Towle (Carnegie Learning), Glenn Martin (I. for Simulation and Training, Univ. of Central Florida).

[*] Workshop Chair.

G. Biswas et al. (Eds.): AIED 2011, LNAI 6738, pp. 634, 2011.

Author Index